INFORMATION TO USERS

This manuscript has been reproduced from the microfilm master. UMI films the

text directly from the original or copy submitted. Thus, some thesis and

dissertation copies are in typewriter face, while others may be from any type of

computer printer.

The quality of this reproduction is dependent upon the quality of the copy

submitted. Broken or indistinct print, colored or poor quality illustrations and

photographs, print bleedthrough, substandard margins, and improper alignment

can adversely affect reproduction.

In the unlikely event that the author did not send UMI a complete manuscript and

there are missing pages, these will be noted. Also, if unauthorized copyright

material had to be removed, a note will indicate the deletion.

Oversize materials (e.g., maps, drawings, charts) are reproduced by sectioning

the original, beginning at the upper left-hand comer and continuing from left to

right in equal sections with small overlaps.

Photographs included in the original manuscript have been reproduced

xerographically in this copy. Higher quality 6" x 9" black and white photographic

prints are available for any photographs or illustrations appearing in this copy for

an additional charge. Contact UMI directly to order.

Bell & Howell Information and Learning 300 North Zeeb Road, Ann Arbor, MI 48106-1346 USA

800-521-0600

AN INFORMATION MODEL FOR MANAGING DESIGN CHANGES IN A COLLABORATIVE MULTI-DISCIPLINARY DESIGN ENVIRONMENT

Ahmed H.M. Mokhtar

A Thesis
in
School for Building
Faculty of Engineering and Computer Science

Presented in Partial Fulfilment of the Requirements for the Degree of

Ph.D. in Engineering (Building)

at Concordia University

Montreal, Quebec, Canada



National Library of Canada

Acquisitions and Bibliographic Services

395 Wellington Street Ottawa ON K1A 0N4 Canada Bibliothèque nationale du Canada

Acquisitions et services bibliographiques

395, rue Wellington Ottawa ON K1A 0N4 Canada

Your file Votre référence

Our file Notre rélérence

The author has granted a nonexclusive licence allowing the National Library of Canada to reproduce, loan, distribute or sell copies of this thesis in microform, paper or electronic formats.

The author retains ownership of the copyright in this thesis. Neither the thesis nor substantial extracts from it may be printed or otherwise reproduced without the author's permission.

L'auteur a accordé une licence non exclusive permettant à la Bibliothèque nationale du Canada de reproduire, prêter, distribuer ou vendre des copies de cette thèse sous la forme de microfiche/film, de reproduction sur papier ou sur format électronique.

L'auteur conserve la propriété du droit d'auteur qui protège cette thèse. Ni la thèse ni des extraits substantiels de celle-ci ne doivent être imprimés ou autrement reproduits sans son autorisation.

0-612-39624-X



Abstract

AN INFORMATION MODEL FOR MANAGING DESIGN CHANGES IN A COLLABORATIVE MULTI-DISCIPLINARY DESIGN ENVIRONMENT

Ahmed H.M. Mokhtar

The presence of incompatibility errors in construction technical documents is a major problem for the construction industry. An analysis of the production process of these documents reveals that managing design changes constitutes a main source for incompatibilities. More specifically, failure to propagate design changes among the design team is a principal cause of problems. The large amount of design data that is generated within a multi-disciplinary design environment makes this task very complex, especially when considering the fact that the involved design disciplines are, in most cases, separated both spatially and educationally. This thesis presents the development of a model that uses information technology to address the problem. The core concept of the model is a central database that functions as a repository of active building components. Each building component in this database not only carries its design data, but is also capable of recognizing the design disciplines that are affected by any change in these data and automatically send them messages. The active building components are able to perform this task because they are equipped with the necessary linking knowledge. Such knowledge is necessary to propagate the effect of a design change by one discipline on

other disciplines involved in the design of the same building. The linking knowledge is acquired from the designers and is implemented in the form of rules. A management database is also developed as part of the model's central database. This management database makes the model easily adaptable to any building configuration, an essential requirement due to the uniqueness of every building project. It also provides the model with the capability for not only tracking past design changes but also planning and scheduling future design changes as well. The model has been successfully implemented on a client-server network environment and validated using both hypothetical and real design cases.

Acknowledgements

THANKS TO GOD FOR EVERY THING I WAS ABLE TO ACHIEVE AND FOR EVERYTHING I TRIED BUT FAILED TO ACHIEVE.

I ALSO WISH TO THANK ALL THE INDIVIDUALS WHOSE ASSISTANCE AND ENCOURAGEMENT WERE INSTRUMENTAL IN THE REALIZATION OF THIS WORK. I AM DEEPLY GRATEFUL TO MY SUPERVISORS, DR. CLAUDE BÉDARD AND DR. PAUL FAZIO FOR THEIR GUIDANCE, CONSTRUCTIVE ADVICE, THOUGHTFUL RECOMMENDATIONS, AND FINANCIAL SUPPORT. I WOULD LIKE ALSO TO THANK THE EXAMINATION COMMITTEE FOR THEIR VALUABLE CRITICISM.

THE CONTINUOUS REASSURING AND ASSISTANCE FROM BOTH DR. OSAMA MOSELHI AND DR. SABAH ALKASS WERE INSPIRING IN CONTINUING THIS WORK SUCCESSFULLY. THE HELP OF MR. SYLVAIN BELANGER WAS VERY FACILITATING DURING THE COMPUTER IMPLEMENTATION PART OF THIS RESEARCH.

I AM VERY GRATEFUL TO ALL MY COLLEAGUES SPECIALLY, DR. ADEL ABDOU, DR. KHALID EL-RAYES, AND MOHAMMAD AL-HOUSEIN FOR THEIR USEFUL COMMENTS, HELPFUL ADVICE, AND SOLID SUPPORT.

VERY SPECIAL THANKS TO MY PARENTS FOR THEIR UNREPLACABLE GIVING AND CONSISTENT MOTIVATION.

MY DEEPEST APPRECIATION FOR MY WIFE AND FOR MY CHILDREN FOR THEIR LOVE, PATIENCE, AND ENCOURAGEMENT.

Table of Contents

igures	IX
elatures	XIV
ations	xıv
for User Interface Diagrams	XV
r One	
RODUCTION	
1.1.1 Background	
Research Scope and Objectives	10
Outline of the Thesis	12
R Two ORMATION TECHNOLOGY IN DESIGN:	
Introduction	13
2.2.1 Agent-Based Federated Architecture Approach	15 18 19
2.3.1 On Definitions and Implementation	2\ 2\ 2\
	Motivation and Problem Definition

	2.3.3 Discussion	28
2.4	Design Process Management	28
	2.4.1 Design Changes	
	2.4.2 Design Information.	
	2.4.3 Design Rationale	
2.5	Summary of Findings	32
Снарте	r Three	
	POSED INFORMATION MODEL: ONCEPTUAL PERSPECTIVE	
3.1	Objectives and Model Characteristics	34
3.2	The Model Architecture	35
	3.2.1 Basic Parts	
	3.2.2 The Concept of Active Building Components	
	3.2.3 Propagating Current Design Changes	
	3.2.4 Tracking Past Design Changes	
	3.2.5 Planning and Scheduling Future Design Changes	
3.3	Discussing the Model Advantages and Limitations	52
3.4	Summary	55
Снарте	R FOUR	
	E CENTRAL PROJECT-DATABASE: ETAILED PERSPECTIVE	
4.1	Introduction	56
4.2	Flexible repository of active building components	57
	4.2.1 Providing Flexibility	
	4.2.2 Making the Building Components Active	62
4.3	Assembly and Configuration	
	4.3.1 Assembly of the Building-Components-Database of a New Project	
	4.3.1.1 The General-Database	
	4.3.1.2 The Assembly Process	
	4.3.2 Configuring the Project-Database of an Existing Project	
	4.3.2.2 Configuring Prebuilt Rules	
	4.3.2.3 Configuring the Project Menu	
4.4	Summary	79

CHAPTER FIVE

MANAGING DESIGN CHANGES
A DETAILED PERSPECTIVE

5.1	Introduction	80
5.2	Propagating Current Design Changes	80
0.2	5.2.1 Objectives	80
	5.2.2 Methodology	
	5.2.2.1 Acquire the linking knowledge	82
	5.2.2.2 React to Design Changes	88
	5.2.2.3 Deliver Design Change Messages	92
	5.2.3 User Interface	94
5.3	Tracking Past Design Changes	98
J.J	5.3.1 Objectives	98
	5.3.2 Methodology	98
	5.3.3 User Interface	101
	J.J.J. User Interface	
5.4	Planning and Scheduling Future Design Changes	103
	5.4.1 Objectives	103
	5.4.2 Methodology	106
	5.4.2.1 Collect Data	106
	5.4.2.2 Organize Changes	112
	5.4.2.3 Schedule Changes	115
	5.4.2.4 Calculate Costs and Durations	116
	5.4.3 User Interface	119
5.5	Summary	124
Снарте	er Six	
	LIDATION OF THE MODEL	
6.1	Introduction	125
6.2	Computing Environment	123
63	Validation	126
0.5	6.3.1 Approach	126
	6.3.2 First Level of Validation.	
	6.3.3 Second Level of Validation	
	6.3.3.1 Assembly and Population of the Project-Database	120
	6.3.3.2 Propagating Current Design Changes	135
	6.3.3.3 Tracking Past Design Changes	139
	6.3.3.4 Planning and Scheduling of Future Changes	
	6.3.4 Third Level of Validation	
6.4	Summary and Conclusions	153
	11	v. n. n. n. m

Снарте	ER SEVEN	
Sun	MMARY, CONTRIBUTIONS, AND RECOMMENDATION	S
7.1	Summary	159
	Contributions	
7.3	Recommendations for future work	164
Rei	FERENCES	166
APPEND	oix A	
DA'	TABASE STRUCTURE IN MS ACCESS	
A.1	Management-database	
	A.1.1 Conceptual Data-Structure	
Δ 2	General-Database	
11.2	A.2.1 Conceptual Data-Structure	177
· -	BROUTINES FOR PLANNING AND SCHEDULING	
	SIGN CHANGES	1.70
B.1	Subroutine to generate the various change-paths	
	B.1.2 Generate the first change-path	179
	B.1.3 Generate the other change-paths	181
B.2	Subroutine to generate the logical sequence of design changes	
	B.2.1 Initializations B.2.2 Process the initial design change	
	B.2.3 Process the rest of the design changes in the change-path	184
	B.2.4 Generate the dummy design changes	186
Append	IX C	
Lis	TINGS OF LEVEL5 OBJECT MODULES	
C.1	Configuration Module	190

C.2	Data Manipulation Module	.208
C.3	Message Module	.226
C.4	Tracking Module	.231
C.5	Virtual Change Module	.239
C.6	Planning and Scheduling Module	.250

List of Figures

Figure 1-1 Production process of construction technical documents during detailed design stage.	4
Figure 2-1 Organization of design agents in federation architecture. (Khedro et al. 1994, redrawn)	16
Figure 3-1 Architecture of proposed information model.	36
Figure 3-2 Interface for data manipulation module.	38
Figure 3-3 Configuration of a building component using configuration module	40
Figure 3-4 Assembly of building components in building-components-database from building systems available in general-database	40
Figure 3-5 Concept of active building components	41
Figure 3-6 Message received by envelope designer informing of design change made by architect	44
Figure 3-7 Message received by structural engineer informing of design change made by envelope designer.	45
Figure 3-8 Interface for tracking module.	47
Figure 3-9 Results of request made by design manager to track design changes made to a building component.	47
Figure 3-10 Results of request made by design manager to track automated messages that are sent because of a design change.	48
Figure 3-11 Scenario for interrelated design changes	49
Figure 3-12 Alternate change-paths for scenario in Figure 3-11	51
Figure 3-13 Bar chart for alternate change-path (a) in Figure 3-12	52
Figure 4-1 Sample of data in data-table "Window" which describes instances of the building component "Window"	59
Figure 4-2 Configuration of the building component "Circular_Window" using configuration module.	61
Figure 4-3 Sample of data in Data-Table 4-3 that describes data-structure of the	***

building components "Window" and "Circular_Window"	62
Figure 4-4 Combination of relational database and object-oriented technologies	63
Figure 4-5 Sample of data in Data-Table 4-4 that describes data-structure of variety of building components.	68
Figure 4-6 Sample of data in Data-Table 4-5 that defines building components that belong to a building system	68
Figure 4-7 Sample of data in Data-Table 4-6 that records current projects	69
Figure 4-8 Sample of data in Data-Table 4-7 that describes some prebuilt rules	70
Figure 4-9 Sample of data in Data-Table 4-8 that describes configuration of project menu.	75
Figure 4-10 Configuration of a prebuilt rule using configuration module	76
Figure 4-11 Project menu as it functions in data manipulation module.	77
Figure 4-12 Interface to configure project menu to allow access to available building components.	77
Figure 4-13 Diagram to illustrate user interface in configuration module	78
Figure 5-1 Sample of data in Data-Table 5-1 that describes some prebuilt rules	84
Figure 5-2 Sample of data in Data-Table 5-2 that describes some dynamically built rules.	84
Figure 5-3 Sample of warning message about design change that results in major design modifications.	85
Figure 5-4 Acquisition of dynamically built rule using relation-acquiring sub- module	87
Figure 5-5 Sample of data in Data-Table 5-3 that describes some message elements	90
Figure 5-6 Diagram to illustrate user interface in data manipulation module	96
Figure 5-7 Diagram to illustrate user interface in message module	97
Figure 5-8 Sample of data in Data-Table 5-4 that describes some design changes and enables their tracking	99

Figure 5-9 Diagram to illustrate user interface in tracking module	102
Figure 5-10 Scenario for interrelated design changes	105
Figure 5-11 Starting a new case in virtual change module.	107
Figure 5-12 Data provided by designer for proposed design change	108
Figure 5-13 Response to virtual change message in virtual change module	110
Figure 5-14 Data provided by designer for proposed design change	111
Figure 5-15 Sample of data in Data-Table 5-6 that defines sequence of changes	111
Figure 5-16 Sample of data in Data-Table 5-7 which is collected from designers through virtual change module	112
Figure 5-17 Sample of data in Data-Table 5-8 that defines alternative change-paths	113
Figure 5-18 Alternate change-paths for scenario in Figure 5-10	114
Figure 5-19 Bar chart for alternate change-path (b) in Figure 5-18	116
Figure 5-20 Partial data from Data-Table 5-7 that is used for calculating costs and durations.	118
Figure 5-21 Diagram to illustrate user interface in virtual change module	121
Figure 5-22 Interface to show historical data about certain design change	122
Figure 5-23 Diagram to illustrate user interface in planning and scheduling module	122
Figure 5-24 Interface to show detailed data of a design change.	123
Figure 5-25 Interface to show results of all alternative change-paths so as to enable comparing them	123
Figure 6-1 Sample of statements recorded by "history" tool in LEVEL5 OBJECT	127
Figure 6-2 Assembly of building components in building-components-database from building systems available in general-database	131
Figure 6-3 Adding building component from general-database to building- component-database	132
Figure 6-4 Configuration of a building component using configuration module	132

Figure 6-5 Input data to structural design translate	ed to prebuilt rule133
Figure 6-6 Input data to structural design translate	ed to prebuilt rule133
Figure 6-7 Input data to HVAC design translated	to prebuilt rule134
Figure 6-8 Input data to envelope design translate	ed to prebuilt rule134
Figure 6-9 Dynamically built rule acquired from the conditions used to design depth	structural designer defining one of h of Open_Web_Steel_Joist135
Figure 6-10 Message received by structural desig made by architect	ner informing of design change
Figure 6-11 Message received by HVAC designe made by architect	r informing of design change
Figure 6-12 Message received by envelop design made by architect	
Figure 6-13 Message received by structural desig made by envelope designer	
Figure 6-14 Message received by structural desig made by HVAC designer	
Figure 6-15 Results of request made by design m made to a building component	anager to track design changes
Figure 6-16 Results of request made by design m messages that are sent because of o	anager to track automated certain design change140
Figure 6-17 Scenario for extended design change	s with several alternatives143
Figure 6-18 Bar chart and data for alternate change Figure 6-17	ge-path (1) for the scenario in
Figure 6-19 Bar chart and data for alternate change Figure 6-17	ge-path (2) for the scenario in144
Figure 6-20 Bar chart and data for alternate change Figure 6-17	ge-path (3) for the scenario in
Figure 6-21 Bar chart and data for alternate change Figure 6-17	ge-path (4) for the scenario in

Figure 6-22 Bar chart and data for alternate change-path (5) for the scenario in Figure 6-17146
Figure 6-23 Bar chart and data for alternate change-path (6) for the scenario in Figure 6-17146
Figure 6-24 Comparison among various alternate change-paths for the scenario in Figure 6-17147
Figure 6-25 Plan view of an exterior column and the surrounding components as designed (Fazio 1990 reproduced)
Figure 6-26 Plan view of the exterior column and the surrounding components as constructed (Fazio 1990 reproduced)
Figure 6-27 Prebuilt rule to capture link between construction tolerance and design of envelope and encasement
Figure 6-28 Dynamically built rule to capture link between width of the gypsum board encasement and fireproofing thickness
Figure 6-29 Automated message to inform architect about change in thickness of fireproofing
Figure A-1 Conceptual data-structure for management-database presented in EXPRESS-G
Figure A-2 Conceptual data-structure for general-database presented in EXPRESS-G177
Figure B-1 Result of applying steps Sub1-D till Sub1-H on example given in Figure 5-10
Figure B-2 Result of applying steps Sub1-I and Sub1-J on example given in Figure 5-10
Figure B-3 Matrix to represent start nodes and end nodes for change-path in Figure 5-18b
Figure B-4 Flowchart for sub-routine developed to generate various change-paths188
Figure B-5 Flow chart for sub-routine developed to define sequence of performing design changes so as to eliminate redundancy

Nomenclatures

- n Total number of design changes in a change path.
- x A design change.
- CT Expected impact of a change-path on the building construction costs.
- Cx Value of attribute "DIFFERENCE_IN_BUILD_COST" in Data-Table 5-7.
- DT Design cost related to a change-path.
- h Cost of man hour in the design firm.
- Mx Value of attribute "REQUIRED_MAN_HOURS" in Data-Table 5-7.
- RT Level of recommendation for a change-path.
- Rx Value of attribute "RECOMMENDATION_LEVEL" in Data-Table 5-7.
- TT Additional design time required for a change-path.
- Tx Value of attribute "DURATION" in Data-Table 5-7.

Abbreviations

AEC	Architecture/Engineering/Construction.
ΑI	Artificial Intelligence.
AACE	American Association of Cost Engineers.
ASCE	American Society of Civil Engineers.
CADD	Computer Aided Design and Drafting.
CIFE	Centre for Integrated Facility Engineering.
CMU	Concrete Masonary Units.
CPM	Critical Path Method.
DBMS	Data Base Management System.
DICE	Distributed and Integrated environment for Computer aided Engineering.
DIS	Draft International Standards.
DXF	Drawing interchange file.
EDM	Engineering Data Model.
HVAC	Heating, Ventilation, Air Conditioning.
IAI	Industrial Alliance for Interoperability.
IBDE	Integrated Building Design Environment.
IFC	Industry Foundation Class.
IGES	The Initial Graphics Exchange Specfications.
ISO	International Organization for Standardization.

MIT Massachussette Institute of Technology.

Information Technology.

MS MicroSoft.

IT

NIST National Institute of Standards and Technology.

OO Object-Oriented.

SEED Software Environment to support Early phase in building Design.

SQL Structured Querey Language.

STEP STandards for the Exchange of Product data.

Legend for User Interface Diagrams

\$57(viijo):	TEATHERNAUMOR.
Define tracking constraints	· New screen
Show building component under modification, its instance, and its modified attribute.	Note
SI	Decision node
Display related messages	Model output
Input the data of the design change	Link to a floating screen
Choose a project	Link to a screen that replaces the current screen
Choose a related attribute	User input in the current screen

1.1 MOTIVATION AND PROBLEM DEFINITION

1.1.1 BACKGROUND

Construction teams erect buildings on the basis of construction technical documents. Incompatibility errors in these documents result in change orders, contractual disputes, cost overruns, time delays, compromise to quality, frustration and client dissatisfaction. Common types of errors are:

- Inconsistency in design information. For example, the location of a specific column is not identical when comparing the architectural and the structural drawings.
- Mismatch between connected components. For example, HVAC ducts dimensions,
 which are given in the mechanical drawings, do not match the dimensions of the
 related pass-holes in the structural beams which are given in the structural drawings.
- Components malfunction. For example, electric supply in a room is designed to serve a classroom activity while architectural drawings indicate that the same room has been re-designed as a computer lab.

Unfortunately, such errors have become very common in the construction industry (Tilley and Barton 1997). Some contractors even depend on these errors for generating their

profit margin. Owners, on the other hand, are either discouraged to invest in buildings or try to compensate the effect of these problems in the pricing of services or products realized within the constructed facility. This may negatively affect not only the building industry, but also eventually the national economy.

Construction technical documents are produced by the design team during the detailed design stage of a project life cycle. These documents are the result of collective efforts from specialists who belong to various design disciplines. These specialists, who are usually geographically separated, make autonomous design decisions with respect to their own discipline. These decisions, nevertheless, are inter-dependent and need to be coordinated so as to maintain compatibility among the various systems and components in the building under design. The coordination process which occurs during the detailed design stage is very complicated. This is due to the generation of an enormous amount of design data from the various specialized disciplines. Unfortunately, with escalating complexity in buildings, the increasing specialization in design teams, and the tightening of financial and time resources available to designers, the coordination task is rapidly increasing in complexity. The building industry is in urgent need for research work that addresses the problem and contributes to finding practical solutions. This research is a step towards satisfying that need.

1.1.2 OVERVIEW OF THE DETAILED DESIGN STAGE

During the detailed design stage, abstract concepts presented in the preliminary design of

a building are transferred into technical documents. These documents should contain sufficient design information to enable the erection of the building. The production process of these documents, which appear in the form of working drawings and specifications, is an elaborate one. It requires members of the design team to satisfy many constraints such as owner demands, code requirements and limitations of the project budget. In addition, it requires from them to ensure the compatibility of design information presented in the technical documents. This research is primarily concerned with the common failure of designers to satisfy this last demand. An analytical overview of the production process of construction technical documents is therefore presented. The overview concentrates on current techniques for cross-disciplinary coordination of design information, hence revealing the main problems that cause the process to fail.

Figure 1-1 illustrates the results of the analysis. The figure shows that the production process of technical documents starts by extracting the data available in preliminary design documents. From these documents, the architecture design team develops the main features of the building in the form of drawings (Point 1, Figure 1-1). These main features usually include geometric details of the structural elements, walls, partitions, doors, windows, stairs, elevators, and the function of the various spaces. Several copies are made of these drawings and distributed to other disciplines in the design team (Point 2, Figure 1-1).

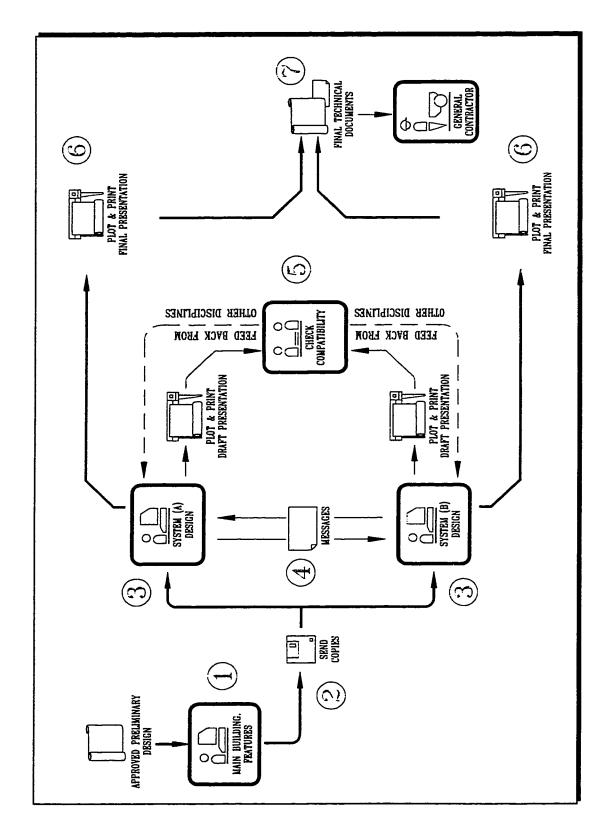


Figure 1-1 Production process of construction technical documents during detailed design stage.

Next, each discipline (including architecture) starts to make design decisions for the detailing of a specific building system (Points 3, Figure 1-1). These decisions aim to satisfy various design requirements and are restricted by codes and regulations. More important to our study, these decisions depend also on design decisions that are made by other disciplines. For example, to design an HVAC component in a certain room, designer "A" searches for the data needed as input to the HVAC design process, e.g. space function, dimensions, heat produced from lighting fixtures, and thermal resistance of external walls. Such input data is actually output data of other disciplines. Both the space function and dimensions are data that is defined by the architect; the lighting fixtures are defined by the illumination engineer; the thermal resistance of the external walls is defined by the envelope designer. Designer A tries to find this data in the drawings that are provided by the architect. Most probably, only the data for the space function and the space dimensions are available in these drawings. To get the rest, designer A needs to communicate with the illumination engineer and the envelope designer (Point 4, Figure 1-1). Once all the input data is obtained, designer A can make design decisions on the HVAC components of the room. As long as all elements of input data remain unchanged, the HVAC component will remain compatible with the other components in the room.

Unfortunately, design changes - inevitably and continuously - occur for many reasons.

These include the need to satisfy new or modified requirements by the owner, to reduce construction or maintenance costs, to rectify design mistakes, or to improve the design.

For example, the architect may change the function of the space from an office to a computer lab. Such a change is usually performed by the architecture discipline through renaming the function of the space on the floor plan drawing. Once that "simple" change is performed, the HVAC component - designed by A - instantly becomes incompatible with the architectural design. It remains as such till designer A becomes aware of the design change and modifies the HVAC component using modified input data.

The main problem therefore is how to make designers of any discipline aware of all changes that are performed by other disciplines and that affect their own design. This is a difficult problem to solve because of the lack, in our judgment, of the knowledge that "links" the output of one design discipline to the input of another design discipline. This lack of "linking" knowledge is primarily due to the education and training of professionals in the building industry. It is difficult, for example, for an architect to know the input data for an HVAC engineer. The architect might not know that space function is a critical input for the HVAC system design. Consequently, s/he may change the space function without informing anyone because it seems unnecessary to do so. As a result, the HVAC engineer has no way to recognize that a design change affecting HVAC has been made.

From this simple example, one can foresee the difficulty which design teams face when dealing with a project that has thousands of design data that are mostly interrelated. Not surprisingly, some professionals describe design changes in a multi-disciplinary environment as a "mess". Nonetheless, techniques are used by professionals in order to

cope with the problem (Point 5, Figure 1-1). Some of these techniques are:

- a) Regular coordination meetings where designers from various disciplines group together for the purpose of finding out about the important changes that each discipline has done. Each discipline usually reviews the drawings of the other disciplines to look for modified data. Drawings from different disciplines are sometimes put together for comparison. Recently, with development in computer technology, such meetings become less frequent. Reference drawing files are increasingly used to show the latest drawing version of various disciplines instantly. However, this technique of reviewing the documents of other disciplines has a serious deficiency. It depends on designers to discover the changes that are relevant to their own design, mainly through looking at drawings. As a result, many design changes which do not clearly appear on drawings (e.g. thermal resistance of a wall or a small change in a dimension) are usually missed. Since large projects may include hundreds of drawings, it is impractical to review such a large amount of data regularly.
- b) The designer who makes a change informs every other designer in the design team.

 This technique may work fine in small projects, yet in larger or more complex projects, it becomes impractical. Every designer has to issue too many design change memos and will also receive as many. Most of these memos are irrelevant to many designers and soon lose credibility. Designers become discouraged to spend their time writing and reviewing them.

components in a project. The main problem with this technique is that the design modifications that are required when incompatibility errors are discovered may generate in turn other incompatibilities. The technique cannot continuously monitor the effect of design changes. In addition, the check lists are rarely customized so as to adapt to every building project. Many incompatibilities are therefore skipped by this technique.

The production process of construction technical documents (points 3, 4, and 5 in Figure 1-1) is cyclically repeated till deadline for submission is reached. Each discipline then finalizes its own documents (Point 6, Figure 1-1) and all final documents are gathered from the various disciplines and given to the general contractor (Point 7, Figure 1-1).

1.1.3 DIAGNOSIS OF THE MAIN PROBLEMS

From the above analysis of the process, three major problems can be diagnosed as the causes for incompatibility errors.

1.1.3.1 Managing Design Changes

Design changes continuously occur within all disciplines. The current coordination techniques provide weak mechanisms to manage design changes and to accommodate their effect on the design information across participating disciplines. More specifically, there is a lack in the ability of designers for propagating and communicating design changes effectively. Others also support this finding. Hegazy and Khalifa (1996)

conducted a survey among 12 leading design firms in Canada and showed that disseminating project information and administering design changes are two important aspects of multi-disciplinary coordination which cause problems in building design. Cornick (1990) states also that "the problems caused in modern buildings seem much more due to deficiencies in managing communication during the design process than to merely technological factors".

1.1.3.2 Communication Media

Drawings are used as the principal media to exchange and document design information. Drawings are excellent for describing shape, geometry, proportions and other visual characteristics of the design. Yet, drawings are very poor to show other characteristics such as heat resistance, load capacity, and price. Such non-visual characteristics are as essential as the visual ones for interdisciplinary exchange of information. Problems with the drawing-centered environment are also reported by others (Voeller 1996).

1.1.3.3 Source of Final Documents

The final construction technical documents are collected from a variety of sources. This can lead to inconsistency in information when the same building element is described in different ways in several documents. For example, the location of a specific column is not identical when comparing the architectural and the structural drawings

Recently, CADD software developers have tackled the second and the third problems.

Software such as Bentley Microstation TM, ArchT TM and Pro-Reflex TM provide a data-

centric environment and central single model for the designed building. Technical drawings are then generated from such models, ensuring consistency among various drawings. Nevertheless, other types of incompatibilities (as given in section 1.1.1 on page 1) continue to exist. These incompatibilities are attributed to the failure to address the first diagnosed problem: managing design changes.

1.1.4 CONCLUSION

The analytical overview of the detailed design stage shows that a main challenge in maintaining compatible design information resides in the accommodation of design changes. More specifically, propagation of design changes to the affected disciplines constitutes the critical point. The various techniques and software that are currently used by designers to coordinate their design information appear inadequate. There is a clear need for an approach that targets this important problem.

1.2 RESEARCH SCOPE AND OBJECTIVES

The main objective of this research is to assist the building industry in eliminating incompatibility errors that commonly exist in design information of building projects, a pervasive problem that is very costly to the industry. The research focuses on the design changes during the detailed design stage as these changes remain the main cause of incompatibility errors.

The research aims to develop a computer-based model that uses Information Technology

to provide the building industry with a practical solution to the problem. The developed model however is not limited to the propagation of current design changes, rather it is extended to enable the tracking of past design changes and the planning and scheduling of future ones.

The research also aims to implement the model in a client-server computing environment which is similar to that regularly used by building design professionals. Furthermore, it aims to validate not only that the model is able to provide capability to manage design changes but also that the model can actually improve current practices with these capabilities.

The research however does not intend to provide a fully functional information system which should include complete software, hardware specifications, operating procedures, training process for operators, and full collection of the needed data.

The information model developed in this research does not intend to cover all types of information necessary for the design of a building. Codes, regulations, and design constraints are among the design information that are excluded. The model focuses on information needed for design changes. The model is also intended for sequential type of project delivery systems. It is not intended for other types of delivery systems such as fast track system. Designers who will use the model are expected to collaborate using client-server networks and therefore they should be located in the same building or nearby buildings. Other types of networks such as wide area networks, intranets, or internet are

beyond the scope of this model.

1.3 OUTLINE OF THE THESIS

This thesis documents the development of an information system model for managing design changes in a collaborative multi-disciplinary building design environment. Chapter one presents the motivation for this research and defines the specific problem. Chapter two reviews research elsewhere in the use of information technology for the design of constructed facilities. Chapter three introduces a conceptual perspective for the proposed information model and discusses the advantages and limitations of this model. Chapters four and five present the detailed development of the main components of the model and show how the model concepts are realized with these components. Chapter four focuses on the central project-database and how it is developed to adapt easily to various building configurations. Chapter five provides the detailed methodologies that make the model capable of managing design changes. Chapter six explains the validation process for the model. Chapter seven summarizes the research and highlights its contributions to the construction industry. It also provides recommendations for future research work.

INFORMATION TECHNOLOGY IN DESIGN: A LITERATURE REVIEW

2.1 INTRODUCTION

In chapter one, three major problems are diagnosed as the causes for incompatibility errors in construction technical documents. These problems are: managing design changes, communication media, and source of final documents (see section 1.1.3 on page 8). As a first step to find an appropriate solution that overcomes these three problems, this chapter reviews available research work that is concerned with the use of information technology in the design of constructed facilities. This area of research is relatively new and increasingly captures the attention of researchers in many parts of the world. This attention can be attributed to the growing familiarity of the construction industry with CADD (Computer Aided Design and Drafting) systems. It can be also attributed to the potential of CADD systems to address many problems in such a scattered, multi-disciplinary, and information-rich industry. This review is organized into three sections:

- The first section is concerned with approaches that use computer technology to enhance collaborative design environments. Each of the reviewed approaches is evaluated against its ability to provide solution to the previously diagnosed problems.
- The second section focuses on efforts to establish data-structures (data models) that
 can support design data in a central database. Such a central database is considered to
 be necessary for ensuring a single source of design information. The literature review

in this group aims to investigate the availability of standards or at least sufficiently advanced models that can be utilized in this research to develop a central database for building data.

 The third section surveys research projects that deal with managing the building design process, an area to which this research belongs, so as to review utilized concepts and techniques.

2.2 COLLABORATIVE DESIGN APPROACHES

The use of computer technology to enhance collaborative design environments is the focus of several research works. The following summarizes the main approaches found in literature, along with discussions regarding their ability to solve the three main problems diagnosed in section 1.1.3.

2.2.1 AGENT-BASED FEDERATED ARCHITECTURE APPROACH

Developed at Stanford University, it connects two or more independent engineering software that need to share information about the current status of a design (Khedro et al. 1994, 1995 and 1996, McGraw et al. 1996, Jones and Riley 1995, Khedro 1994, Huyn et al. 1993, Cutkosky et al. 1993). The system architecture includes several design agents (Figure 2-1). Each design agent is composed of a human designer and a software agent (design software plus communication capabilities). Every software agent interacts with other software agents through facilitators that coordinate the exchange of design information among the software agents. When a designer finishes a design task, s/he

sends the design information to the facilitator that s/he is connected to. The facilitator is able to recognize other interested users in each element of that design. It groups the design elements and forwards them to the relevant users. These users receive the information, check it, evaluate its effect on their design, and start negotiation with each other in real time to solve conflicts in design. In view of the diagnosed problems in section 1.1.3, this approach has a fundamental drawback which is the lack of a single source of final design information. However, when it comes to managing design changes, the approach has the advantage of reducing the amount of information received by every user as only the relevant information is sent to the various users. This is an important feature that has been incorporated in the information model proposed in this thesis. The federated architecture approach nevertheless depends on the designers to recognize design changes. This leaves a lot of room for error and is similar to current practice (section 1.1.2). The approach also requires a significant deployment of computing resources and communication links across separate locations, a requirement that would set such an approach beyond the reach of most building designers.

2.2.2 KNOWLEDGE-BASED APPROACH

The Knowledge-based approach has been used to help integrate software that are used in engineering design. Fenves et al. (1990, 1994) introduced a prototype system called the Integrated Building Design Environment (IBDE) at Carnegie Mellon University. The system intends to vertically integrate the various software that are used in the various stages of the life cycle of a constructed facility. Its demonstration and test domain,

however, is limited to architectural planning, structural design and construction planning of high-rise office buildings. All the involved software use knowledge-bases that support their functions. Project data is maintained in a project database whereas communication between the different software relies on a message blackboard.

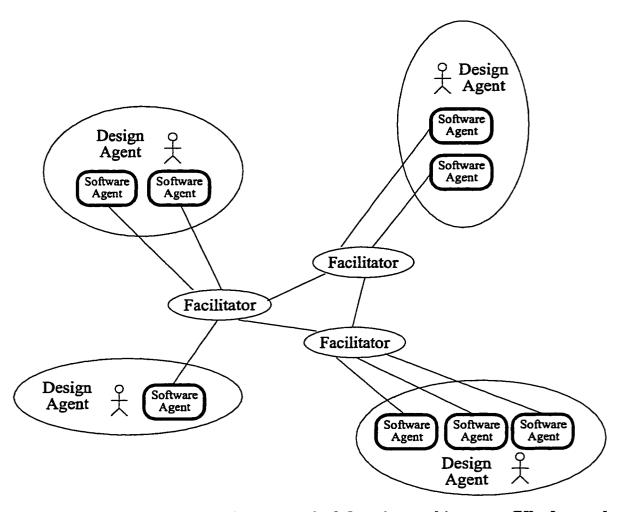


Figure 2-1 Organization of design agents in federation architecture. (Khedro et al. 1994, redrawn)

Another system called DICE (Distributed and Integrated environment for Computer aided Engineering) has been under development at the MIT (Massachussette Institute of Technology) (Sriram et al. 1989, 1990, and 1994, Sriram and Logcher 1993, Ahmed et al.

1992, Wong and Sriram 1993). The focus of the project is on preliminary and detailed structural design. DICE consists of a blackboard, knowledge modules, and control mechanism. DICE blackboard is implemented on a central object-oriented database. This central database does not contain all the product data but is limited to the data used for communication among the knowledge modules.

A more recent model is SEED (Software Environment to support Early phase in building Design) (Flemming and Woodbury 1995) also from Carnegie Mellon University. SEED aims at providing computational support for the early design phase. It addresses specifically architectural programming, schematic layout design and the generation of a fully three dimensional configuration of physical building components like structure and enclosure. SEED contains an object database to store and retrieve different design versions and design alternatives. Designers can specify and modify dynamically and interactively design requirements. Given explicit requirements, SEED can be asked not only to propagate design changes after some interactive modifications by the designer, but also to show other designers feasible ways of solving a design problem using case-based reasoning.

In the knowledge-based approach, collaboration among software is achieved by the use of precoded coordination knowledge. The coordination knowledge is based on the various software meta-knowledge (knowledge about the composition and the computational process of the software) and hence it needs to be updated to reflect any upgrading in the involved software. In order to manage design changes, the knowledge needs to

accommodate various possible relations that may exist among different building components, a requirement that is almost impossible to fulfill due to the unique nature of every building project. Such a nature makes these relationships vary from one project to the other. When this is combined with different composition of design teams in different projects, it seems that this approach is not suitable for the complex environment of building design.

2.2.3 CONSTRAINTS MANAGEMENT APPROACH

This approach has been proposed as a method to share design knowledge without linking together design software (El-Bibany and Paulson 1994, El-Bibany 1992 and 1996, Bowen and Bahler 1993). A constraint is taken as a relation which states what should be true about one or more design entities. Managing these constraints is achieved by creating and maintaining data-dependency structures that reflect any changes in the design knowledge and identify specific knowledge items that cause conflicts. The users have the ability to create relations dynamically between entities. Yet, the entities themselves are fixed in the system. We view the ability of this approach to dynamically create relationships between entities as essential to adhere to the fact that every building project is unique in these relationships. However, it is also important to be able to easily change the building entities to reflect the requirements of a specific project. In case of a design change that violates any of the constraints which were previously built, the system sends warning messages to involved parties informing of source of violation. This is also an important feature that is used by the model proposed in this thesis for managing design changes.

The constraints management approach however requires each discipline to provide all necessary equations and functions that define and constrain the design of every building component in order to built the data dependency structures. This requirement results in duplicating most of the equations and the constraints that are already used by the design software. The process as such becomes very complicated when considering the number of disciplines and the number of components involved in the design of a building.

2.2.4 CIRCLE APPROACH

Aiming to integrate some AEC (Architecture/Engineering/Construction) software that are used by participants in a project, the circle approach (Fischer and Kunz 1995) has each software linked to exactly one predecessor and one successor software. Thus, changes made to the design are propagated automatically to all other software. It is expected that participants will have private copies of all software on the circle for a particular project. The circle approach does not use any form of centralized repository for design data. As such, it cannot satisfy the need for having a single source for the final design documents which can insure the consistency of building information. It propagates new versions of design, but depends on users to recognize any change in the design that might affect them which is similar to current practice.

2.2.5 DISCUSSION

The approaches introduced above for the use of computers in collaborative design environment are not mutually exclusive but rather, share common features. All

approaches use data as the means to exchange information among participants. This supports the notion that drawings should not be - in the computer age - the media for recording design information. Both the agent-based approach and the circle approach do not use any form of centralized repository for design data, nor for data coordination knowledge. As such, these approaches cannot insure the consistency of building information. Both the agent-based approach and the circle approach propagate new versions of design, but depend on users to recognize any change in the design that might affect them. The knowledge-based approach coordinates design information using detailed coordination knowledge linked with the used software. As discussed above, this approach seems impractical for the complexity of building design environment. The constraints management approach requires duplicating most of the equations and the constraints that are already used by the design software, which makes its use time-consuming and redundant.

None of the reported approaches appears suitable to provide remedy for all the problems diagnosed in section 1.1.3. However, some reported capabilities are useful to this research such as the capability of sending information to the interested designers only and the dynamic capturing of coordination knowledge from designers instead of exclusively using precoded knowledge. The information model proposed in this thesis include these capabilities.

2.3 PRODUCT DATA MODELING

Product data modeling may also be viewed as one of the approaches to facilitate design collaboration. Because of the wide interest by researchers to develop product data models and the need to have a central database to provide a single source for design information, product data modeling is reviewed in more detail. The purpose is to investigate the availability of standards or at least sufficiently advanced models that can be utilized by this research in developing a central database for building data. The review starts by defining product data models and overviews its successful implementation in some research areas. A description of main product data models follows in order to explore the existence of any common or necessary features for developing these models. Finally, discussion and highlight of finding is presented.

2.3.1 ON DEFINITIONS AND IMPLEMENTATION

A model is a simplified representation of a part of the real world. It captures some, but not all, of the characteristics of that part (Lave and March 1975). For example, an architectural perspective of a building, which can be referred to as a graphical model, represents visual relationships between the various elements of the building envelope. Meanwhile, a group of equations represents the transfer of heat through that envelope and is called a mathematical model. One type of model that has emerged with the development of computers is Data Models which describe a part of the reality in form of its data elements. Data models are divided into three types (Elmasri and Navathe 1994): conceptual models, which tell what kind of information is used to describe some aspect of

reality and how such information is internally structured (Björk 1989); implementation models, which are translations of conceptual models for a specific database management system; physical models, which represent how the data is placed on a computer hardware system.

Product Data Models are a distinct category of the conceptual type of data models. These are tailored to represent data of engineering products. There is no single definition that is agreed upon for a product data model. The COMBINE project (Augenbroe 1993) defines it as "a complete conceptual description of a product, capable of structuring all the information necessary for the design, manufacture, and use of the product." Björk and Penttilä (1989) define it as a "conceptual structure specifying what kind of information is used to describe a building (product) and how such information is structured." Tolman, Kuiper, and Luiten (1990) define it as an "information model of a product, describing the 'reality' of a product in its different life cycle stages." Eastman (1992a) defines the product data model as "the database model supporting the design, fabrication, operation and other uses of some type of product." From these definitions, common denominators can be extracted: product data models are conceptual models; these represent all the product data; this representation spans throughout the product life cycle. This conclusion can be confirmed by the International Organization for Standardization (ISO) definition of product data (ISO 1989) as "The totality of data elements required to completely define a product; this includes geometry, topology, relationship, tolerance, attributes, and features necessary to completely define a component, part or an assembly of parts for the

purpose of design, analysis, manufacture, test and inspection." Product data models are, therefore, perceived to be good tools for both the exchange and integration of product information.

Because of the potential of product data models to facilitate the integration and exchange of product information, researchers in various parts of the world have adopted them to represent the Building as a product. The AEC Building Systems Model in the USA (Turner 1990), the RATAS model in Finland (Björk 1989), and the COMBINE IDM model in Europe (Dubois and Parand 1993) are examples of that utilization.

Other research work has been conducted so as to create an environment and a set of concepts that link building data models to various aspects of the building delivery process. The available research was in areas like:

- Early design cost control (Tsou 1992),
- Computer-aided architectural design (Turner 1992),
- Conformance with regulations (DeWaard 1992),
- Evaluating building performance (Augenbroe 1992),
- Maintenance information management (Svensson 1993),
- Renovation design (Vahala 1995),
- Design of building envelope (Rivard et al. 1995),
- Design of precast concrete facades (Karhu 1997), and
- Structural synthesis and evaluation (Bakkeren and Tolman 1995, Enseleit et al.

1995).

Another notable effort in using database to record building design data is through the International Alliance of Interoperability (IAI). The IAI is an alliance of groups in the building industry that aims to integrate the AEC/FM (Architecture, Engineering, Construction / Facility Management) industry by specifying Industry Foundation Classes (IFCs) as a universal language to improve the communication, productivity, delivery time, cost, and quality throughout the design, construction, operation and maintenance life cycle of buildings (IAI 1996). IFCs constitute a library of commonly defined objects that create "intelligent" project data such as the properties, behavior, and graphical representation of building components (Herold 1997).

Similar effort is done by ISO which is in the process of establishing a universal standard for the representation and exchange of product data (ISO 1989). This standard is usually referred to as STEP (STandards for the Exchange of Product data), formally ISO 10303 (Froese 1996). Its objective is to provide a mechanism that is capable of describing product data throughout the life of a product independent of any particular computer system. A comparison among current standards to communicate data [e.g. DXF (1995) and IGES (Reed et al. 1990)] and those suggested by IAI and STEP is presented by Arnold and Teicholz (1996).

2.3.2 Analysis of Some Building Product Data Models

The following is a review of some of the major building product data models developed

in research. The purpose is to understand the various concepts and approaches that categorize and organize the data that exists in a building.

2.3.2.1 The AEC Building Reference Model

This is a high level conceptual product data model for AEC products that concentrates on Buildings (Turner 1990). It defines a building project as a unique object made of a building and a site. A building is an object with one or more properties such as type, primary activity, and secondary activity. It decomposes into a number of building systems. Building system examples are structural, electrical, circulation, plumbing, heating, and lighting. A site is also an object with one or more properties such as humidity ratio, temperature, and view. It decomposes into site systems which can be electrical, fiesh water and disposal, electric, and gas.

The AEC Building Reference Model adopts a hierarchy of four levels: system, system component, system component port, system component port joint. A system exists to satisfy a human or natural need; an example is the building structural system. A system is a collection of system components. A system component such as a wall may belong to various systems, such as the enclosure system, the acoustical system, and the structural system. The wall component has different functions in each of these systems. System component ports are used to connect components in the same systems or in different systems. A door is a port that connects a room component to a corridor component. Each system component port is joined with another system component port by a system component port joint. The AEC model provides description for only the spatial system,

the enclosure system, and the structural system of a building.

2.3.2.2 The COMBINE IDM Model

COMBINE IDM stands for "COmputer Models for the Building INdustry in Europe - Integrated Data Modeling" (Dubois 1992, Dubois and Parand 1993). It is part of the COMBINE project which has a primary focus on the energy aspect of buildings. The root object in this model is the Construction Project which may contain one or more Buildings. The Building is seen as an assembly of systems. The systems are: spatial system, fabric system, technical system, functional system, and external environment system. The overall space is divided into a set of zones, and each zone has some homogeneous behavior. A zone is created from the specific view of an expert. Due to the focus of the model (energy analysis), the fabric system main element is the building enclosure, and the HVAC (Heating, Ventilation, Air Conditioning) system is the only one available as a technical system.

2.3.2.3 The RATAS Model

RATAS is the basic model for the building products in Finland. It stands for "Computer Aided Design of Buildings" in the Finish language (Björk 1989). It aims to cover the design, production and maintenance stages of the building project. Concepts such as objects, attributes, relationships are used in the model. Objects are collections of data about a "thing", whether physical or abstract. To each object, a number of attributes can be associated that describe the properties of that object. Attribute types in RATAS are:

numeric, text, pictures, codes, and lists which make it capable of containing all kinds of data describing a building. Five functional levels for objects are distinguished in RATAS: building, system, subsystem, part, and detail. There is only one building level object per building. Its attributes are about the site, climate, type of building, construction cost, total size etc. Systems can be spaces, load-bearing, mechanical, electrical, or heating. Subsystems (or Groups) divide systems into functional parts such as floor or hospital ward. Parts constitute the vast majority of objects and are usually tangible entities such as building elements or technical devices. Their typical attributes are location and shape. Detail level includes data of objects that are subdivisions of parts like the different parts of a window. Usually these data will be included in general access databases. The model uses two types of relations: 'Part of and 'Connected to'. The 'Part of relation links objects from different levels while the 'Connected to' is more typical in part and detail levels, and usually connects objects at the same level.

2.3.2.4 Other Models

Another product data model is the Construction Project Reference Model (Rezgi and Depras 1995) where the building is described according to five complementary systems: structure, work, space, technical and separation systems. The purpose of this model is to enable the generation of construction documents from a single source of information. A multi-dimensional model of buildings (3P Model) which integrates the process, product, and participants is proposed by Bédard and Rivard (1995). It classifies buildings into structure, envelope, services, and interior systems.

2.3.3 DISCUSSION

The review of the available models shows that classification of building data varies considerably from one model to another. It depends mainly on the purpose of each model. The common feature in the models is the division of a building into group of systems. However, these divisions vary from one model to the other. Each model concentrates on a specific stage on the building delivery process and details a limited number of systems to fit its purposes. Therefore, no standards are currently available to follow nor is there any model that is sufficiently developed to be used in this research. Large groups of researchers have developed some of the models, yet the models still have limited capabilities. This shows the extreme difficulty to represent buildings as data models due to the unique nature of building projects. Therefore, we believe that any data model that represents building projects mainly needs to be flexible and oriented to be suitable for the project at hand. The use of IFCs might achieve this need, however, little information is publicly available now in order to consider their use in this research.

2.4 DESIGN PROCESS MANAGEMENT

Research in the use of information technology for managing the design of a building project, which this research belongs to, is still uncommon. However with growing interest in both the use of information technology in collaborative design and the use of product data modeling in the construction industry, we believe that interest in using information system for managing design will also grow. The following reviews available research in three issues that are important to the design process management.

2.4.1 DESIGN CHANGES

Design changes, which are the main issue addressed by this research, has not yet received much attention in the literature. A project by Krishnamurthy and Law (1995a, 1995b, and 1996 and Krishnamurthy 1996) proposes a three-layered model of versions, assemblies, and configuration. A version is a description of a primitive entity, an assembly describes a composite entity that belongs to a single discipline, while a configuration describes the overall project. The model monitors independent design activities by systematically tracking components descriptions in individual disciplines. It uses the concept of equivalent operations to compute changes. The advantages of this approach is its ability to compute changes between different versions of the design. Yet, it remains for the designer who performs the changes to know whether or not others are affected by the change. Also, designers themselves need to recognize any inconsistencies by reading drawings of the latest design version.

The EDM (Engineering Data Model) which has been developed by Eastman intends to support integrity management among various intelligent design software packages that are connected to the model (Eastman et al. 1995, Eastman 1992b, 1994). The model includes constraints that check the status of rules and specific knowledge embedded in the software. When a design change occurs to an instance, all design instances that use that data of the changed instance are flagged for rechecking. However, no information is given to the user on what data element has changed and why a data element is flagged for rechecking.

2.4.2 **DESIGN INFORMATION**

Management of design information was also the focus of some research work. Platt (1996) focused on civil engineering projects and tried to manage their design information through process modeling rather than product modeling. Process modeling has information on how a product is transformed from its initial state to the final deliverables. Hence, the project information recorded on computer contains not only data about the project component but also about the process used to record this data. Rezgui has developed an approach to tackle the problem of integrity and consistency in the production of construction documents (Rezgi and Depras 1995). It is based on building a construction project reference model and a document reference model. Both models are linked with an association model that indexes building components to documentary items. The research focuses on how to extract construction documents from a project reference model and tries to ensure the integrity of these documents. It does not focus on how to ensure the integrity of the design information itself. The International Organization for Standardization (ISO) is also in the process of establishing a standard layering structure to be used in building design to facilitate the transfer and management of information (Björk et al. 1997).

Exchanging design information among designers has been investigated by Vries and Somers (1995) who developed a process model for that purpose. They suggested that a protocol to exchange information should ensure that both senders and receivers of information should interpret it the same way. Also all the required and correct

in the information model proposed in this thesis. The use of the internet to exchange and manage design information has been also investigated. Guenster (1996) discussed the benefits, costs, and lessons from using the internet with a design workgroup. Goodman and Chinowsky (1996) discussed the requirements for managing design information through an inter-disciplinary team when the internet is utilized. Rojas (1997) described a process for developing a web-centric system that enables cooperative engineering.

2.4.3 DESIGN RATIONALE

Several research projects have addressed the issue of acquiring the rationale behind a design decision. Peña-Mora and others (1995) studied the representation, use and communication of design rationale for conflict mitigation. They developed a model that is capable of representing design knowledge in term of the reasoning process used by the designer. The model also provides computer support for capturing designer's reasoning process. This capability appears useful to easily extract required knowledge from designers. The proposed information model in this thesis therefore includes similar capabilities. De la Garza and Alcantara (1997) used parametric dependency networks to represent design rationale. These networks can show how one particular design decision affects other decisions. The use of this type of network requires significant effort and training from the design team. In general, the aim of research work in capturing design rationale is to define the motives for a design decision. The acquired knowledge is different from that needed to propagate design changes. The latter requires knowledge

that defines the consequences of a design decision.

2.5 SUMMARY OF FINDINGS

From the literature review, it is evident that the building industry is still at the exploration stage on how to use information technology in design and design management. Several approaches have been tested for collaboration in a multi-disciplinary environment. Each approach is oriented to solve a specific problem. None of them is capable to offer a solution for all problems diagnosed in section 1.1.3. Building product data models are successfully used to solve several research problems. However, different researchers present very different compositions for these data model. No standard is available nor is there sufficiently developed model that can be used. A flexible data model that can be configured for a specific building project appears to be the practical answer. Literature review on managing design process reveals that few research work studied how to deal with design changes. As shown in section 2.4.1, these researches study the problem from a viewpoint which differs from this thesis. Krishnamurthy and Law (1995a, 1995b, and 1996 and Krishnamurthy 1996) focus on comparing various design versions and define embedded differences whereas Eastman (1992b, 1994, 1995) is concerned with how to propagate design changes among intelligent design software. Throughout the literature review, some capabilities appeared useful and will be incorporated in developing the proposed information model. These capabilities are:

• Dynamic capturing of knowledge from designers during the design development.

Such capturing capability should be supported by appropriate computer interface.

- Acquiring and storing some information about the design process itself and not limiting the information used in the model to that describing the building components.
- Propagated design information needs to reach only the interested designers not all the designers.
- Exchange of design information has to be performed in a standard fashion that can be understood clearly.

The following chapters present the proposed information model in both conceptual and detailed perspectives.

CHAPTER THREE

PROPOSED INFORMATION MODEL: A CONCEPTUAL PERSPECTIVE

3.1 OBJECTIVES AND MODEL CHARACTERISTICS

The analysis for the production process of construction technical documents diagnosed three main problems that result in the inclusion of incompatibility errors. The analysis also revealed that design changes remain as the principal challenge. The information model developed in this research aims to overcome this challenge. The model also needs to provide solution to the other diagnosed problems (see section 1.1.3 on page 8). The information model therefore needs to have the following characteristics:

- It is capable of managing design changes. This capability is seen not to be limited
 to help propagation of design changes but it is also expected to cover the need for
 tracking past design changes and for providing the capacity to plan and schedule
 future design changes.
- 2. It uses data rather than drawings as the main media for storing and communicating design information.
- 3. It has a single source for design information that carries a unique description for every component in a building project.

Through the following sections, the architecture of the model is described in a conceptual perspective so as to reveal the main ideas rapidly (see also Mokhtar, Bédard, and Fazio

1998). The description is supported by simple examples that demonstrate the capabilities of the model. A detailed perspective is presented in the next two chapters.

3.2 THE MODEL ARCHITECTURE

3.2.1 BASIC PARTS

The heart of the model, as shown in Figure 3-1, is a central *Project-Database*. Unlike other models, this project-database is uniquely composed of two parts, the *Building-Components-Database* and the *Management-Database*. The building-components-database functions as a repository of all the design data that is necessary to describe every building component in a certain project. Such data is detailed enough to produce construction technical documents that are suitable for erecting the building. The management-database contains data that is required to manage the various functions of the information model. The role of this database will be incrementally revealed throughout the rest of the thesis. To generate the data that populates the building-components-database, each *Designer* is linked to the central project-database by a group of modules called *Designer System*. The *Design Manager*, who is responsible for managing the production of the technical documents, is also linked to the central project-database through a group of modules called *Design Manager System*.

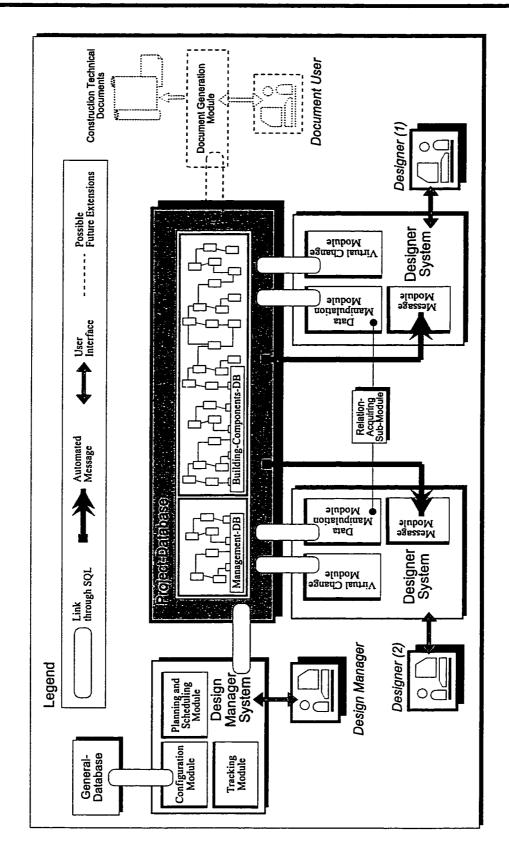


Figure 3-1 Architecture of proposed information model.

When the building components database is fully populated, construction technical documents can be produced with the use of the *Document Generation Module* which can create customizable documents about the project through its link to the project database (Mokhtar and Bédard 1994). This module, shown in dotted lines in Figure 3-1, resides beyond the scope of this thesis.

The detailed design stage starts with the architectural team translating ideas in preliminary design documents into elementary building components such as walls, columns, doors, and windows. Most of the data that describes these components is textual in nature. Therefore, its value can be keyed in the building-components-database. Some of the data is geometric in nature, such as the dimensions, and is usually documented through a computer-aided drafting tool. The function of that tool is to acquire, through its interface, the geometric data and to translate it into alphanumeric characters. The model currently does not include such a tool and the designers therefore are expected to directly key in the geometric data directly in textual format.

Designers interact with design data in the database in order to view, modify, or add values. This interaction is materialized through the *Data Manipulation Module*. As Figure 3-2 illustrates, a designer can - through this module - create a new instance or manipulate the value of any attribute of any instance of any building component. The type of manipulation (view and/or modify) should depend on the authority that a designer has over a specific building component. The data saved in the building-components-database reflects decisions taken by the design team members.

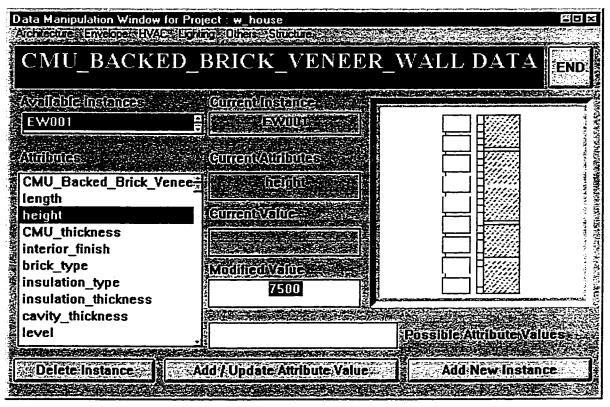


Figure 3-2 Interface for data manipulation module¹.

Yet, to enable the designers to record data about the various building components of a certain project, the building-components-database needs to have a suitable "data-structure" (or "product data model"). This data-structure should echo the type of building components that exist in the project. Because buildings are unique products, every building project contains different combination of building components. Hence, every building project needs a building-components-database with a unique data-structure. The building-components-database therefore should be flexible so as to adapt to the requirement of every project. Such flexibility is realized in the model through the use of the management-database. More details are given in chapter four and also in Mokhtar,

¹ All linear measurement are in mm units.

Bédard and Fazio (1997b). The data-structure of the building-components-database can be manipulated through the *Configuration Module* which is part of the design manager system. Figure 3-3 illustrates part of that module where an attribute (vapor barrier type) is added to the building component "CMU_backed_brick_veneer_wall" as part of creating that component in the building-components-database. Such manipulation, which is carried out by the design manager, is facilitated through the use of a *General-Database*. This general-database functions as a source of data-structure for a large variety of building components. Figure 3-4 illustrates the use of the general-database to retrieve the data-structure of the building components that belong to an HVAC system into the building-components-database. The configuration of the data-structure for the building-components-database needs to be performed before beginning the development of the detailed design. Yet, it can also be updated during that development.

3.2.2 THE CONCEPT OF ACTIVE BUILDING COMPONENTS

As recognized before from analyzing the production process of construction technical document, design changes constitute a major source of incompatibility errors. The crux of the design change problem is the failure on the design team's part to propagate design changes. As found from the analysis of the detailed design (section 1.1.2), this failure is due to the lack of the "linking" knowledge that is necessary to perform this task. Also, it is due to the large amount of data that is generated and needs to be monitored and communicated during the development of detailed design.

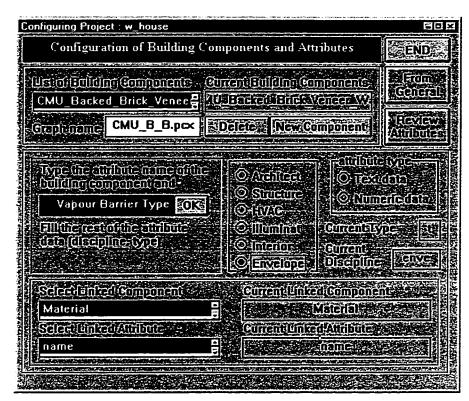


Figure 3-3 Configuration of a building component using configuration module.

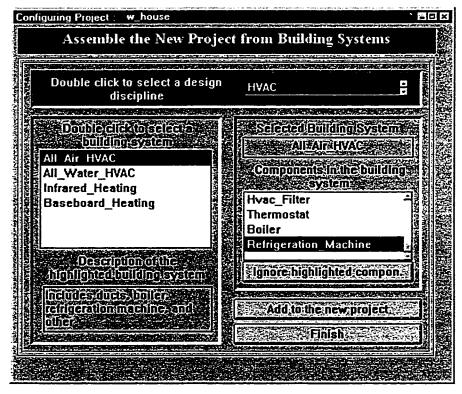


Figure 3-4 Assembly of building components in building-components-database from building systems available in general-database.

The model introduces a concept that directly tackles the crux of the problem. It assigns the responsibility of propagating design changes to the building components themselves. As such, when a designer modifies any attribute that describes a property of a building component, the component itself becomes responsible to find out who in the design team is affected by that design change. The component becomes also responsible to send messages to the affected designers. These messages inform them about the design change and advise each one of them on how such a change may have an impact on his/her design (Figure 3-5).

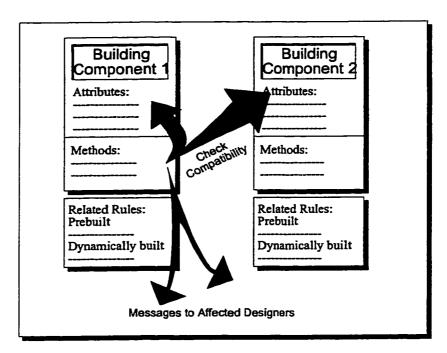


Figure 3-5 Concept of active building components.

For example, if the envelope designer decides to increase the height of a wall, the wall itself notifies the structural engineer of this decision and advises him/her that such a decision may affect the design of the girder that carries this wall. By depending on the

building components themselves rather than the designers to propagate design changes, we are eliminating the possibility that the envelope designer, who initiates the change, may forget to propagate the change, think that it is too insignificant to mention, or not be aware that another discipline is affected. We also compensate for the ability of the structural designer to discover such a change when reviewing the wall detail drawings.

On the basis of the above example where only one element of data has changed, it is evident that the proposed concept of having building components propagate design changes becomes highly advantageous in real cases where hundreds of such changes occur daily within the thousands of data elements that exist while developing the detailed design of a building project.

With such concept, the building-components-database is no longer a mere repository of idle data, but rather a repository of "active" building components that help the design team coordinate the building design information.

To make building components active in conducting their assigned task, they are connected with methods (groups of procedures). These methods make a building component active as soon as a design change occurs to any of its attributes. The components are also equipped with the necessary "linking" knowledge which identifies the disciplines that are affected by a specific design change and how they are affected.

The model is developed so as to acquire this knowledge in the form of rules. These rules,

like the building components, are customizable so as to reflect the requirements of a specific building project. The model considers two types of rules, *prebuilt* rules and *dynamically built* rules. Prebuilt rules are acquired before the beginning of the detailed design stage while configuring the model. The dynamically built rules are acquired "on the fly" during the detailed design stage (i.e. during the use of the model). For example, a prebuilt rule would reflect the fact that a change in the height of a wall affects the structural system design. An example of a dynamically built rule would be the design decision to make the height of a wall in a one-story building equals the clear height plus 1000mm.

3.2.3 Propagating Current Design Changes

When a design change occurs to a building component attribute, the component becomes active and checks all relevant prebuilt and dynamically built rules. If any of these rules applies to the design change, the component automatically sends messages to the affected designers.

Figure 3-6 gives an example of a message that is automatically sent to the envelope designer as a result of a design change made by the architect. The designer receives the message through the *Message Module* (see Figure 3-1 on page 36). This module allows the designer to review and modify the rules that cause messages to be sent, thus providing the model with the flexibility to adopt new relations that designers may develop among building components. When the envelope designer in turn takes a corrective action,

another message (Figure 3-7) is automatically sent to any other affected designer.

With the concept of active building components, we are now in a position to propose a solution to the main source of problems when design changes occur. Naturally, we expect that the involved designers will assume the responsibility to take the necessary actions as soon as they are informed of design changes.

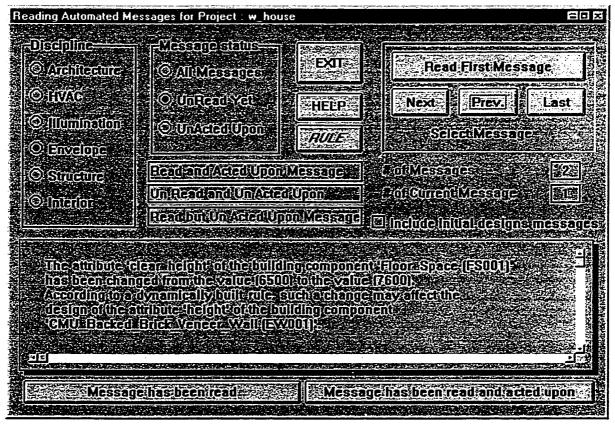


Figure 3-6 Message received by envelope designer informing of design change made by architect.

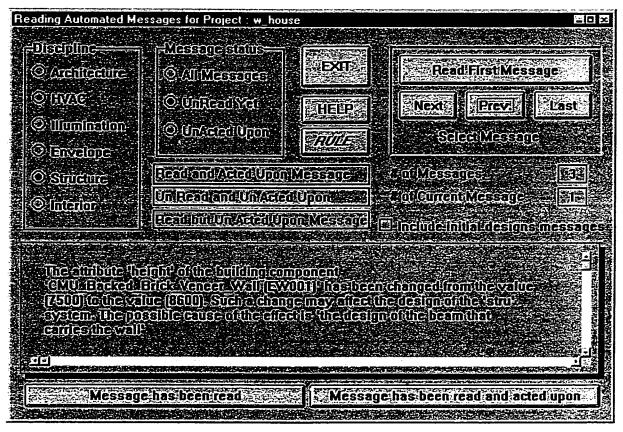


Figure 3-7 Message received by structural engineer informing of design change made by envelope designer.

3.2.4 TRACKING PAST DESIGN CHANGES

For design managers to manage the production of construction technical documents at the detailed design stage better, the history of design changes in a project represents an important source of information. The tracking of such history enables design managers to analyze the development of the production process. They can then evaluate the performance of the various designers in a project (e.g. designers who make poor design decisions which need to be later modified, costing the design team time and money), and can also determine responsibilities for design incompatibilities. With this in mind, the model is developed to provide the following capabilities through its *Tracking Module*

(see Figure 3-1 on page 36):

• Retrieval of design changes that are:

made by a discipline (e.g. structure),
made to a building component (e.g. concrete beams),
made to an instance of a building component (e.g. concrete beam001),
made to a single attribute of a specific instance of a building component (e.g.
depth of concrete beam001).

The retrieved design changes can also be limited over a period of time.

• The model can show the design manager all automated messages that have been sent by a specific building component in response to the design changes affecting it. The model also indicates which discipline has received each message and whether designers of this discipline have read the message and taken corrective action.

Figure 3-8 shows the design manager interface to the tracking module while Figure 3-9 gives the result of a request to monitor the design of a building component. Figure 3-10 shows one of the messages which were automatically sent to a member of the design team (as in Figure 3-7) because of a specific design change. The color of the message determines whether the designer who received the message read it and whether a corrective action was taken or not.

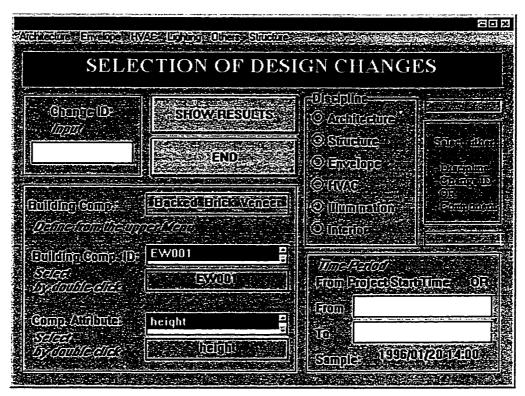


Figure 3-8 Interface for tracking module.

				BOX
QUERY RESULTS				
and the contract of the contra	aphilia interior at the Analysis of Tark	The second secon	يعيطه ومستركون وماليا المعرس كالوادا	
Deciding Edding Components Co	mponent(D	Com	ponent Attr	ibutes
· 1 7/9	EW0012		Ahelght	
Time Parpit	Erone (0. 119.32.3		
	Tos			
FIROUMANIQUES ASSISTANCE OF THE	υ.			
TO DESCRIPTION OF THE PROPERTY	Compile	Attributes	SEED CLEEK	News
			A CONTRACTOR OF THE PARTY OF TH	Value
1997/06/2015-10 enve Red Bried Ver				75001
MANUACA INSTANCE EXCEPTION AND	EW001	height	7500	86003

		វ ខេ	- TS 1 - 1 - 2 - 2 - 2 - 2 - 2 - 2 - 2 - 2 -	
ලේ නොන්න නිව්යන්න වැටින්න ලේ නොනෙන දේ ශානන්නේන්න	iteaningion हर्माक्ष्यम	īc E	· OK	

Figure 3-9 Results of request made by design manager to track design changes made to a building component.

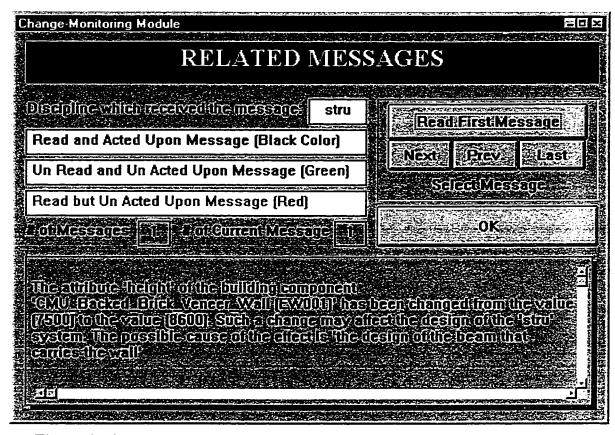


Figure 3-10 Results of request made by design manager to track automated messages that are sent because of a design change.

3.2.5 PLANNING AND SCHEDULING FUTURE DESIGN CHANGES

Many design changes impact only a limited number of building components. Yet, some changes result in a sequence of design changes to related building components. For example, in a specific building project, the owner needs to change the function of a certain space from an executive office to a computer lab. Such a change (indicated as A1 in Figure 3-11) which is to be performed by the architect leads to several other changes that need to be performed by other disciplines. The lighting designer needs to modify the type of luminaires that are used in the space (L1). The interior designer needs to change the wall finishing (I1). The HVAC designer needs to increase the size of the supply duct

to the space. S/he has the option to do that by increasing the width of the duct (H1) or the depth of the duct (H2).

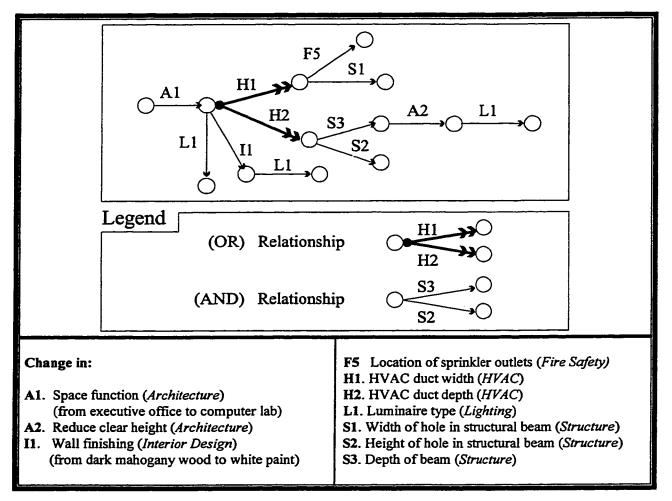


Figure 3-11 Scenario for interrelated design changes

When these design changes (L1, I1, and H1 or H2) are performed, each of them will result in the need for further design changes. As the wall finishing changes (I1), the type of luminaires (L1) needs to be reevaluated and may be changed. If the width of the duct is changed (H1), then the location of the sprinkler outlet needs to be changed (F5) and the width of the hole in the structural beam, where the duct passes, needs to be changes (S1). However, if the depth of the duct is changed (H2), different changes are needed. The

height of the hole in the structural beam will be changed (S2) as well as the depth of this beam (S3). The later change will result in lowering the suspended ceiling level and reducing the clear height (A2). Due to the lowered clear height, the type of luminaires (L1) needs to be reevaluated and may be changed.

This scenario illustrates the "domino" or interrelated effect that a design change in one discipline can have on several other disciplines. For simplicity, changes that recursively affect each other in closed loop fashion are omitted. Such types of design changes represent a challenge for the design team, specially the design manager, in many aspects. Among these aspects is how to predict the interrelated design changes that result from an initial design change such as (A1). The lack of the linking knowledge that enables such prediction makes this task very difficult. Another aspect is the possibility to have more that one solution to accommodate a design change as in the case of the HVAC designer who has the alternative to follow change (H1) or change (H2). Each alternative results in different "group of design changes" (or change-path). It is difficult not only to determine all the possible change-paths to accommodate an initial design change but also to select which one of them is the best to follow. Another aspect of the challenge is the possibility that some design changes such as (L1) may be repeated several times. It is hard to know when such a change should be performed in order to avoid the repetition.

To assist the design team in facing the challenges of interrelated design changes, the capability of planning and scheduling future design changes is built in the model. A methodology has been developed and implemented for this purpose using both the

Virtual Change Module and the Planning and Scheduling Module shown in Figure 3-1 on page 36. Figure 3-12 shows the result of applying the methodology on the example given in Figure 3-11. The figure illustrates that the methodology identifies the possible changepaths and properly locates design changes so as to avoid repetitions. Figure 3-13 shows the same result as it is displayed to the design manager for one of the possible changepaths. The detailed development of the methodology is presented in chapter five and in Mokhtar, Bédard and Fazio (1997a).

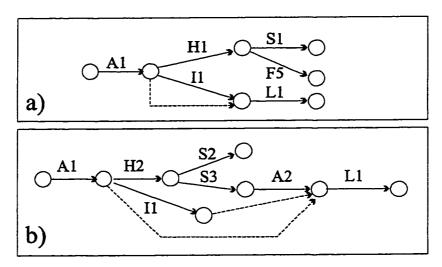


Figure 3-12 Alternate change-paths for scenario in Figure 3-11

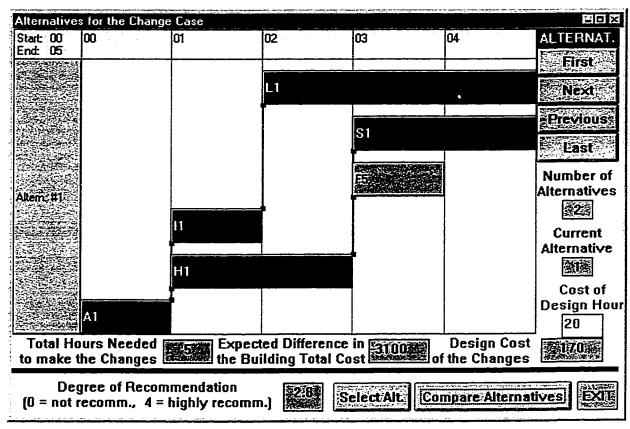


Figure 3-13 Bar chart for alternate change-path (a) in Figure 3-12

3.3 DISCUSSING THE MODEL ADVANTAGES AND LIMITATIONS

The architecture of the model provides a data-centric rather than a drawing-centric environment. It also provides a single repository of the design data. As such, the model architecture solves two of the diagnosed problems (section 1.1.3) that cause incompatibility errors in construction technical document. The capability of the model to automatically propagate design changes solves the main design change problem. This capability ensures that every member of the design team is updated with the data that s/he is using as input to the design of his/her building system. The quality of the input data is absolutely essential to the quality of the design of any building system and its

compatibility with other building systems. There is no way for a good designer to provide suitable design with wrong or outdated data. However, the model does not guarantee that a design decision that is taken by a designer is the appropriate one.

Some approaches to address the design change problem help designers in recognizing these changes through providing them with the updated drawings of the other disciplines. These approaches depend on the designers to recognize the design changes through comparing the updated and old drawings. The approach taken here is to directly inform only the affected designers with textual data that clearly shows the change. As such, no room is left for failure to recognize small changes or non graphical changes.

The approaches that use knowledge-based systems to coordinate design information depends on built-in coordination knowledge that should automatically correct any incompatibility errors. It is mainly oriented to help inexperienced designers who may not be able to take correct decisions. In contrast, the approach of this model is to recognize the experiences of the involved designers. Yet, it supports them with correct and updated information that they need to make their design decisions. We see this as a more realistic approach. It uses computers in what they excel at, which is dealing with large amounts of data, and designers in what they excel at, which is creativity that is built on experience. Because of its dependence on knowledge that is provided by the developers, the knowledge-based approach will always have a limited capacity to coordinate design data (El-Bibany 1992). Such knowledge is difficult to reflect the relationships among different building components which vary from one building to the other. This model uses only

linking knowledge and does not use any induction knowledge.

The model is developed in such a fashion that provides flexibility to both the datastructure of the building components and the linking knowledge that define the relations among these components. Such dual flexibility is a great advantage to the model as it reflects the nature of building projects. However, this flexibility requires the model to be configured to suit every project. This consumes extra time from the design team. Nevertheless, this time is seen as an investment that will pay off in reducing the long time that is usually consumed in coordinating and in solving incompatibility problems.

The model capability to plan and schedule future design changes is unique and has not been addressed before to our knowledge. This capability, however, depends on collaboration among members of the design team.

The architecture of the model currently lacks a tool to accept and show graphical data. The inclusion of such a tool will greatly ease the use of the model. The problem of data security and control of access is not addressed by the model nor the problem of the legal boundaries among the various involved disciplines. The model is more oriented toward design firms that include all the needed design disciplines under one legal entity. However, with the growing use of "Partnering" in the construction industry, this limitation may be overcome. The model is developed with a vision to use client-server networks and therefore limited to be used within one building or close-by buildings. It is also oriented to sequential type of project delivery systems. It cannot function, for

example, in a fast track delivery system.

3.4 SUMMARY

The crux of the design change problem, which is propagation of design changes, is directly tackled by the proposed information model. The model introduces the concept of active building components where each component is responsible for propagating design changes that occur to any of its attributes. The model is also developed with a unique management-database. This management-database allows the model to have flexible data-structure that is adaptable to the requirements of any building project. It also provides the model with the capability to track past design changes and plan and schedule future design changes. In this chapter, the overall architecture of the model is presented in conceptual perspective. The following two chapters detail the development of these concepts.

CHAPTER FOUR

THE CENTRAL PROJECT-DATABASE: A DETAILED PERSPECTIVE

4.1 Introduction

The heart of the proposed information model, as shown in Figure 3-1 on page 36, is the central project-database. This project-database is composed of two parts, the management-database and the building-components-database. The management-database contains data that is required to manage the various functions of the information model. The building-components-database functions as a repository of all the design data that is necessary to describe every building component in a certain project. To enable recording this data by the design team, the building-components-database needs to have a suitable "data-structure" that echoes the nature of building components that exist in a certain project. However, the building-components-database is not a mere repository of idle data, but rather a repository of "active" building components that help the design team coordinate the building design information.

This chapter shows the used technologies and the detailed development for the project-database so as to built it as a flexible repository of active building components. The chapter also shows the process of assembling and configuring the project-database through the use of the configuration module and with the assistance of the general-database (see Figure 3-1 on page 36).

4.2 FLEXIBLE REPOSITORY OF ACTIVE BUILDING COMPONENTS

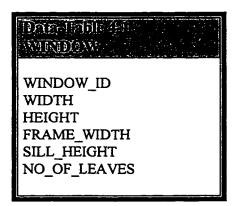
The primary function of the central project-database is to be a repository of active building components. Because of this function, structuring the project-database needs to satisfy two main requirements. The first requirement is to make the database flexible enough to accept the design information of any building. The second requirement is to have the building components in this database active in managing design changes as this is the main objective of the model. This section shows how the model is developed so as to satisfy these two requirements.

4.2.1 Providing Flexibility

Buildings are unique products. A hospital contains very different components from those of a warehouse. Furthermore, a warehouse may share some components with other warehouses but not all. For example, an air-heated warehouse may contain air distribution ducts while a radiant heated warehouse will not. Therefore, the definition of a data-structure (data-model) for the building-components-database may follow either one of the following two strategies. The *first* strategy is to create a universal data-structure for the building-components-database that covers all possible building projects. The *second* strategy is to customize the data-structure of the building-components-database to suit individual building project. The *first* strategy should result in a reusable and therefore desirable data-structure, yet it requires the inclusion of a complete set of available and newly introduced building components. As the number of these components is virtually unlimited, this strategy rapidly becomes impractical. In this research, the *second* strategy

is used to define the data-structure of the building-components-database which is the flexible part of the project-database. The management-database, on the other hand, has a fixed data-structure. For reference purposes, appendix "A" displays the conceptual data-structure of the management-database. Copies of its data-tables are also used through this chapter to facilitate the explanation.

To fulfill the flexibility requirement, the *Relational Database* technology is utilized to structure the project-database. This technology allows dynamic and transparent manipulation of data-structure and data values through an interface program by using the Structured Query Language (SQL). With such technology, each *building component* (e.g. window, beam, duct) is represented as a *data-table*. Each data-table has a different data-structure that represents the attributes describing each component. Data-Table 4-1 shows a sample of the structure for the data-table that represents the building component "Window". The *instances* of a building component (e.g. window001, window002) are stored in their own data-table. Figure 4-1 shows how these instances are stored in Data-Table 4-1.



30. mg/f/K	e canal	ाहरूको ।	IR TOUR TOUR	्यः आधित्वयुर्गे	s modere (
Window001	3000	1200	200	900	3
Window002	2500	1200	200	900	2

Figure 4-1 Sample of data in data-table "Window" which describes instances of the building component "Window".

The use of SQL statements enables adding, deleting and modifying data-tables that represent various building components in the building-components-database. For example, if the architect decides to use circular windows in addition to rectangular windows, s/he needs to have a data-table that contains the attribute "Radius" instead of the attributes "Width" and "Height" that are in Data-Table 4-1. Asking the design manager to create such a data-table, s/he specifies the name and the attributes of the data-table. Using SQL, the data-table "Circular_Window" is created in the building-components-database. The SQL statements for this operation include these shown in the zigzag¹ frame. The first statement builds a new data-table with an initial attribute "Circular_Window_ID" that accepts alphanumeric values. The second and the third statements modifies the created data-table by adding two other attributes, "Radius" and "Frame_width" that accept numeric values.

CREATE TABLE [Circular_Window] [Circular_Window_ID] TEXT; ALTER TABLE [Circular_Window] ADD [Radius] NUMERIC (5); ALTER TABLE [Circular_Window] ADD [Frame_width] NUMERIC (5);

The interface program (the configuration module) creates these SQL statements

¹ Text within a zigzag frame is extraction from the program code.

automatically, the design manager needs only to provide the name of the building component and its attributes (Figure 4-2). S/he does not need to know how this data is structured in the building-components-database. The created data-table (Data-Table 4-2) can now carry the data of all the circular windows in the project. The configuration module also enables the modification of the data-table at any later time to accommodate the need to describe the building component with further details.

CIRCULAR_WINDOW_ID
RADIUS
FRAME_WIDTH
SILL_HEIGHT

Yet to enable the use of SQL in providing this capability, a specially developed data-table (Data-Table 4-3) in the management-database is used¹. This data-table records data that describes the data-structure of the various data-tables in the building-components-database. As such, the structure of these data-tables can be tracked and manipulated. Figure 4-3 shows sample data of Data-Table 4-3. The sample shows the data that describes the data-table "Window" and the data-table "Circular_Window".

¹ Names of all data-tables that belong to management-database start with MNG.

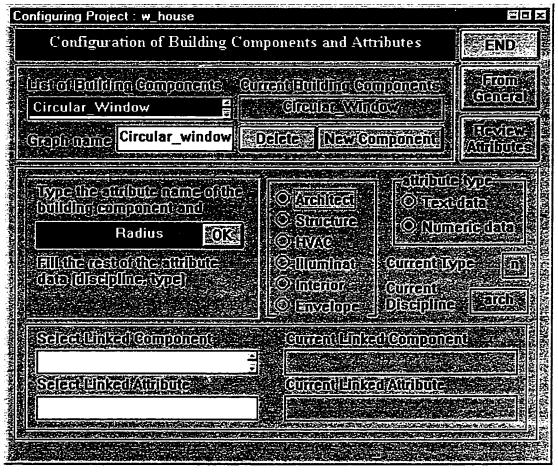


Figure 4-2 Configuration of the building component "Circular_Window" using configuration module.

MEGRANCES MINDRES MARKINGARDS

ID
BUILDING_COMPONENT
ATTRIBUTE_OF_BUILDING_COMPONENT
ATTRIBUTE_TYPE
DESIGNING_DISCIPLINE
LINKED_BUILDING_COMPONENT
LINKED_ATTRIBUTE_OF_BUILDING_COMPONENT
GRAPH_NAME

'प्रतृति			TOTAL CONTRACTOR	Treating training that	in on a ora
	Window	Window_id	T	Arch	Window.pcx
20	Window	Width	N	Arch	
21	Window	Height	N	Arch	
22	Window	Frame_width	N	Arch	
23	Window	Sill_height	N	Arch	
	Window	No_of_leaves	N	Arch	
25	Circular_Window	Circular_Window_id	T	Arch	Circular_Win
26	Circular_Window	Radius	N	Arch	
27	Circular_Window	Frame_width	N	Arch	

Figure 4-3 Sample of data in Data-Table 4-3 that describes data-structure of the building components "Window" and "Circular_Window".

4.2.2 Making the Building Components Active

The second requirement for the design of the central project-database is to make the building components active in managing design changes. To fulfill this requirement, the Object-Oriented (OO) technology is adopted. In this technology, an object is a collection of related variables and procedures (Taylor 1992). Objects encapsulate methods, which are groups of procedures. When a designer modifies the design of a building component (e.g. window), the attributes of this building component (e.g. width, height) are retrieved into an OO environment (Figure 4-4). Each attribute is temporarily represented as an

¹ The headings are truncated and some may have been removed in order for the figure to fit on the page.

object and consequently becomes encapsulated with a method. The method is triggered when a design change occurs to the value of the attribute. The method retrieves some rules that are related to the attribute (e.g. a 20% change in window height affects the daylight design) and checks the applicability of the rules to the design change. If any of the rules is applicable, the method automatically sends messages to the affected designers informing them about the change and asking for an appropriate response (more details in chapter five). The rules are stored in specially developed data-tables that are part of the management-database (for example, see Data-Table 5-1 on page 84 and Data-Table 5-2 on page 84).

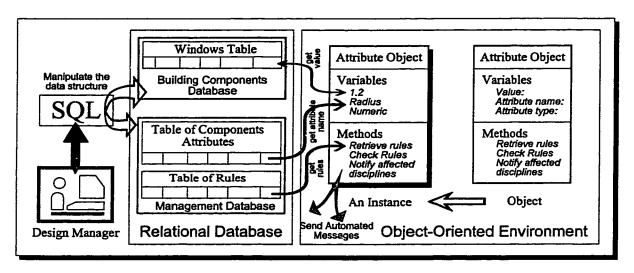


Figure 4-4 Combination of relational database and object-oriented technologies.

With such a combination of the relational database and OO technologies, only the relational database structure needs to be modified so as to accommodate the requirements of a project. Building components can be added, deleted, or modified. Their properties can be also changed. Rules can also be added or modified in the relational database.

The problem of flexibility and extendibility of data-structure may be approached with the sole use of the OO technology. This approach will ultimately face the shortcomings of this technology, mainly (Elmasri and Navathe 1994):

- No high level query language is available so as to allow the dynamic and transparent manipulation of data-structure and data values by the end user through an interface software. The end user therefore needs to know how to use the development environment (usually an OO database management system). Such a requirement will severely limit the practicality of the model.
- Behavior rigidity, where a fixed set of rules has to be pre-determined and prespecified during the model development, makes it unfeasible to add and modify rules to building components during execution to suit specific project requirements.

The approach, which combines both the relational database and the OO technologies, provides a more practical solution.

4.3 ASSEMBLY AND CONFIGURATION

As shown in the previous section, the project-database is built to be a flexible repository of active building components that suits the requirements of individual building projects. As such, the building-components-database part of the project-database needs to be assembled for every new project. The management-database also needs to be configured for the project at hand. While the design team uses the model, some reconfiguration may

be required as well to reflect the changing requirement of a project.

This section shows how the configuration module is developed to enable the design manager perform these tasks. Figure 4-13 on page 78 is a diagram that can be used as a reference to show the user interface through the configuration module.

4.3.1 ASSEMBLY OF THE BUILDING-COMPONENTS-DATABASE OF A NEW PROJECT

When starting a new project, the design manager activates the configuration module. As the configuration module is instructed for a new project, it creates a new project-database file with the name of the project as provided by the design manager. The new project-database file is created as a copy of an already built standard database that only includes unpopulated data-tables of the management-database (section A.1.2). No data-table exists a priori in the building-components-database. The design manager is then required to define the data-structure of every building component that exits in the project at hand so as to enable members of the design team to record design information.

Because a building consists of a large number of components, such task is extremely difficult and time consuming. It will definitely discourage the use of the model. Therefore, a general-database is developed to provide assistance to design managers in performing this task. The general-database as shown in Figure 3-1 on page 36 is not part of the central project-database, rather a supporting resource that exists separately and can be used simultaneously by several design managers for different projects.

4.3.1.1 The General-Database

The purpose of the general-database is to provide assistance to design managers in assembling the building-components-database. It is able to do so by containing data-structure for a large pool of building components. Three main observations influence the development of the general-database:

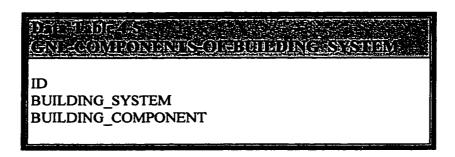
- The first observation is that building components are repeatedly used. As most of the
 building components are manufactured, they have to be widely used to reduce their
 cost. Nevertheless, only a limited number of the available building components are
 used in a certain project.
- The second observation is that building components can be grouped into systems. For example, a column and beam structural system will contain components such as concrete foundation, concrete column, concrete beam, and concrete slab.
- The third observation is that some buildings resemble other buildings in most of their building components. For example, most wood construction homes will share most of their components.

The general-database development, as shown in Figure A-2 in Appendix "A", reflects these observations through assembling building parts into three levels of increasing complexity: Components, Systems, and Configurations. Members at the Components level such as doors, beams, ducts, and luminaries are the primary ingredients in a building. Each

component is described through its attributes. The description is recorded in a special datatable (Data-Table 4-41) which carries data about the data-structure of a variety of building components. Figure 4-5 shows a sample of the data in the Data-Table 4-4. The next level is the Systems level. Examples for members of this level are cast-in-place flat slab concrete structural system, all-water HVAC system, and restroom system. Each system contains a group of related components that form the system. The restroom system, for example, includes lavatory, toilet, urinal, sink, partition, and mirror. Each of these components is fully described in the components level (Data-Table 4-4). A special data-table in the general-database (Data-Table 4-5) carries the data which identifies the building components in various building systems. Figure 4-6 shows a sample data in the Data-Table 4-5. The highest level is the Configurations level. Members of this level are complete building configurations. Examples are a concrete structure low-rise office building with all-air HVAC system, or a steel structure warehouse with infrared heating. Such configurations are required when a data-structure for a similar project needs to be assembled. The generaldatabase carries a list of existing projects in Data-Table 4-6. Figure 4-7 shows a sample of the data in Data-Table 4-6.

¹ Names of all data-tables that belong to general-database start with GNL.

DAG PIDIC COMPONENTS ATTRIBUTES. ID BUILDING COMPONENT ATTRIBUTE OF BUILDING COMPONENT ATTRIBUTE TYPE DESIGNING DISCIPLINE LINKED BUILDING COMPONENT LINKED ATTRIBUTE OF BUILDING COMPONENT GRAPH NAME

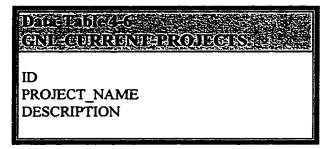


			TO THE		er lieu zetaki		
389	Lavatory	Lavatory_id	t	arch			Lavatory.pcx
390	Lavatory	model	t	arch			
	Lavatory	color	t	arch			
392	Lavatory	width	n	arch			
	Lavatory	depth	n	arch			
394	Lavatory	height	n	arch			
395	Lavatory	mounting_level	n	arch			
396	Lavatory	located_in_space	t	arch	Space	Space_id	

Figure 4-5 Sample of data in Data-Table 4-4 that describes data-structure of variety of building components.

		2.555.55
48	Restroom_Components	Restroom_Partition
49	Restroom_Components	Restroom_Door
50	Restroom_Components	Міггог
51	Restroom_Components	Toilet
52	Restroom_Components	Urinal
53	Restroom_Components	Lavatory

Figure 4-6 Sample of data in Data-Table 4-5 that defines building components that belong to a building system.



聚的數	s projecté da mes	and the second s
5	commer_I	3 story commercial building with columns and beams concrete structure and all water HVAC system
6	w_house	2 story ware house building with open web steel joist structure on steel columns and all air HVAC

Figure 4-7 Sample of data in Data-Table 4-6 that records current projects.

The role of the general-database is not limited to be a large pool of data-structure. As the prebuilt rules are usually valid for most buildings (for example, a change in the thermal resistance of an external wall affects the HVAC system design), they are considered reusable pieces of information and can be included, with some minor modifications, in every new project. The general-database therefore includes this type of rules. The prebuilt rules are stored in Data-Table 4-7. Figure 4-8 shows a sample of the data in Data-Table 4-7.

PREBUILT_RULE_ID BUILDING_COMPONENT ATTRIBUTE_OF_BUILDING_COMPONENT AFFECTED_DISCIPLINE PERCENT_CHANGE_TO_HAVE_AN_EFFECT RESULT_IN_MAJOR_MODIFICATION_? RULE IGNORED ?

Data-Pable 4-7

CAUSE OF THE EFFECT

1:			न्ति ५०० स्टब्स्		7	3, 37	
19	Window		hvac	0.15			Heat gain or loss through the window
20	Window	width	hvac	0.25	No	No	Heat gain or loss through the window
21	Window	width	illu	0.25	No	No	Daylight entering the space

Figure 4-8 Sample of data in Data-Table 4-7 that describes some prebuilt rules.

The general-database is also implemented with the use of the relational database technology. The configuration module uses the SQL capabilities to pick a component, a system, or a configuration from the general-database and reconstruct it in the building-components-database of a new building project.

4.3.1.2 The Assembly Process

As the design manager activates the configuration module to assemble the building-components-database of a new project, s/he is provided with three options (selection icon S2 in Figure 4-13). The design manager can perform the assembly task through building components, systems, or existing projects. These options follow the same rational used in developing the general-database.

In case the design manager decides to assemble the building-components-database from another project, the configuration module connects itself to the project database of that project. It reads the data in Data-Table 4-3 (page 62) in the management-database of that project and creates in the building-components-database of the new project all the datatables that represent the building components and their attributes. The configuration module also populate the Data-Table 4-3 in the management-database of the new project with data that describe the created building components. The prebuilt rules are retrieved

as well from the existing project and recorded in the new project.

In case the design manager decides to assemble the building-components-database of the new project from the building systems (in the general-database), the configuration module recognizes, from Data-Table 4-5 (page 68) in the general-database, all the building components that belong to that system. The configuration module retrieves the data-structure of each of these building components from Data-Table 4-4 (page 68) in the general-database. It creates in the building-components-database of the new project all the data-tables that represent these components. It also populates Data-Table 4-3 (page 62) in the management-database of the new project with data that describe the created building components. The relevant prebuilt rules are also retrieved from Data-Table 4-7 (page 69) in the general-database and recorded in the management-database of the new project in Data-Table 5-1 (page 84).

In case the design manager decides to use the components level in the general-database, the configuration module retrieves the data-structure of the needed component from Data-Table 4-4 (page 68) in the general-database and creates a data-table for that component in the building-components-database. Similar to the previous cases, the configuration module populates the Data-Table 4-3 (page 62) in the management-database of the new project with data that describe the created building component. The relevant prebuilt rules are also retrieved from Data-Table 4-7 (page 69) in the general-database and recorded in the management-database of the new project in Data-Table 5-1 (page 84).

Using the general-database in assembling the building-components-database facilitates the task considerably. However, the assembled data-structure has to be fine-tuned for the requirements of a specific project. Such fine-tuning is performed using the same capabilities that exist to configure an existing project (project that already uses the model). Next section shows how the configuration module is developed for this purpose.

4.3.2 CONFIGURING THE PROJECT-DATABASE OF AN EXISTING PROJECT

During the development of detailed design, the requirements of a building project slightly vary. Some new building components may need to be added, some existing building components may need to be described in further detail, or the rules that link some building component may need to be modified. Because the project-database is built to be flexible, such variations can be accommodated in the model.

The configuration module is developed to provide the design manager with the ability to configure the project-database. As Figure 4-13 shows (selection icon S3), there are three categories of configuration to an existing project. These are the configuration of building components, the configuration of prebuilt rules, and the configuration of the project menu which allows designers to access the building components in the project-database.

4.3.2.1 Configuring Building Components

The configuration module provides the design manager with several options to configure building components. As selection icon S4 in Figure 4-13 shows, the design manager can create data-structure for a new building component, modify or delete the data-structure of

Ahmed Mokhtar, Ph.D Thesis72

an existing building component.

When creating data-structure for a new building component, the design manager is assisted with an access to the general-database. If the needed building components is available in the general-database, the design manager simply pick the component. Similar to the process in section 4.3.1.2, the configuration module uses the capability of SQL to retrieve the data-structure of the new component from the general-database and to build it in the building-components-database. Data-Table 4-3 in the management-database is also populated with data that describes the newly created building component. However, If the needed building component does not exist in the general-database, the design manager creates through the configuration module a new data-table in the building-components-database. The new data-table represents the new building components. The design manager needs to specify all the attributes that describe the component. This is similar to the creation of the Circular Window in section 4.2.1.

In case the design manager needs to modify the data-structure of a certain building component (example in Figure 3-3 on page 40), the configuration module shows the design manager the current data-structure of the data-table that represent this building component. It does this by retrieving the relevant data from Data-Table 4-3 (page 62) in the management-database. As the design manager modifies the data-structure of the building component, the configuration module using SQL capabilities adds, deletes, and modifies attributes of the data-table of the building component in the building-components-database. The changes are also reflected in Data-Table 4-3 in the

management-database.

In case the design manager wants to delete unused building component, the configuration module also uses SQL to remove the data-table of this component from the building-components-database. It removes the corresponding data from Data-Table 4-3 as well.

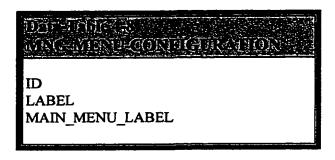
4.3.2.2 Configuring Prebuilt Rules

The design manager can also manipulate the prebuilt rules (Figure 4-10 on page 76). S/he has an access to all the attributes of all the building components in the building-components-database. For each attribute, s/he is able to browse through all the rules that are related to the attribute. The design manager is able to make modifications in any rule and update the rule. S/he is also able to temporarily ignore a rule (selection icon S6 in Figure 4-13). For a certain attribute, the design manager is able to create new rule and define all the data element of this new rule. The configuration module, in each case, makes modifications in Data-Table 4-7 (page 69) in the management-database using SQL capabilities.

4.3.2.3 Configuring the Project Menu

Another necessary configuration that is needed for the model to function is the project menu. This menu enables the designers to access the building components in the building-components-database through the modules of the model as shown in Figure 4-11. The design manager classifies the building components into groups as shown in Figure 4-12. Each group has a menu heading (e.g., Architecture, Structure). Data about

the contents of a menu is stored in Data-Table 4-8 in the management-database. Figure 4-9 shows a sample of the data in Data-Table 4-8. The same menu items appear in all modules that require access to the building components in the building-components-database as in Figure 4-11.



	THE COURSE WHITE CONTROL STORMS TO SHEET HERE!								
8	Concrete_Topped_Steel_Decking	Structure							
12	Steel_Angle_Connection	Structure							
13	Steel_Plate_Connection	Structure							
35	Open_Web_Steel_Joist	Structure							

Figure 4-9 Sample of data in Data-Table 4-8 that describes configuration of project menu.

Configuring Project : w_house	富田文
PREBUILT RULE (CONFIGURATION
<u>10 දින්න</u> රාජ්ය විශ්ය ව විශ්ය විශ්	Salad े श्वितिक रिकारकारतः CMU_Backed_Brick_Veneer_W
orthe following Evil (fing component	Salasic Gomponani/Aluliule height
GYU Boered Boer Vencer Wall than the design of the bear that	Eni Gunantitabled Filler (Filler English) Gunantitabled Ruler (Filler English) Wiedet Diedolline Rebled General Rule
the design of the beam that a carries the wall	O Architect Tables Street Street
lignored please wire YES No hopored please wire YES No hopored files wile to be no hopored by the light of th	O HVAC O III umineti O interior O Envelope Nev-Hules
Updaterture	END

Figure 4-10 Configuration of a prebuilt rule using configuration module.

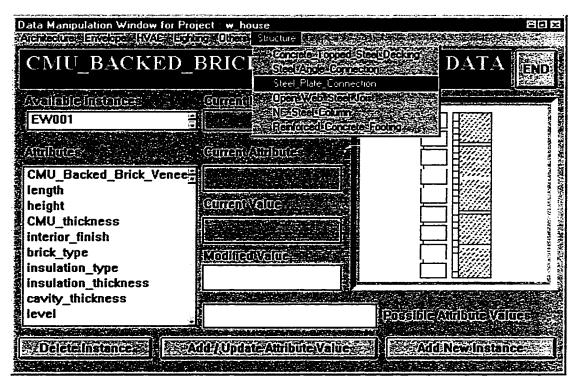


Figure 4-11 Project menu as it functions in data manipulation module.

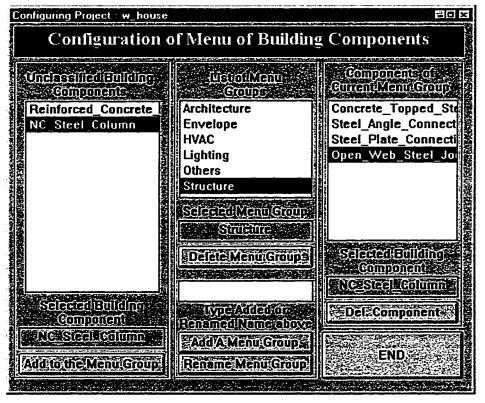


Figure 4-12 Interface to configure project menu to allow access to available building components.

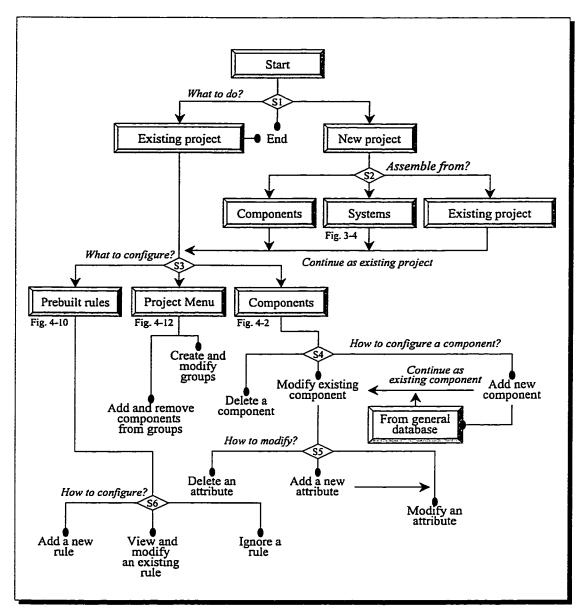


Figure 4-13 Diagram to illustrate user interface in configuration module¹.

¹ Please refer to the "Legend for User Interface Diagrams" on page XV.

4.4 SUMMARY

The building-components-database that carries the design data for the various components in a building project needs to be flexible in its data-structure so as to adapt to the variety of building designs. In addition, to conform with the core concept of the information model, the components in this database need also to be active in helping the management of design changes. The use of a management-database that carries information about the data-structure of the various building components, plus the use of combined Object-Oriented and Relational Database technologies, enable the satisfaction of these requirements. The development of a general-database with three levels of building components assembly facilitates the data-structure of the building-components-database. As such, the capability of the model to manage design changes can be utilized in virtually any type of projects. This is seen to be vital for the professional utilization of the model.

CHAPTER FIVE

MANAGING DESIGN CHANGES: A DETAILED PERSPECTIVE

5.1 Introduction

In chapter three, a conceptual perspective of the developed information model was presented. The perspective exhibited the model capabilities to manage design changes. These capabilities are not limited to propagating current design changes; rather they are extended to include tracking past design changes and planning and scheduling of future ones. This chapter exhibits the development of these capabilities in detail. For each capability, the objective is stated, the developed methodology is described, and the user interface is illustrated.

As will be shown throughout this chapter, the management-database plays a key role in the development of the model capabilities to manage design changes. For reference purposes, appendix "A" displays the conceptual data-structure of the management-database. Copies of its data-tables are also used throughout this chapter to facilitate explanations.

5.2 Propagating Current Design Changes

5.2.1 OBJECTIVES

As discussed in section 1.1.3 on page 8, a major cause of failure to accommodate design

changes in a multi-disciplinary design team is the lack of an effective mechanism to propagate design changes. The objective of providing the model with this capability — which is key to the model development — is to overcome this cause of failure. The aim is to provide the design team with an effective methodology that assists in accommodating design changes.

5.2.2 METHODOLOGY

Based on our professional experience, we argue that the inability of the design team to propagate design changes is due to two difficulties. The first difficulty is the large number of building components data that exists in a project. It is almost impossible to manually follow the changes that are experienced by the individual data items in a project. The second difficulty is the lack of the linking knowledge which tells what disciplines are affected by a specific design change in a building component and how they are affected. The lack of such linking knowledge is mainly due to the different educational background of the designers who belong to the various design disciplines.

To overcome the first difficulty, the model (as illustrated in section 3.2.2 on page 39) assigns the task of propagating design changes to the building components themselves, hence reducing the need for the inefficient human monitoring of the design changes. To overcome the second difficulty, the building components are provided with the required linking knowledge.

However, the main challenge that remains is in acquiring the linking knowledge. How

can we know the design disciplines that are affected by a design change, for example, in the height of the wall? Whoever attempts to answer this question should have design experience in all the disciplines in the design team, a requirement that is almost impossible to satisfy.

5.2.2.1 Acquire the linking knowledge

The developed methodology overcomes this challenge through reversing the knowledge acquisition process. Instead of trying to define the disciplines that are affected by a change in the value of every attribute of every building component, we define the building components attributes that are needed by every discipline. Unlike the normally required process, the reversed process is easily performed by asking each designer about the input data that s/he uses during the design of a certain building system. This input data is then translated into the form of attributes of building components. For example, the structural engineer defines that one of the input that is needed to design beams is the weight of the brick walls that are carried by the beams. The input data "weight of the brick wall" is then translated to the attributes "height, width, and brick type" of the building component "brick wall". If the value of any of these attributes is changed (by the architect or the envelope designer), then the structural engineer needs to be notified.

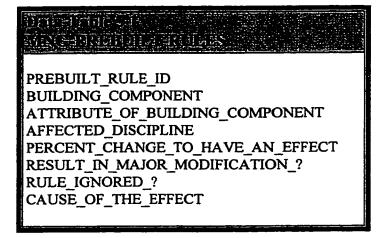
The input data acquired from the designers is stored in specially structured data-tables that are part of the management-database (Data-Table 5-1 on page 84 and Data-Table 5-2 on page 84). A sample of the data used in these data-tables is shown in Figure 5-1 (page 84) and Figure 5-2 (page 84). The acquired data are used in "if-then" rules that constitute

the linking knowledge. The rules are developed in such a fashion that makes them adaptable for the requirements of individual buildings. They contain variables that are later replaced by values during the execution of the model. A sample of these rules is shown in the following zigzag frame¹. The variables are %n, %nOld, and %nNew. The rule indicates that any change that is more than the percentage %n should result in sending a message.

%n := PERCENT_CHANGE_TO_HAVE_AN_EFFECT
%nOld := ATTRIBUTE_OLD_VALUE
%nNew := ATTRIBUTE_NEW_VALUE
IF (%n = 0) OR (%nOld * (1 - %n) > %nNew) OR (%nOld * (1 + %n) < %nNew) THEN SEND_MESSAGE_ELEMENTS</pre>

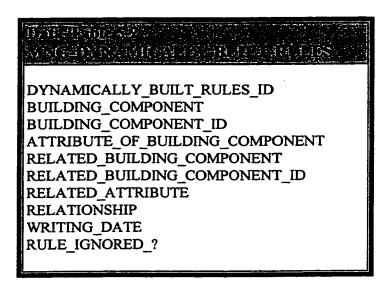
The values that replace the rules' variables are obtained from the design data and vary from one project to the other. The rules are classified into two categories, *prebuilt* rules and *dynamically built* rules. The prebuilt rules are acquired during the configuration process of the model. These usually represent general linking knowledge that is known beforehand. For example, a prebuilt rule would reflect the fact that a change in the height of a wall affects the structural system design. The dynamically built rules are acquired during the use of the model to represent the linking knowledge that can only be defined during design. An example of dynamically built rules would be the design decision to make the height of a wall in a one-story building equal to the clear height of the floor space plus 1000mm.

¹ Text within a zigzag frame is extraction from the program code.



			01.50				
12	Space	used_for_activity	hvac	0	No	No	heat gain and HVAC requirements in the space
34	CMU_Ba	height	stru	0.10	No	No	the design of the beam that carries the wall
50	Open_W	length	stru	0.20	Yes	No	major impact on the joists design

Figure 5-1 Sample of data in Data-Table 5-1 that describes some prebuilt rules1.



阿克森斯	Id new components			terstear pay	related	relatedbatt	WAR AND AND THE COURSE OF THE
	n Web Steel Joist						Boiler.weight<=2000
6 CM	U_Backed_Brick_V	EW001	height	Floor_Spac	FS001	clear_heigh	Wall.height=Floor_Space.clear_height+1000

Figure 5-2 Sample of data in Data-Table 5-2 that describes some dynamically built rules.

¹ The headings are truncated and some may have been removed in order for the figure to fit on the page.

To acquire the prebuilt rules, the design manager requests from the senior designer of each discipline to define all the input data needed to design the building system under his/her responsibility. Also defined is the percentage of change to each of these inputs that can affect the design. For example, the structural engineer defines that the "height" of "CMU_backed_brick_veneer_wall" is one of the input to design the beams that carry these walls. He also defines that a change of 10% of the "height" would impact the design of the beams. The design manager – using the configuration module (Figure 4-10 on page 76) – keys in this data. The data is stored transparently to the design manager as data elements of a prebuilt rule in Data-Table 5-1 as shown in Figure 5-1.

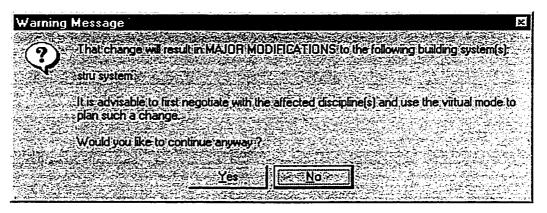


Figure 5-3 Sample of warning message about design change that results in major design modifications.

A designer can also define if a given change in an input data can result in a major redesign in his or her system. For example, the structural engineer defines that an increase of more than 20% of the span of the open web steel joists will require a complete redesign of all the joists. If the architect attempts to increase the distance between the supporting column by more than 20%, s/he will be immediately notified by a warning message that such a change has a major impact on the structural system (Figure 5-3). S/he

then has the choice to continue with the change or consult first with the structural engineer.

To acquire the dynamically built rules, a technique has been developed. When a designer inserts or changes a value of a building component attribute, s/he is automatically asked to provide relationships with other building components (if they exist) which represent the constraints that govern the inserted or changed value. Figure 5-4 (page 87) shows a sample of a relationship that the envelope designer defines after inserting the value of the attribute "height" of the building component "CMU backed brick veneer wall". The relationship states that the "height" of the "CMU backed brick veneer wall" should equal the "clear height" of the "floor space" + 1000mm. The data is stored transparently to the designer in Data-Table 5-2 (page 84) automatically. The component that is currently under design is stored in the data field "BUILDING COMPONENT" along with the current instance and attribute in the fields "BUILDING_COMPONENT_ID" and "ATTRIBUTE OF BUILDING COMPONENT" consecutively. The building component, which the envelope designer makes a relationship with, is stored in Data-Table 5-2 in the field "RELATED BUILDING COMPONENT" along with its instance attribute in the fields "RELATED BUILDING COMPONENT_ID" "RELATED ATTRIBUTE" consecutively.

The "clear height" of the "floor space" is a design decision that is taken by another discipline (the architect). The architect may not be aware that the envelope designer took a design decision that is based on his/her design decision for the clear height of the floor

space. The architect may later change the clear height without being aware of the effect on the envelope designer. However, the linking knowledge acquired from the envelope designer makes the building component "floor space" aware of the envelope designer dependency on its clear height. The "floor space" will immediately inform the envelope designer when the architect changes its "clear height".

As the designers of various disciplines populate the building-components-database, the linking knowledge is also built up to deal with any potential design changes that may be later performed.

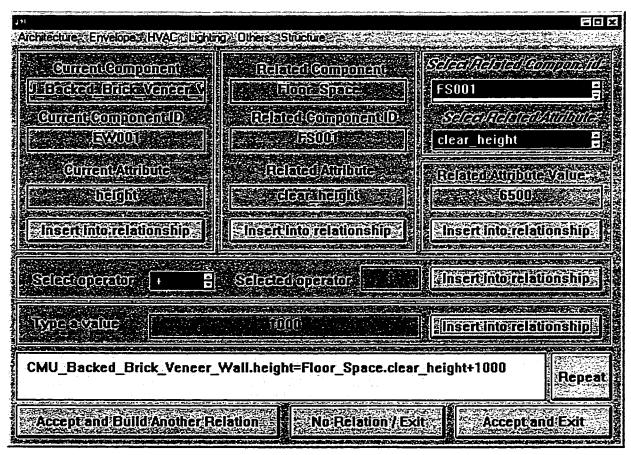


Figure 5-4 Acquisition of dynamically built rule using relation-acquiring submodule.

5.2.2.2 React to Design Changes

When a designer performs a design change, for example, to the attribute "Attr_X" of the instance "Inst_Y" of the building component "Comp_Z", the component becomes active and automatically reacts to the change as follows:

1. It retrieves from Data-Table 5-1 (page 84) and Data-Table 5-2 (page 84) all the data rows that are relevant to the design change. In Data-Table 5-1, the relevant rows are those with the field "BUILDING_COMPONENT" which have the value "Comp_Z" and the field "ATTRIBUTE_OF_BUILDING_COMPONENT" which have the value "Attr_X". In Data-Table 5-2, the relevant rows are those with the field "RELATED_ BUILDING_COMPONENT" which have the value "Comp_Z"; the field "RELATED_ BUILDING_COMPONENT_ID" which have the value "Inst_Y"; and the field "RELATED_ATTRIBUTE" which have the value "Attr X". A SQL statement similar to the following is used:

SELECT * FROM [MNG-PREBUILT-RULES] WHERE BUILDING_COMPONENT = "Comp_Z" AND ATTRIBUTE_OF_BUILDING_COMPONENT = "Attr_X" AND RULE_IGNORED_? = No

- 2. The data in each of the retrieved rows is used to check if some rules are violated by the design change.
 - a) In case of data from Data-Table 5-1 (prebuilt rule), the percent change in value of the changed attribute is evaluated. If it exceeds the value specified by the senior designer (and saved in the field "PERCENT_CHANGE_TO_HAVE_AN_EFFECT"), then the process continues as in part 3. If

Ahmed Mokhtar, Ph.D Thesis 88

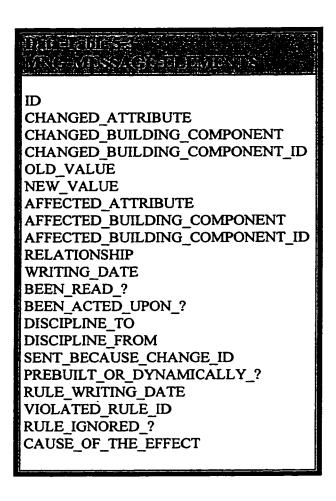
not, then the data of the next row is examined. Statements similar to the following are used.

In case of data from Data-Table 5-2 (dynamically built rule), the relationship built by the designer is examined. If it fails then the process continues as in part 3. If the relation continues to be valid then the data of the next row is examined. Statements similar to the following are used (based on the data in the second row in Figure 5-2 on page 84 and the changed attribute is the height of the floor space FS001). The statement search for the height of the instance "EW001" of the component "CMU_Backed_Brick_Veneer_Wall" under the condition that the height is 1000mm higher than the clear_height of the instance "FS001" of the component "Floor Space".

SELECT CMU_Backed_Brick_Veneer_Wall.height from CMU_Backed_Brick_Veneer_Wall, Floor_Space where (CMU_Backed_Brick_Veneer_Wall.CMU_Backed_Brick_Veneer_Wall_id = 'EW001') AND (Floor_Space.Floor_Space_id = 'FS001') AND (CMU_Backed_Brick_Veneer_Wall.height=Floor_Space.cle ar_height+1000)

IF no value is retrieved THEN SEND_MESSAGE_ELEMENTS

3. Messages are automatically sent by the building component to inform the affected designer about the change. The messages are saved in the form of data values that are recorded in a specially structured data-table (Data-Table 5-3) in the management-database. The data field that are to be filled depend on the type of rule that is used by the changed building component.



and the country	(a) Direction (dure	nosy.	105.28	alece	alesie	Heece	Did	diseli	9.8	orehull ore	100
65 clear_height		FS001	6500								Dynamically	
71 height	CMU_Backed_	EW001	7500	8600				Stru	enve	28	Prebuilt	34

Figure 5-5 Sample of data in Data-Table 5-3 that describes some message elements.

- In case that a prebuilt rule is violated, the following data are sent to the a) Data-Table 5-3.
 - CHANGED_ATTRIBUTE
 - > CHANGED BUILDING COMPONENT
 - > OLD VALUE

(of the changed attribute)

- > NEW VALUE
- (of the changed attribute)
- ➤ WRITING DATE (from the computer clock)
- ➤ DISCIPLINE TO (AFFECTED DISCIPLINE as in Data-Table 5-1 on page 84)
- DISCIPLINE FROM (DESIGNING_DISCIPLINE as in DATA-TABLE 4-3 on page 62)
- > SENT BECAUSE CHANGE ID (CHANGE 1D from DATA-TABLE 5-4 on page 99)
- ➤ PREBUILT OR DYNAMICALLY ? (prebuilt)
- > VIOLATED RULE ID (PREBUILT_RULE_ID from Data-Table 5-1)
- > CAUSE OF THE EFFECT (form Data-Table 5-1)
- In case that a dynamically built rule is violated, the following data are sent b)

to the Data-Table 5-3

- > CHANGED_ATTRIBUTE
- CHANGED BUILDING COMPONENT
- CHANGED_BUILDING_COMPONENT_ID
- > OLD VALUE

(of the changed attribute)

- > NEW VALUE
- (of the changed attribute)
- ➤ AFFECTED_ATTRIBUTE (ATTRIBUTE_OF_BUILDING_COMPONENT as in Data-Table 5-2 on page 84)
- ➤ AFFECTED_BUILDING_COMPONENT (BUILDING_COMPONENT as in Data-Table 5-2)
- > AFFECTED BUILDING COMPONENT ID (BUILDING COMPONENT ID as in Data-Table 5-2)
- > RELATIONSHIP (RELATIONSHIP as in Data-Table 5-2)
- ➤ WRITING_DATE (from the computer clock)
- > DISCIPLINE TO (DESIGNING_DISCIPLINE of the affected BUILDING_COMPONENT as in DATA-TABLE 4-3 on page 62)
- ▶ DISCIPLINE FROM (DESIGNING_DISCIPLINE of the changed BUILDING_COMPONENT as in DATA-TABLE 4-3 on page 62)
- > SENT BECAUSE CHANGE ID (CHANGE_ID from DATA-TABLE 5-4 on page 99)
- > PREBUILT OR DYNAMICALLY ? (dynamically built)
- ➤ RULE WRITING DATE (WRITING_DATE as in Data-Table 5-2)
- > VIOLATED RULE ID (DYNAMICALLY_BUILT_RULES_ID form Data-Table 5-2)

A sample of part of these message elements is shown in Figure 5-5.

5.2.2.3 Deliver Design Change Messages

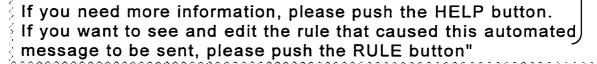
As the model is currently implemented, designers are expected to regularly check their message modules to find out about any design change messages. Sound and/or visual alert can be adopted to inform about the arrival of a new message. In any case, once a certain discipline starts the message module, the data in the Data-Table 5-3 that has the name of that discipline in the field "DISCIPLINE_TO" are automatically retrieved. From this data, messages are automatically built using a standard message structure that embeds variables. For example, the standard message that is sent as a result of a dynamically built rule is as follows:

"The attribute '", CHANGED_ATTRIBUTE, "' of the building component '", CHANGED_BUILDING_COMPONENT, " (", CHANGED_BUILDING_COMPONENT_ID, ")' has been changed from the value (", OLD_VALUE, ") to the value (", NEW_VALUE, ").

According to a dynamically built rule, such a change may affect the design of the attribute '", AFFECTED_ATTRIBUTE, "' of the building component '", AFFECTED_BUILDING_COMPONENT, " (", AFFECTED_BUILDING_COMPONENT_ID) ")'.

It is advisable to take some of the following steps:

- * Redesign the affected building component.
- * Negotiate the change with designing discipline.
- * Modify the rule that caused this message.



Semino

SOFTING

This standard message is shown to the designer (Figure 3-6 on page 44) after replacing the variables with values from Data-Table 5-3 as follows:

The attribute 'clear_height' of the building component 'Floor_Space (FS001)' has been changed from the value (6500) to the value (7600).

According to a dynamically built rule, such a change may affect the design of the attribute 'height' of the building component 'CMU_Backed_Brick_Veneer_Wall (EW001)'

It is advisable to take some of the following steps:

- * Redesign the affected building component.
- * Negotiate the change with designing discipline.
- * Modify the rule that caused this message.

If you need more information, please push the HELP button.

If you want to see and edit the rule that caused this automated message to be sent, please push the RULE button

The message as shown has the following characteristics:

- 1. It is sent only to the affected designers.
- 2. It is unambiguous, it defines exactly the attributes and the building components that have changed. It also informs about the old value of the attribute and the new value.
- 3. When the message is initiated through a prebuilt rule, it defines what are the possible effects of the design change on the affected building system.
- 4. When the message is initiated through a dynamically built rule, it defines only the attributes and the building components that are affected by the design change.
- 5. Finally, it provides advice on how to react to the design change.

The affected designer who receives the message is expected to take a corrective action in response to the design change. The message module allows the designer to review and modify the rules that cause messages to be sent, thus providing the model with the flexibility to adopt new relations that designers may develop among building components. Such capability is possible because of the link between Data-Table 5-3 (page 90) and both Data-Table 5-1 (page 84) and Data-Table 5-2 (page 84) (Data-Table 5-1 and Data-Table 5-2 are not linked). This link is through the field "VIOLATED _RULE_ID" in Data-Table 5-3 and each of the field "PREBUILT_RULE_ID" in Data-Table 5-1 and "DYNAMICALLY_BUILT_RULES_ID" in Data-Table 5-2.

5.2.3 USER INTERFACE

Two modules of the information model are used to realize the model capability to propagate design changes. The first module is the data manipulation module. Through this module, a designer records design decisions into the building-components-database. S/he also defines the dynamically built rules by accessing the relation-acquiring submodule. The second module is the message module where designers receive automated messages from the building components.

The interface of the data manipulation module starts by providing the designer with access to all projects that are currently under design so as to select one of them. When a designer selects a project, the data manipulation module is connected to the project-database of this project. The interface allows the designer to select a specific building

component and manipulate any instance of this component. The interface therefore provides access to all the building components that exist in the building-components-database. Because every project contains different building components, the interface has a customizable project menu that provides access to the various building components. The configuration module enables such customization in the project menu as shown in Figure 4-12 on page 77. Once a building component is selected, the designer has the option to create new instances of the component or work with existing ones (selection option [S1] in Figure 5-6). The designer is also able to delete the selected instance or modify the value of its attributes [S2]. Therefore the interface shows all the attributes of any selected building component as they are structured in the building-components-database. Figure 3-2 on page 38 shows a sample of the interface of the data manipulation module.

Once a designer changes the value of any attribute, the relation-acquiring sub-module starts. The interface for this module shows the building component, the instance, and the attribute that are currently under change. The interface provides the designer with the ability to select any attribute of any instance of any building component in the building-components-database which can have a relation with the component that is currently under design. The interface also provides the designer with the capability to build any relation between the components. It also allows multiple relations. Figure 5-4 on page 87 shows a sample of interface for the relation-acquiring sub-module.

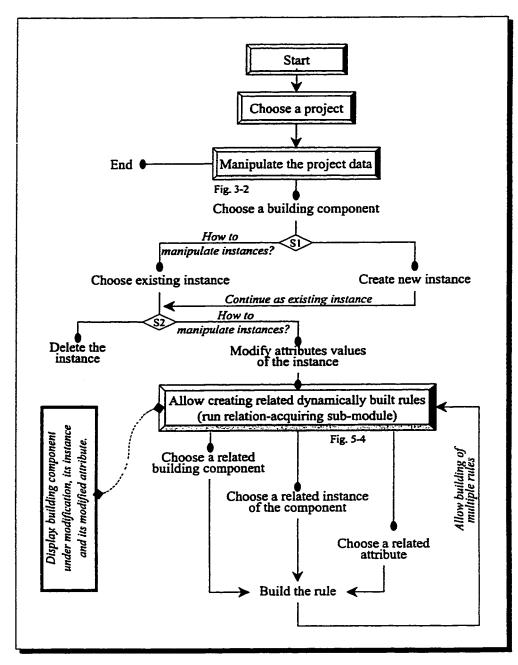


Figure 5-6 Diagram to illustrate user interface in data manipulation module¹.

¹ Please refer to the "Legend for User Interface Diagrams" on page XV.

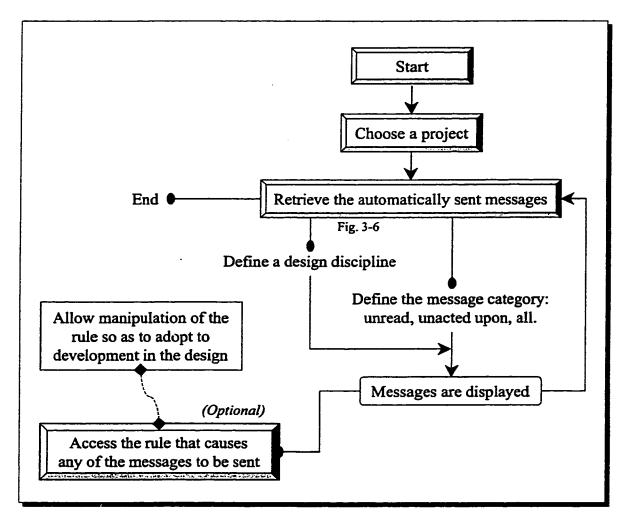


Figure 5-7 Diagram to illustrate user interface in message module.

The message module also starts by allowing the designer to choose between all the projects that are currently under design. The module is then connected to the relevant project database. The interface to retrieve the required messages is able to get only the messages that are related to a specific discipline. The designer can choose among retrieving all the messages since the start of the project, only the unread messages, or the messages that are read but have not yet been reacted to (unacted upon). The designer also has access to the rule that causes any message to be automatically sent. Such access allows the designer to modify the rule so as to be updated with current relations in the

project. The designer can make the building component ignore a certain rule if it is temporary or no longer applies to the design. Figure 3-6 on page 44 shows a sample interface for the message module.

5.3 TRACKING PAST DESIGN CHANGES

5.3.1 OBJECTIVES

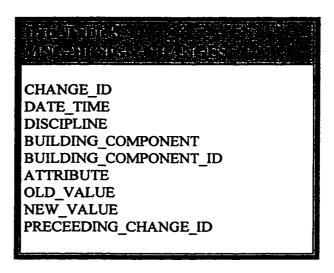
By tracking past design changes of a project, managers of design teams are provided with the capability to:

- 1. Observe and analyze the development of the detailed design.
- Evaluate the performance of the various designers in a project. Some designers make ad hoc design decisions which need to be later modified, costing the design team time and money.
- 3. Determine responsibilities for design incompatibilities in case of failure.

5.3.2 METHODOLOGY

The model keeps record of every design change that is initiated by any member of the design team. When a designer performs a design change, the changed building component automatically connects to a specially structured data-table in the management-database (Data-Table 5-4 on page 99). The component sends data elements about the design change that include the changed attribute, the building component, and the instance of this component. The component also sends the value of the changed attribute

before and after the design change. In addition, the component recognizes - from the computer clock – the day and time of the design change and records them in Data-Table 5-4. It also recognizes the discipline that is responsible for the change through contacting Data-Table 4-3 on page 62 and records the result in Data-Table 5-4. A sample of the data used in Data-Table 5-4 is shown in Figure 5-8.



			1000		Heldber		olekala)		
27	1997/06/29	11:14	arch	Floor_Space	FS001	clear_height	6500	7600	26
28	1997/06/29	11:29	enve	CMU_Backed_Brick	EW001	height	7500	8600	27

Figure 5-8 Sample of data in Data-Table 5-4 that describes some design changes and enables their tracking.

With the data-structure of Data-Table 5-4, the data-table can be queried according to the requirements of the design manager. The following is a sample query statement using SQL to extract data. The query obtains all the design changes that occur to the height of the instance "EW001" of the component "CMU_Backed_Brick_Veneer" during a one month period starting from June 20, 1997 at noon.

SELECT * FROM [MNG-DESIGN-CHANGES] WHERE building_component = "CMU_Backed_Brick_Veneer" AND building_component_ID = "EW001" AND attribute = "height" AND date_time >= "1997/06/20 12:00" AND date_time <= "1997/07/20 12:00"

The design manager needs not to build such a query. It is automatically and transparently built inside the tracking module. The design manager needs only to provide the query data through the interface with the tracking module (see Figure 3-8on page 47).

The management-database is developed with links between the design changes and the related automated message that are sent by the changed building component. This link is made through the attribute CHANGE_ID in Data-Table 5-4 which is referenced by the attribute SENT_BECAUSE_CHANGE_ID in Data-Table 5-3 on page 90. When the building component records the design change data in Data-Table 5-4, the data-table automatically assigns a CHANGE_ID to the design change. This CHANGE_ID is retrieved and utilized by the building component when recording the automated messages that result from that design change in Data-Table 5-3 (page 90). Through such a link, all the messages that are automatically sent by the changed building component are shown to the design manager (sample in Figure 3-10 on page 48). The messages are retrieved from Data-Table 5-3 using SQL. A sample for a utilized SQL statement is shown in the following zigzag frame. The statement extracts all the message elements that were sent because of the design change number 28.

SELECT * from MNG-MESSAGE-ELEMENTS WHERE SENT_BECAUSE_CHANGE_ID = 28

The design manager is also shown whether the designer who receives any of these messages has read it or not and whether he has taken a corrective action or not. Such capability helps design managers define responsibilities in case of failure in the compatibility of design information due to design changes.

5.3.3 USER INTERFACE

Tracking of past design changes is performed through the tracking module. The interface of the tracking module starts with allowing the design manager to select from all the projects that are under design (Figure 5-9). The tracking module is then connected to the appropriate project-database. The design manager can then define the constraints that limit the tracking of the change data according to his/her requirements. The first constraint, which is optional, is the time period for the design change. The design manager is then able to select between tracking all the design changes that are done by a certain design discipline or tracking the design changes of a certain component (selection [S1] in Figure 5-9). In case of selecting a specific building component, the design manager has the option to further limit the component to one of its instances and/or one of its attributes. Because every project contains different building components, the interface has a customizable project menu that provides access to the various building components. The configuration module provides the capability to make such customization in the project menu as shown in Figure 4-12 on page 77. Figure 3-8 on page 47 shows a sample interface of the tracking module. Once the constraints are defined, the required design changes are shown. For each change, the interface shows the

time, the discipline, the building component, the instance, the attribute, the old value, and the new value of the design change.

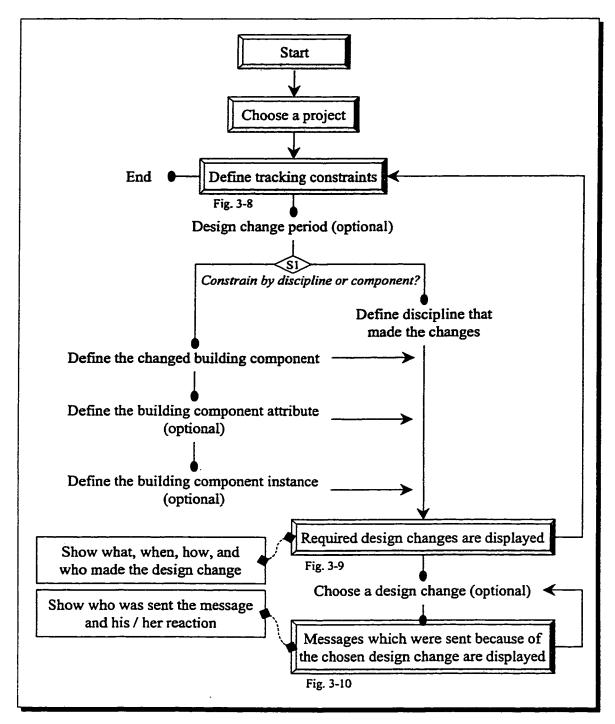


Figure 5-9 Diagram to illustrate user interface in tracking module.

The constraints that define the shown design changes appear to the design manager. Figure 3-9 on page 47 shows a sample of the interface for the results of a tracking query. The design manager can select one of the displayed design changes and track all the automated messages that are sent by the changed building component to the affected designers. This is achieved by double clicking on one of the shown design changes. Figure 3-10 on page 48 shows a sample of the interface that shows such messages. The interface provides the design manager with the total number of messages and enable him/her to browse through the messages. For each message, the design manager sees which discipline received the message, the text of the message, and whether the message was read and acted upon or not.

5.4 PLANNING AND SCHEDULING FUTURE DESIGN CHANGES

5.4.1 OBJECTIVES

As discussed in section 3.2.5 on page 48, some design changes have interrelated effects on several other building components in an extended fashion. Figure 5-10 on page 105 shows a sample of such changes (Figure 5-10 is a copy of Figure 3-11 on page 49 and is explained in section 3.2.5). Design managers need answers to some important questions before the design team is engaged in that type of changes. These questions are:

I. What are the interrelated design changes necessary to accommodate an initial design change?

- II. Are there alternative ways to accommodate the initial change? In the example in Figure 5-10, the HVAC designer can accommodate the change in two different ways. Each alternative, composed of a sequence of changes, is termed as a change-path.
- III. Are there design changes that may be repeated within a change-path and how to avoid these repetitions? In the example in Figure 5-10, the type of luminaires (L1) is redesigned several times.
- IV. Ultimately, how long will it take to fully accommodate the initial design change through a sequence of interrelated design changes or change-path? Are some of these changes critical to determine the total duration along a given change-path? These are important questions especially when the initial design change is requested by the owner. The design manager can then ask for a penalty-free extension for submitting the working drawings late.
- V. How many design man-hours are required to accommodate the initial change? The answer to this question can help determine additional design fees.
- VI. How much is the change expected to impact the construction costs of the building?

 Some interrelated design changes in the change-path may incur so much expenditure as to have the initial change reconsidered.
- VII. What level of recommendation is assigned to every change in a change-path by the

designer? A level of recommendation indicates whether or not a designer sees the change improving the building system that s/he is responsible for. It can also indicate that the change is impossible.

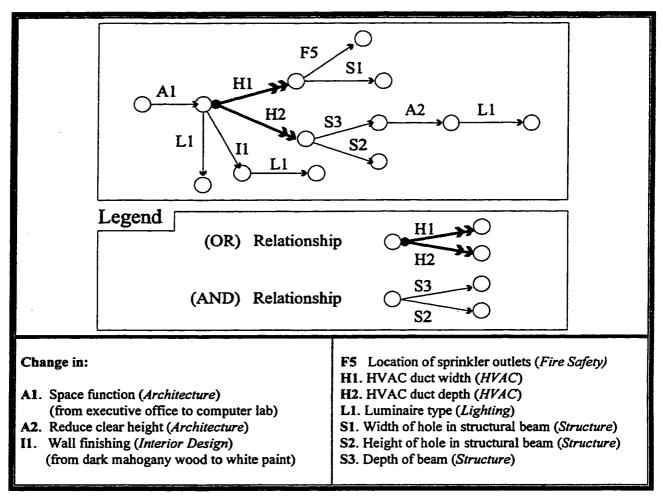


Figure 5-10 Scenario for interrelated design changes.

The objective of the planning and scheduling capability of the model is to provide answers to these questions. Consequently, an informed decision can be taken about whether or not the change is to be done and the best way to do it.

In traditional design environments, answers to these questions are not easy to obtain. Two

types of difficulty are identified. The first type is due to the lack of information. This is apparent when trying to answer question # I on page 103. It is hard for the design manager and for individual team members to identify all the disciplines that are affected by a specific change in a given building component. Change-paths as shown in Figure 5-10 are difficult to define because there is a scarcity of knowledge regarding how design changes are linked to one another. The second type of difficulty stems from the fact that some information exists but is scattered among various disciplines that work independently and are separated geographically, and there is no structured methodology to collect and organize this data.

5.4.2 METHODOLOGY

The methodology proposed uses the active components of the project-database in order to overcome these two types of difficulties. The methodology consists of four stages which are described below:

a) Collect data.

c) Schedule changes.

b) Organize changes.

d) Calculate costs and durations.

5.4.2.1 Collect Data

Collecting the data is accomplished through the *Virtual Change Module* of the information model (Figure 3-1 page 36). When interrelated design changes need to be studied, the designer who is responsible for the initial change starts a 'New Case' in the virtual change module. S/he defines what needs to be changed as shown in Figure 5-11

Ahmed Mokhtar, Ph.D Thesis 106

where the architect selects the attribute "used_for_activity" of the instance "space009". The architect changes the function of that space from an executive office to a computer lab as shown in Figure 5-12. Such a change is not actually done in the building-components-database; nevertheless, the linking knowledge embedded in the management-database (Data-Table 5-1 on page 84 and Data-Table 5-2 on page 84) is used to recognize the disciplines that are affected by the change. Automated messages are sent to these disciplines by populating a specially structured data-table (Data-Table 5-5 on page 109) with data about the virtual change. The elements of this message (called virtual message) are similar to those of the regular automated messages that are used in propagating design changes (Data-Table 5-3 on page 90). The only difference is in including a field for CASE_ID. The CASE_ID is automatically generated when a designer selects a new case (Figure 5-11). Its function is to link all the data and messages that belong to a group of interrelated virtual design changes in the various data-tables in the management-database.

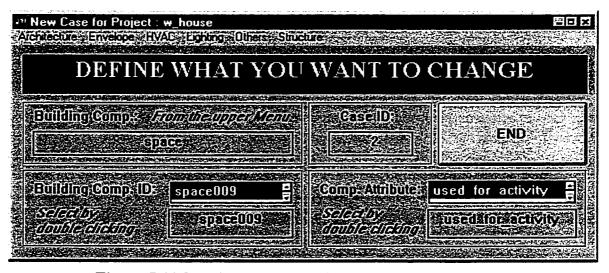


Figure 5-11 Starting a new case in virtual change module.

The affected designers receive the messages also in the virtual change module (Figure 5-13 on page 110). The messages notify the designers about the initial change and ask for a response. In Figure 5-13, a sample message to the HVAC designer would read as follows:

"In VIRTUAL MODE, The attribute 'used_for_activity' of the building component 'space (space009)' has been changed from the value (Executive Room) to the value (Computer Lab). Such a change may affect the design of the 'HVAC' system. Please indicate the expected modifications in the building components."

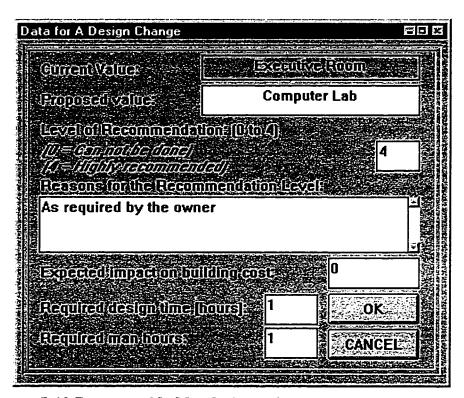


Figure 5-12 Data provided by designer for proposed design change.

Each affected discipline defines what components in its system need to be changed and also how they are to be changed. For example, the HVAC engineer, using his/her design experience, defines that the change in the space activity requires a minimum 30% increase in the supply duct size. S/he can do that in two ways: either by increasing the

width (dimension_x) of the duct from 600 mm to 800 mm (Figure 5-13 and Figure 5-14) or by changing the duct depth from 400 mm to 550 mm. Once the HVAC designer specifies the two possibilities, automated messages are sent to the disciplines that are affected by each of these changes. The affected disciplines read the messages and react to them in similar fashion.

ASIMORIO ECESTA (22 SILVET PROPERTY (49) ATM ID CHANGED_ATTRIBUTE CHANGED_BUILDING_COMPONENT CHANGED_BUILDING_COMPONENT_ID OLD VALUE NEW VALUE AFFECTED ATTRIBUTE AFFECTED_BUILDING_COMPONENT AFFECTED BUILDING COMPONENT ID RELATIONSHIP WRITING DATE BEEN READ ? BEEN ACTED UPON? DISCIPLINE TO DISCIPLINE FROM SENT BECAUSE CHANGE ID PREBUILT_OR_DYNAMICALLY_? RULE_WRITING_DATE VIOLATED_RULE_ID CASE ID CAUSE OF THE EFFECT

The data specifying the sequence of changes and the alternatives are collected in a specially structured data-table (Data-Table 5-6 on page 111). Each design change is automatically assigned a code (e.g. A1, H1) that is used in the data-table. The details of

the design change are stored in Data-Table 5-7 on page 112. Both Data-Table 5-6 and Data-Table 5-7 are linked through the design change code. A sample of the collected data is illustrated in Figure 5-15 on page 111 which represents the sequence of design changes that appear in Figure 5-10 on page 105.

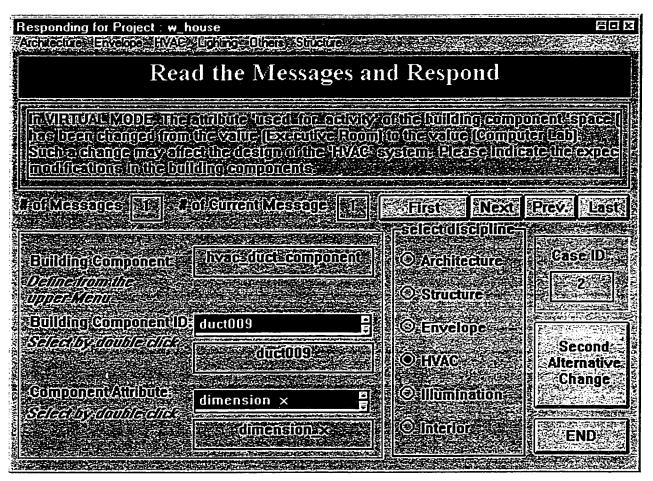


Figure 5-13 Response to virtual change message in virtual change module.

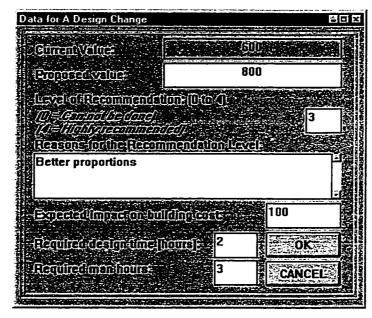
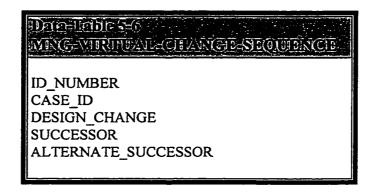


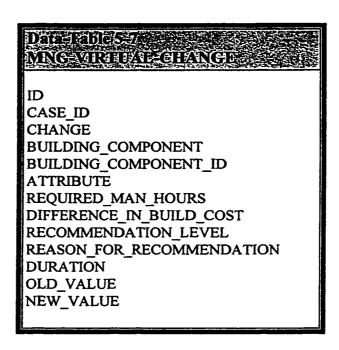
Figure 5-14 Data provided by designer for proposed design change.



	लेक्ट पिट ह	(CENTE PER ANTICE	La Control Control	ं विकास है अपस्य प्रकार
116	2	NULL	A1	
117	2	Al	HI	H2
118	2	Al	Ll	
119	2	Al	II.	
120	2	11	Ll	
121	2	H2	S3	
122	2	S3	A2	
123	2	HI	F5	
124	2	H1	SI	
125	2	A2	Ll	
126	2	H2	S2	

Figure 5-15 Sample of data in Data-Table 5-6 that defines sequence of changes.

Additional data is also collected from designers during the process. As Figure 5-14 shows, when the HVAC designer specifies a change in the duct width, the system requires him/her to answer questions about the actual design time required to make the changes; the additional man-hours needed; the expected difference on building cost when the change is done; and the level of recommendation for that change. This data is saved in Data-Table 5-7 and a sample of the data is shown in Figure 5-16. The process continues till all related changes are addressed and their data are collected.



	Ī			boldingstor	building	e contribute and	12.11	differe	100	reason for it	Qq.	Secold values.	DE NEVEL YEAR
4	ŀ	2	Al	space	space009	used_for_activity	1	0	4	As required	1	executive office	computer
- 5	ī	2	Hl	Hvac_duct_c	duct009	Dimension_x	3	100	3	Better propor	2	600	800

Figure 5-16 Sample of data in Data-Table 5-7 which is collected from designers through virtual change module.

5.4.2.2 Organize Changes

After data for the interrelated design changes is collected, two problems need to be

addressed. The first problem is to define the number of possible change-paths as well as the actual design changes that compose each of them. A sub-routine is developed to provide the computer with the logic that can address this problem. The sub-routine, which is described in details in section B.1 (page 179), functions on the data stored in Data-Table 5-6 (page 111). It stores the initial design change as an item in the first change-path along with all its successor changes. When there are alternate successors ("OR" arrows in Figure 5-10), it considers first only one of them. For the first alternative, all interrelated design changes are defined and grouped together to constitute the first change-path.

Daetribert Micentribust Chargentaining
ID DESIGN_CHANGE SUCCESSOR
ALTERNATIVE# CASE_ID

MESON OF STREET	STATE OF CHANGE SERVICE	A Successor	National Velocity	
4734	NULL	Al	ALTERNATIVE#1	2
4735	Al	Hl	ALTERNATIVE#1	2
4736	Al	Ll	ALTERNATIVE#1	2
4737	Al	11	ALTERNATIVE#1	2
4738	Hl	F5	ALTERNATIVE#1	2
4739	HI	S1	ALTERNATIVE#1	2
4740	II	L1	ALTERNATIVE#1	2
4741	NULL	Al	ALTERNATIVE#2	2
4742	Al	H2	ALTERNATIVE#2	2
4743	Al	Ll	ALTERNATIVE#2	2
4744	A1	11	ALTERNATIVE#2	, 2
4745	H2	S 3	ALTERNATIVE#2	2
4746	H2	S2	ALTERNATIVE#2	2
4747	II	Ll	ALTERNATIVE#2	2
4748	S3	A2	ALTERNATIVE#2	2
4749	A2	Li	ALTERNATIVE#2	2

Figure 5-17 Sample of data in Data-Table 5-8 that defines alternative change-paths.

The sub-routine then backtracks along the first change-path till an alternative change is encountered (sample is highlighted in Figure 5-15). A new change-path is then generated using the second design alternative (alternate_successor in Figure 5-15 on page 111) along with its successors. The process continues until all change-paths are generated. The result is stored in a specially structured data-table in the management-database (Data-Table 5-8). A sample of the result is illustrated in Figure 5-17 where two change paths (Alternative #1 and Alternative #2) are identified for the sample data in Figure 5-15.

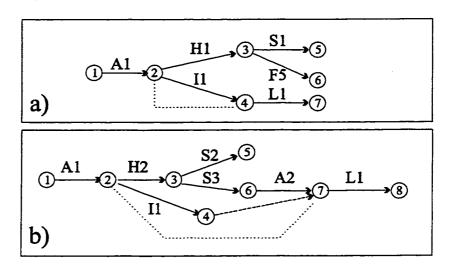


Figure 5-18 Alternate change-paths for scenario in Figure 5-10

The second problem is to define a logical sequence of the design changes in every change-path. As Figure 5-10 shows, some building components may need to be repeatedly redesigned because they are affected by changes in several other building components (for example, the type of luminaire). It is clearly inefficient to have the lighting designer change the type of luminaire each time a change occurs in a building component. It is therefore important to involve the lighting designer only after all these changes are done. Another sub-routine is therefore developed to determine the proper

"location" of such changes so as to minimize redesigning. The sub-routine, which is described in details in section B.2 on page-182, functions on the data stored in Data-Table 5-8. It delays the positioning of the repeated design changes in a change-path till all preceding design changes have taken their own positions. The subroutine also prepares the design change data to be scheduled using a standard critical path algorithm.

Figure 5-18 shows (in conceptual form that is not actually used in the computer) the result of applying the two developed sub-routine to the scenario in Figure 5-10 where two alternate change-paths are generated. The lower portion of Figure 5-18 shows that the change "L1" is located in the flow diagram only in one position that comes after changes A1, I1, and H2-S3-A2.

5.4.2.3 Schedule Changes

Once the change-paths are identified and the sequence of changes is defined for each of them, a bar chart is generated for each path as in Figure 5-19 using the expected duration of individual design changes (collected in the first stage in section 5.4.2.1 on page 106 and stored in Data-Table 5-7 on page 112). Critical changes are also identified using a standard critical path method (CPM) algorithm (Paulson 1995). Figure 5-19 shows the critical design change activities in darker tone.

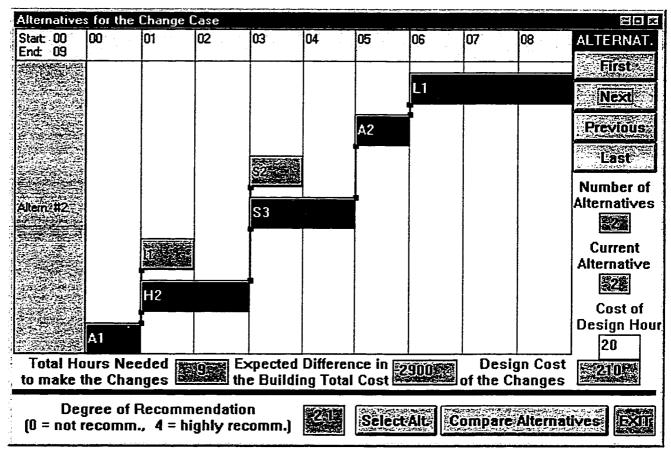


Figure 5-19 Bar chart for alternate change-path (b) in Figure 5-18

5.4.2.4 Calculate Costs and Durations

Using data collected in Data-Table 5-7 (page 112) and shown in Figure 5-20 (page 118), the expected impact on the building construction costs ($C\tau$) is calculated for each changepath. It is the sum of the positive and negative effects of all design changes on the building cost. The following equation is used:

$$C_T = \sum_{x=1}^{n} C_x$$
 where

n is the total number of design changes in a change path.

x is a design change.

C_x is the value of the attribute "DIFFERENCE_IN_BUILD_COST" in Data-Table 5-7.

For example, the change-path shown in Figure 5-18b on page 114 would result in building construction cost savings of \$2900 as indicated in Figure 5-19. Also calculated is the level of recommendation for each change-path (*Rt*). It is the average of the recommendation levels assigned to all the changes in that path. If one of the recommendation level values of a design change path equals zero, it means that the designer of that change indicates that such a change is not possible to be done. Reasons can include violation of code. In such case, the whole change path is given the recommendation level of zero which tells the design manager that such an alternative can not be used. The recommendation level has a value of 2.1 in Figure 5-19. This information provides the design manager with the ability to choose the best change-path from the design point of view. The following equation is used:

$$RT = \left(\sum_{x=1}^{n} R_x\right)/n$$
 (if none of R_x equals zero) or $RT = 0$ (if any of R_x equals zero) where n is the total number of design changes in a change path.

x is a design change.

 R_x is the value of the attribute "RECOMMENDATION_LEVEL" in Data-Table 5-7.

Other calculations are also performed for each change-path, such as the additional design time required to make the changes $(T\tau)$ (9 hours in Figure 5-19) using the equation:

$$TT = \sum_{x=1}^{n} T_x$$
 where

n is the total number of critical design changes in a change path.

x is a design change.

 T_x is the value of the attribute "DURATION" in Data-Table 5-7. and the corresponding design cost $(D\tau)$ (\$210 in Figure 5-19) using the equation:

$$D\tau = (\sum_{x=1}^{n} M_x) \cdot h$$
 where

n is the total number of design changes in a change path.

h is the cost of man hour in the design firm.

x is a design change.

Mr is the value of the attribute "REQUIRED_MAN_HOURS" in Data-Table 5-7.

If the initial design change is requested by the owner, this information would substantiate the design manager request for extra design fees and time so as to accommodate the initial change. If the owner agrees, the initial design change can be accomplished through the best "design path" by switching from virtual to actual design mode. If the calculated impact of the initial design change is minimal, the design manager can proceed to actual change mode without delay.

LEADING!	entities in the contractions	direce conduit of conte	TESTIMENT TO THE STATE OF THE S	(៣) ជាក្រាក
Al	i	0	4	I
A2	1	0	1	1
I1	ı	-3000	2	1
H2	3	100	1	2
Ll	0.5	-200	4	3
S2	2	0	2	I
S3	2	200	l	2

Figure 5-20 Partial data from Data-Table 5-7 that is used for calculating costs and durations.

5.4.3 USER INTERFACE

Two modules are used for the planning and scheduling of design changes. First is the virtual change module which interfaces with designers to collect the data. Second is the planning and scheduling module which shows the results to the design manager. The interface for the virtual change module (Figure 5-21 on page 121) starts by allowing the designer to choose among the projects that are currently under design. The designer has the option to start a new change case or to respond to a virtual change message (selection [S1] in Figure 5-21). In the event of starting a new change case, the designer selects the building component, the instance, and the attribute that s/he intends to change. A sample interface design is shown in Figure 5-11 on page 107. Because every project contains different building components, the interface has a customizable project menu that provides access to the various building components. The configuration module has the capability to make such customization in the project menu as shown in Figure 4-12 on page 77. Once the designer defines the changed building component, s/he is asked to provide the data about the design change. A sample interface design for collecting the data is shown in Figure 5-14 on page 111. To reduce the time needed to fill this data, the interface provides the designer with any historic data about the same design change. Figure 5-22 on page 122 shows a sample of the interface for such historical data. The designer can directly use these data or modify some of them.

In the event of responding to a virtual change message, the interface shows the total number of virtual messages that are sent to a specific discipline. The designer is able to browse through these messages, read each one and respond to it separately. To enable the designer to respond, s/he has access to any building component, any instance, and any attribute of the component. Figure 5-13 on page 110 shows a sample of that interface. Similar to the event of starting a new case, the designer is asked to provide data about the design change. The historical data is also shown to help the designer. The interface also allows the designer to provide a second design change alternative for the same message.

The interface for the planning and scheduling module starts by allowing the design manager to select one of the projects that are currently under design (Figure 5-23 on page 122). S/he selects among the available change cases in the specified project. The interface then shows the results of the change case. The total number of the alternate change paths is shown. The design manager can browse among the various change paths. The results of each change path is shown separately. Figure 5-19 on page 116 shows a sample interface of the results. The bar chart shows every design change in a coded format. The design manager is able to know the details of every design change. This is achieved by clicking on the relevant bar in the bar chart. The interface then shows all the data about the selected design change. Figure 5-24 on page 123 shows a sample of the interface to present the data about a design change. The interface allows the design manager to compare among the various alternatives by showing the results of all the alternatives in a table format (Figure 5-25 on page 123).

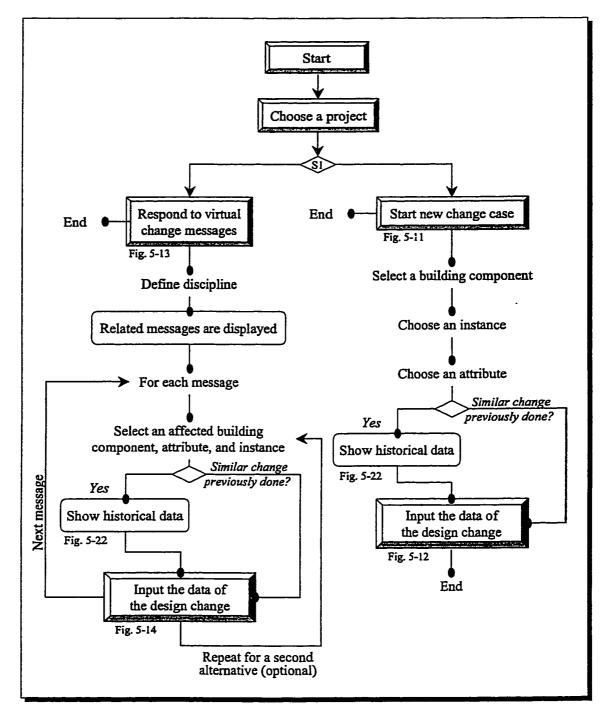


Figure 5-21 Diagram to illustrate user interface in virtual change module.

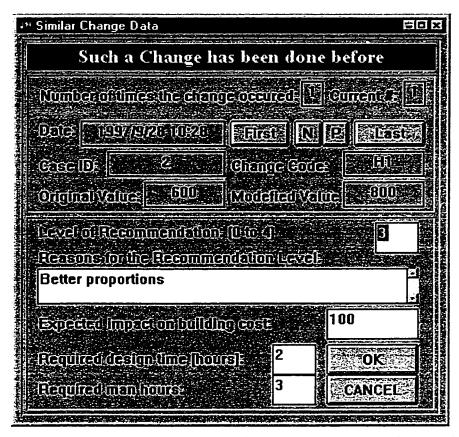


Figure 5-22 Interface to show historical data about certain design change.

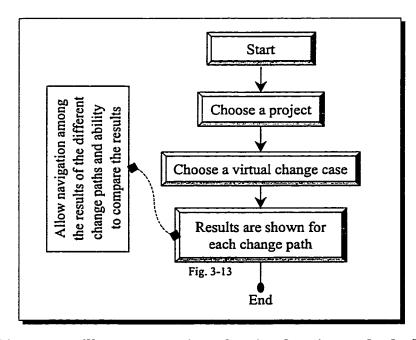


Figure 5-23 Diagram to illustrate user interface in planning and scheduling module.

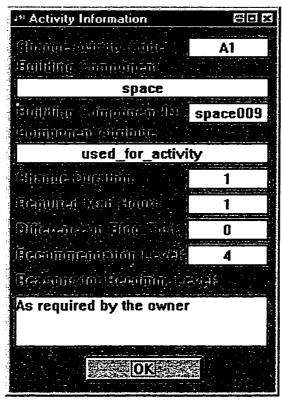


Figure 5-24 Interface to show detailed data of a design change.

विस्मिट्सेश्ट	Greatonelse Destro Greater Greater	: अंग्रास्ट : जी : हिल्लोगामसार्वेः	්රීල්ල්ල්ල්ල්ල්ල්ල්ල්ල්ල්ල්ල්ල්ල්ල්ල්ල්ල	ම්ගියෙකුවේ ග මිග්ගිගුව මුලුල්
Altern. #1	170	2.8	5	-3100
Alem 72	210	21	9.55	-22:111)
y a				

Figure 5-25 Interface to show results of all alternative change-paths so as to enable comparing them.

5.5 SUMMARY

The concept of active building components enables the model to manage design changes in terms of propagating current design changes, tracking past design changes, and planning and scheduling future ones. The building components are made active by providing them with the necessary linking knowledge. The acquisition of this knowledge becomes possible after reversing the knowledge acquisition process. The management-database plays a vital role in supporting the model ability to manage design changes. It sustains the linking knowledge in term of rules elements, records the data regarding every design change, carries the message sent by the building components, and maintains the data needed to plan and schedule future design changes. A methodology that included two specially developed subroutines is introduced to encounter the challenges of interrelated design changes. It makes use of the linking knowledge and the active building components to collect needed data from the designers. It then provides the design manager with clear idea on how such a design change can be done.

VALIDATION OF THE MODEL

6.1 Introduction

This chapter aims at demonstrating that the developed model, as described in previous chapters and implemented, can achieve its stated objectives. The first part of this chapter describes the environment in which the model is implemented in terms of software and hardware. The second part focuses on validating the model through three specific levels of validation.

6.2 COMPUTING ENVIRONMENT

The information model is implemented on a client/server network. NOVEL-NETWARE (1993) is used as the network management software while MS-WINDOWS-95 (1995) is used as the local operating system. The central project-database is implemented in MS-ACCESS (1994). It is installed on the server, hence accessible by all users. The modules in the designer system and the design manager system (see Figure 3-1 on page 36) are implemented in the software environment LEVEL5 OBJECT (1994). Depending on whether the user is a designer or a design manager, his/her own (client) computer contains the relevant modules. The modules - on the client computers - are connected with the central project-database on the server through the Object Database Connectivity (ODBC) component that is part of the MS-WINDOWS-95 system. This component enables the modules to use SQL in order to manipulate the project-database.

Hardware for the client computers consists of four INTELTM based personal computers with 166 MHz Pentium processors and 16 MB of RAM. The server is also an INTELTM based personal computer but with 200 MHz Pentium processors and 64 MB of RAM. These are common type of computers that are used by building design professionals. The computers are located in the same computer lab at the School for Building, Concordia University. Yet, their positions are selected so as none of the users can see the others. This is to emulate the actual design environment where designers of various disciplines are separated spatially.

6.3 VALIDATION

6.3.1 APPROACH

The nature of the developed model makes it unfeasible to compare its outputs with experimental data or data from literature. However, it is important to make sure that the model, as developed, functions properly and achieves its objectives. Therefore, the model is subjected to three levels of validation. The first level aims to show that the model is properly coded on the implementation software and in accordance with the model development. The second level demonstrates that the model provides the capabilities that are expected from its development. The third level establishes that the model improves the current practices of dealing with design changes and can solve their related problems.

6.3.2 FIRST LEVEL OF VALIDATION

This level of validation aims to ensure that the model components are coded according to the described development in chapters four and five. Among numerous tasks in this level, the validation includes making sure that the input data are used correctly, the program routines function accurately, the modules are properly connected to the database, and that the expected output data is presented. Several tools that exist within the development software (LEVEL5 OBJECT) are used in the process. The automated syntax validation ensures that any saved program code in syntactically correct even before compiling the code. The "session monitor" tool allows the developer to dynamically follow the program flow step by step and discover any logical error. The "history" tool is also used to print out all the program flow in a textual format (Figure 6-1), hence it can be reviewed for detecting any source of logical error. Several samples of data are used to validate each subroutine in order to make sure that the code is trouble-free.

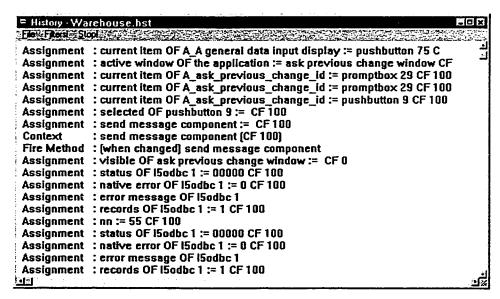


Figure 6-1 Sample of statements recorded by "history" tool in LEVEL5 OBJECT.

6.3.3 SECOND LEVEL OF VALIDATION

This level of validation aims to ensure that the components successfully function together and provide the user with the capabilities that are expected from the model. A setup that simulates a multi-disciplinary design environment is formed for this purpose. In this setup, graduate students who have professional experience represent four design disciplines. These disciplines are architecture, structure, HVAC, and envelope. A graduate student is assigned the role of the design manager. Data is extracted to populate the various parts of the model. The first part which needs to be populated with data is the general-database (see Figure 3-1 on page 36). Two references are used for this purpose "Architectural Graphic Standards" (Ramsey and Sleeper 1994) and "Building Construction Illustrated" (Ching 1991). From these references, a number of building components are extracted along with the attributes that describe each of them. Most of the extracted building components are grouped together to form building systems. Prebuilt rules are also added to many of the building components. With such data, the generaldatabase is ready to be used in assembling the data-structure of the building-componentsdatabase for a variety of building types. The second part that needs to be populated with data is the central project-database. A warehouse project (DeChiara 1980) is selected for this purpose. The warehouse is 50,292mm (165') by 36,678mm (120' 4"). It consists of two main zones, a storage zone that is 6,198mm (20° 4") high and an administration zone that is 3,327mm (10` 11") high.

¹ Project: Sales office and Warehouse for W.W.Grainger, Troy, Mich. Architect: Gillett Associates, Inc.

The difference in height is used as room for mechanical equipment. The administrative zone consists of a general office area, entrance and a sales room, a lunchroom, and two rest rooms.

The structure of the warehouse consists of steel columns that support open web steel joists. The columns are located on non-equally spaced grid. The depth of the open web steel joists that support the building main roof is 457mm (18") while those supporting the floor above the administration zone — where the mechanical equipment are installed - is 356mm (14") in depth. The roof is 38mm (1.5") corrugated steel deck covered with 51mm (2") rigid insulation while the floor above the administrative zone is open web steel joists that carry 51mm (2") concrete on 38mm (1.5") corrugated steel deck. Acoustical ceiling is used in the administrative zone and is suspended from the joists. The external walls are mainly made of CMU_Backed_Brick_Veneer with 38mm (1.5") of rigid insulation. They are supported by reinforced concrete girders. The floor is 127mm (5") concrete slab over sand and gravel fill.

With the use of the data from the warehouse project, the following sections illustrate the validation of the various capabilities of the model. These capabilities are categorized in similar fashion as the model development which is presented in chapters four and five.

6.3.3.1 Assembly and Population of the Project-Database

The first step in using the model is to assemble the data-structure of the buildingcomponents-database so that it can accept the design data of the warehouse. The design

manager performs this task using the configuration module and with the help of the general-database. He starts the process by creating a new project and uses the building systems option to assemble the data-structure (sample is in Figure 6-2). The majority of the building components are assembled through this option. The assembly process automatically adds the relevant prebuilt rules as stored in the general-database. The remaining building components are individually obtained from the general-database (sample is in Figure 6-3) or created from scratch. Each designer reviews the datastructure of the building components that are under his responsibility and may ask the design manager to modify some of these components by adding or editing attributes. For example, the concrete column component that is part of the structural system in the general-database (Figure 6-2) is removed and a steel column component is added (Figure 6-4). The design manager asks each designer about the elements of data which are used as input to the design of his system. Accordingly, the design manager adjusts the automatically added prebuilt rules and creates new ones (samples are in Figure 6-5 to Figure 6-8). After completing the model configuration, each designer uses his own data manipulation module to populate the building-components-database by creating instances of the building components under his responsibility. Occasionally, the designers through the use of the relation-acquiring sub-module create dynamically built rules (sample is in Figure 6-9).

The process proves that the model is capable of flexibly adapting to the data requirements of the project. The initial assembly process of the data-structure takes about one hour for

the warehouse project while adjusting the prebuilt rules take about half an hour for each discipline. This is seen as a reasonable overhead time for using the model. The input of graphical data in form of textual data is difficult. The use of a CADD tool would be useful for this purpose but remains beyond the scope of this project, given time constraints.

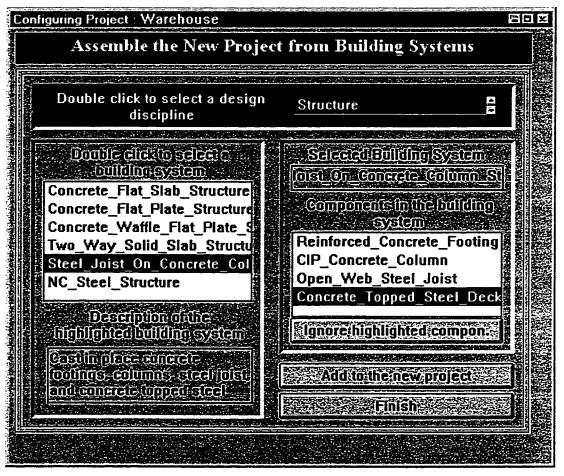


Figure 6-2 Assembly of building components in building-components-database from building systems available in general-database.

H.		e a x
Com	ponents in the General Database	
	ාරල ලැල් වෙල නිල්සේ ව ලංගා නාලයේ	
Fan_Coil		-13
Floor		=
Floor_Space		
Hole_in_CIP	_Concrete_beam	
डिटान्डट १९०० लाम	Final Spice	
	Floor Space id	-
Malmer dille	clear height	
ලාබානයේ(-	floor to floor height	\(\begin{array}{c}
	under_ceiling	- 4
THE STREET	Component	
Add the C		
Service Control	omponence monester (austendankula)	

Figure 6-3 Adding building component from general-database to buildingcomponent-database.

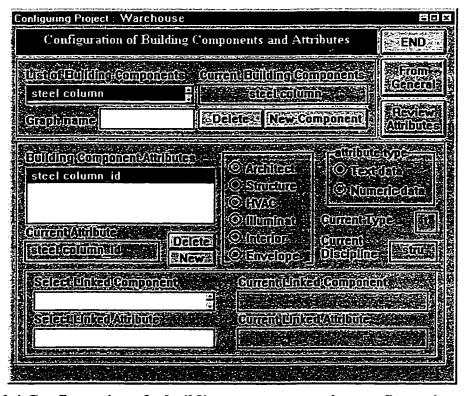


Figure 6-4 Configuration of a building component using configuration module.

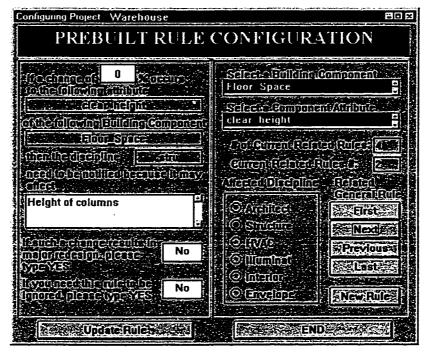


Figure 6-5 Input data to structural design translated to prebuilt rule.

(If any change occurs to clear height of floor space then the structural engineer needs to be notified because it can affect the height of columns).

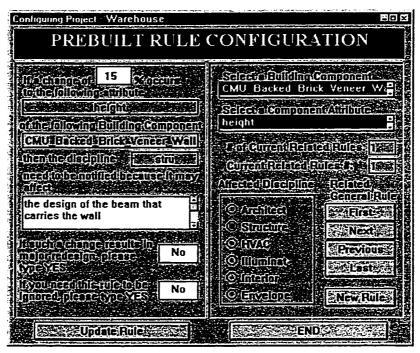


Figure 6-6 Input data to structural design translated to prebuilt rule.

(If a change more than or equal 15% occurs to height of CMU backed brick veneer wall then the structural engineer needs to be notified because it can affect the design of beam that carries the wall)

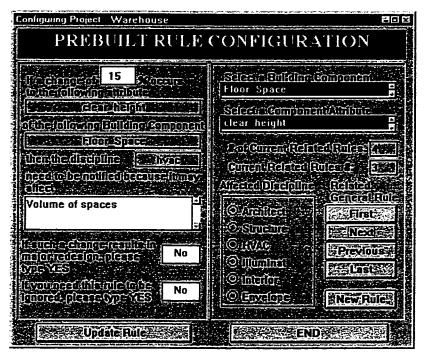


Figure 6-7 Input data to HVAC design translated to prebuilt rule.

(If a change more than or equal 15% occurs to clear height of the floor space then the HVAC engineer needs to be notified because it affect the volume of the conditioned space)

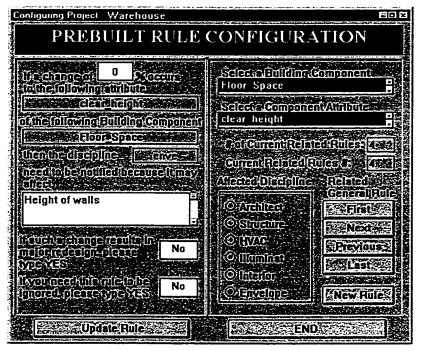


Figure 6-8 Input data to envelope design translated to prebuilt rule.

(If any change occurs to clear height of floor space then the envelope engineer needs to be notified because it affects the height of the wall)

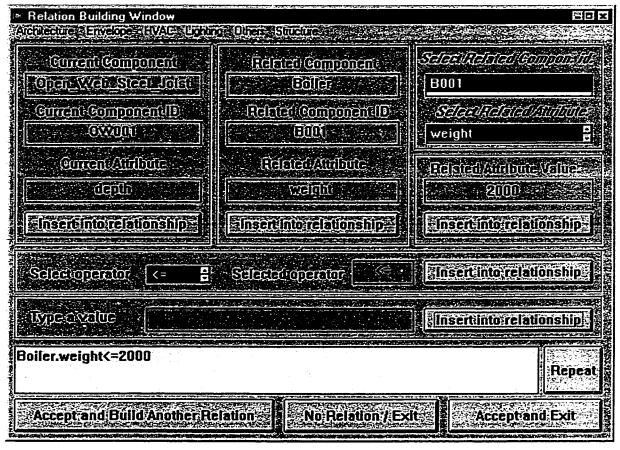


Figure 6-9 Dynamically built rule acquired from structural designer defining one of the conditions used to design depth of Open Web Steel Joist.

(The depth of the open web steel joist Ow001 is determined based on the information that the weight of the boiler above it is less than or equal two tons)

6.3.3.2 Propagating Current Design Changes

To validate the model capability to propagate current design changes, one designer (the architect) introduces a random design change. The design change is to increase the height of the storage zone by 1.2m (4') to satisfy a modification in the owner requirements. Consequently, the changed component automatically sends messages to the affected designers. The structural engineer receives a message informing about the change in height and suggesting that columns design may be affected (Figure 6-10). Similarly, a message is sent to the HVAC designer stating that the HVAC system might be affected

due to the change in the space volume (Figure 6-11). Another message is sent to the envelope designer indicating that the height of the walls is now incompatible with current height of the storage space (Figure 6-12). The structural engineer reads the message, evaluates the design change and only modifies the length of the columns. The HVAC designer reads the message, evaluates the capacity of the Boiler and specifies a larger type. He also resizes the HVAC ducts that supply the storage zone. The envelope designer reads the message and increases the height of the walls. Other automated messages are sent following these design changes. The structural engineer receives two messages. The first informs that the height of the walls has increased, which can affect the design of the beams that carry these walls (Figure 6-13). The second message informs that a heavier Boiler is used, which may affect the design of the floor that carries this equipment (Figure 6-14). The structural engineer re-evaluates the design of the beams that carry the walls and increases their dimensions. He also re-evaluates the design of the columns and the open web steel joists that carry the Boiler and finds these to be safe.

As this scenario illustrates, building components using the captured linking knowledge proves to be capable of adequately propagating design changes and only to the interested designers. The designers are able to perform necessary design modifications in the design information so as to ensure compatibility. The likelihood of errors is greatly reduced.

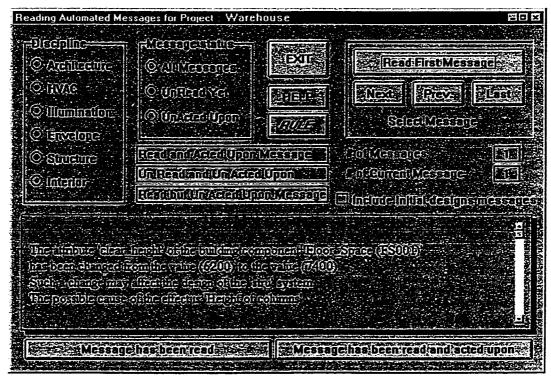


Figure 6-10 Message received by structural designer informing of design change made by architect.

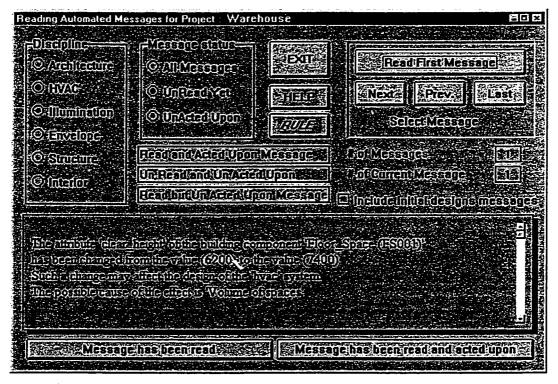


Figure 6-11 Message received by HVAC designer informing of design change made by architect.

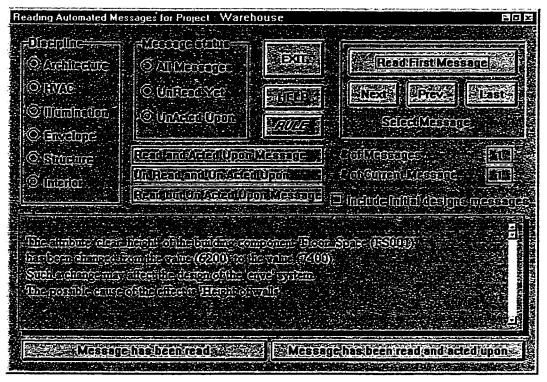


Figure 6-12 Message received by envelop designer informing of design change made by architect.

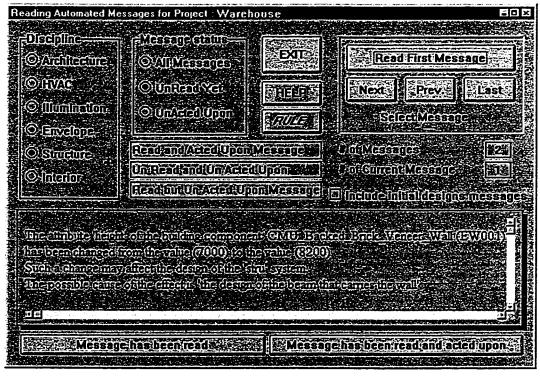


Figure 6-13 Message received by structural designer informing of design change made by envelope designer.

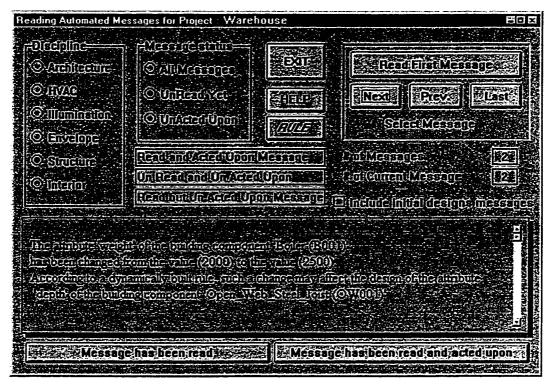


Figure 6-14 Message received by structural designer informing of design change made by HVAC designer.

6.3.3.3 Tracking Past Design Changes

To validate the model capability to track past design changes, the design manager inquires about some design changes which occurred during the scenario in section 6.3.3.2. A sample of these inquiries is about the design changes that occur to the weight of the building component "Boiler". As a result of this inquiry, the change in weight of the boiler appears to the design manager (Figure 6-15). As the design manager double clicks on the design change, the message that is automatically sent to the structural engineer due to this design change appears (Figure 6-16). From the performed inquiries, the model always performed correct tracking of past design changes and their related messages. Hence, its development provides the capability of tracking past design changes.

Change-Monitoring Module		Bex		
QUERY RESULTS				
Desiglar Euldag-Component Euler	Gimenerall UNI	ComponentAndligie 		
Vince Cairot Francial de Carrotine OS	मिलाह रिक			
Ben componen Comple	Ødime.	TIC NAC Velue Velue C		
िएटल जनांक जनां चेत्रदेल जनांक जनां	ാർത്ത് നലത്ത്	् युक्ता (युक्ता व्यक्ता (
e Egglie system op aregnise Egglie system op aregnise	o acido ento macante entr	COK		

Figure 6-15 Results of request made by design manager to track design changes made to a building component.

Change Monitoring Module			
RELATED MES	SAGES		
Discipling which receives the message stru	RendalisaMessing		
Read and Acted Upon Message (Black Color)	Next Prev Last		
Un Read and Un Acted Upon Message (Green)	Secomesone		
Read but Un Acted Upon Message (Red)			
Converge Fi-Conveniverand i	OK		
an Action of the pull think embers in Active 7.44 to Second the pull think embers in Active Congression for pear equilibrity to an including embotics for the configure for the configure of the	anemayalisediedssipok		

Figure 6-16 Results of request made by design manager to track automated messages that are sent because of certain design change.

6.3.3.4 Planning and Scheduling of Future Changes

The aim of this section is to validate the methodology which provides the model with its capability to plan and schedule future design changes. A hypothetical scenario for extended design changes is developed for this purpose (Figure 6-17). The scenario intends to include several design change alternatives that relate to each other differently. Some of these design change alternatives are dependent on the choice in a previous alternative. For example, design change alternative [(A5) or (A6)] is feasible only if the design change (H2) is chosen out of design change alternative [(H1) or (H2)]. On the other hand, design change alternative [(A5) or (A6)] is independent of design change alternative [(H3) or (H5)], yet they coexist in parallel. The scenario also includes repeated design change alternatives like [(H3) or (H5)]. The purpose for developing the scenario with these features is to make sure that the specially developed subroutines (section 5.4.2.2) function correctly in all possible scenarios.

The virtual change module is used to initiate a new case (similar to Figure 5-11 on page 107) and to collect the data about all the design changes in the developed scenario. The planning and scheduling module is then run to generate the various possible change-paths for the scenario. Figure 6-18 to Figure 6-23 present the result as produced by the planning and scheduling module. The model is able to recognize all the possible change-paths and to calculate the needed design time, design cost, impact on building cost and the level of recommendation for every change-path. In change-paths number three (Figure 6-20) and number five (Figure 6-22), the levels of recommendation are calculated with the result of

the value zero. This indicates to the design manager that both of these change-paths can not be used. It is also shown that the reason is design change (A5), that is change in the slab to slab height. The architect assigns a level of recommendation of zero to this design change because, if it is pursued, will result in exceeding the maximum permissible building height according to the code.

The module also generates a comparison table among all the alternate change-paths as shown in Figure 6-24 to help the design manager in selecting the best change-path. All the generated bar charts and the calculations are identical to those performed manually for the scenario, only much faster. As such, the model proves to be capable of providing the expected planning and scheduling capability.

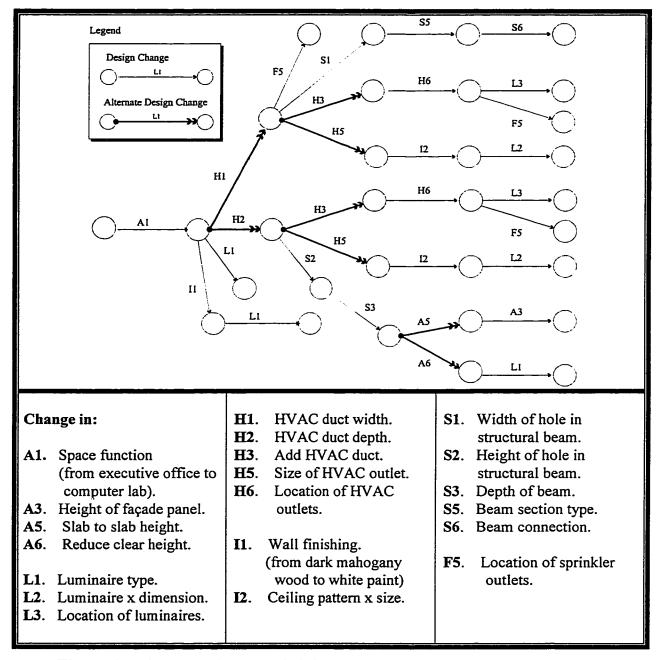


Figure 6-17 Scenario for extended design changes with several alternatives.

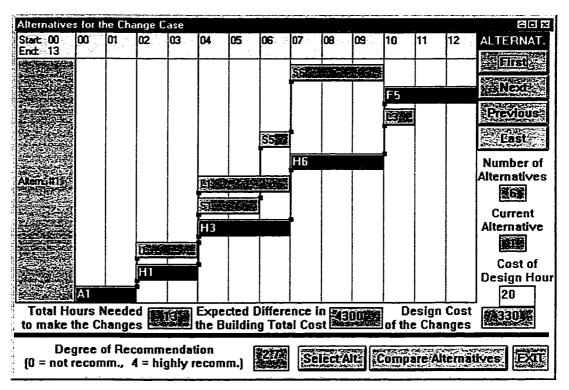


Figure 6-18 Bar chart and data for alternate change-path (1) for the scenario in Figure 6-17

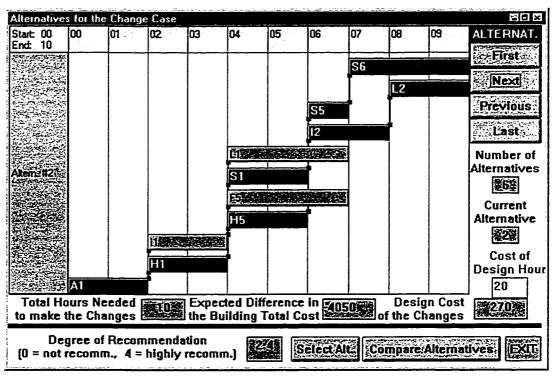


Figure 6-19 Bar chart and data for alternate change-path (2) for the scenario in Figure 6-17

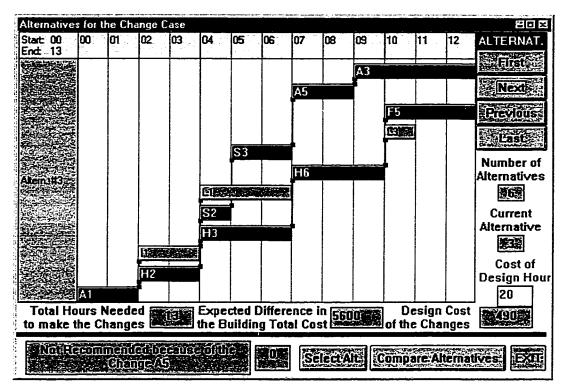


Figure 6-20 Bar chart and data for alternate change-path (3) for the scenario in Figure 6-17

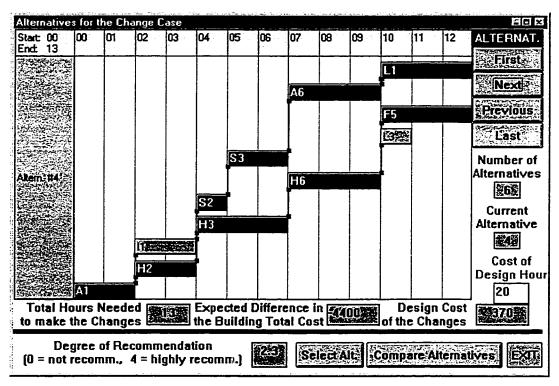


Figure 6-21 Bar chart and data for alternate change-path (4) for the scenario in Figure 6-17

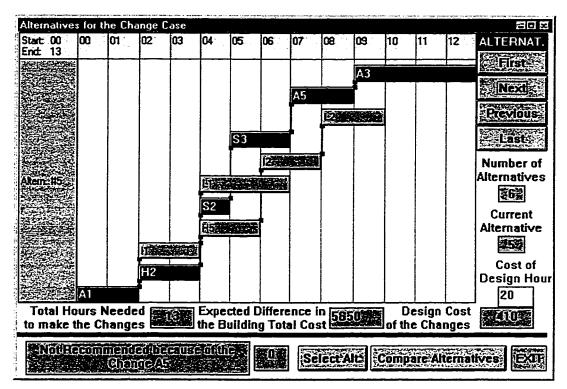


Figure 6-22 Bar chart and data for alternate change-path (5) for the scenario in Figure 6-17

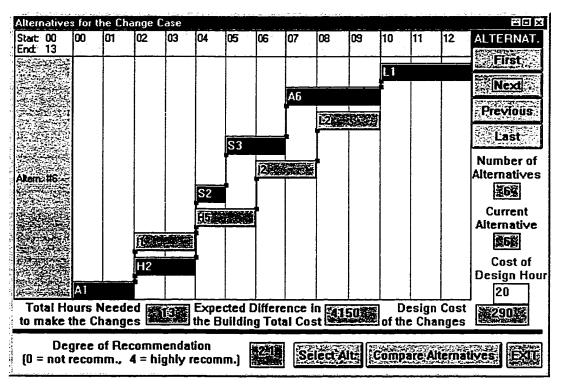


Figure 6-23 Bar chart and data for alternate change-path (6) for the scenario in Figure 6-17

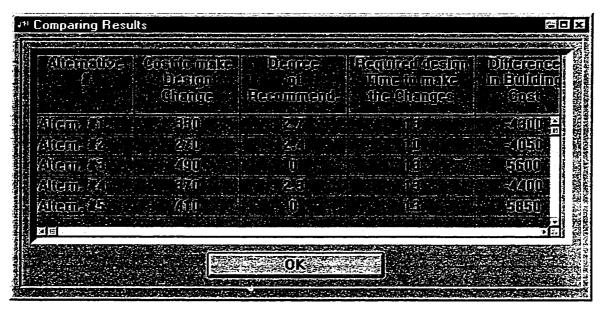


Figure 6-24 Comparison among various alternate change-paths for the scenario in Figure 6-17

6.3.4 THIRD LEVEL OF VALIDATION

The third level is to validate that the model provides a better tool over current practice and help in solving the problem of design changes in a multi-disciplinary team. The validation is performed through using a real case study that is selected from literature (Fazio 1990). The case study exhibits a real situation where incompatibility in design information led to unexpected cost overruns of 2 million dollars in a 122m (400') tall office building in a Canadian city. The building is designed using a column-and-beam steel structure system with reinforced concrete core and slabs. Figure 6-25 shows a plan view of a steel column with surrounding building components. The building envelope consists of metal air pans (component F in Figure 6-25) that are connected to the steel structure by metal clip angles (H). The function of the metal pans is to prevent airflow through the envelope. The external layer of the envelope consists of precast concrete panels (A) that are one story in height, spanning between the steel columns (B). The panels are supported on brackets that are connected to the steel beams. The windows (D) are supported by aluminum angles that are bolted to the concrete panels. Tracks that guide the window washing equipment are embedded on the exterior face of the precast panels and require strict vertical alignment of the precast panels. The columns are covered with sprayed-on fireproofing (C) and are encased by 13mm (1/2") gypsum board on metal studs (E). The induction units (G) of the HVAC system span between the gypsum board encasements of adjacent exterior columns.

Three independent design disciplines, architecture, structural, and mechanical engineering

are involved in the design of the building components that appear in Figure 6-25. However, the design of these components is inter-dependent. The architect designs the precast concrete panels, the gypsum board encasement, the metal pan and clip angles. The design objective of the architect is to get maximum utilization of internal space and maximum window area. To do so, he runs the window up to the encasement and keeps the width of the encasement to a minimum. The vertical component of the precast concrete panel (A) is designed to have the same width as the gypsum board encasement (E) so that encasement does not overlap with the windows. Both of these widths, however, should cover the width of the column flange (B), the thickness of the fire proofing (C) and take into consideration the construction tolerance of the column. The spacing between the clip angles (H) needs to be as wide as possible to reduce the surface area of the metal pan (F) and consequently its cost. Yet, the design of this spacing should take into consideration the width of the column flange and the construction tolerance of the column. The width of the metal pan is defined so that the gypsum board encasement can be fixed to it. The following data is collected and used by the architect to design the building components under his responsibility:

- The width of the steel column flange is 356mm (14").
- The maximum construction tolerance for the steel column is 13mm (0.5").
- The thickness of the sprayed fireproofing is 13mm (0.5").

The structural engineer designs the steel column (B) and later specifies its construction tolerance according to the code. The local code allows a maximum of 127mm (5") for Ahmed Mokhtar, Ph.D Thesis149

construction tolerance at the top of the building. This indicates a clear conflict with the same data element as used by the architect and may push the column outside the encasement. The error can be attributed to either oversight by the architect or poor communication. The structural engineer also specifies the type and the thickness of the fire proofing (C) for the column. A thickness of 13mm (0.5") is found to be inadequate and is later modified to 36mm (1.7/16"), consequently invalidating the value used by the architect for the same data element. The HVAC engineer designs the induction unit (G) to fit between two adjacent column encasements in accordance with the width of the encasement as given by the architect.

During construction, problems start to appear. The first problem is the connection between the welded clip angles (H) and the column flange. The strict vertical alignment of the angles conflicts with the construction tolerance of the columns. At high stories, one side of the angles would fall off the flange. The connection is redesigned and a change order is issued to use C-shape (component J in Figure 6-26) instead of the clip angles. The gypsum board encasement as designed does not cover the column and the sprayed fireproofing. A change order is issued to make the encasement wider. However, the metal pan is not wide enough for fixing the encasement. The issued change order requires the use of Z-clips (M) to support gypsum boards. The extra width of the encasement makes it impossible to fit the induction units in their place. Because these units are already delivered to site, a change order is issued to trim their length. These series of change orders also result in several construction delays. The contractors make a claim of \$6

millions for cost overruns due to conflicts in design. The case is eventually settled out of court for \$2 millions.

The proposed information model is used for this design case to examine the model ability to overcome such design incompatibility errors as described. The first step in using the model is to assemble the data-structure of the building components using the configuration module. The ability of the model to flexibly adapt to any design configuration makes it possible for the design manager to develop such an assembly. Prebuilt and dynamically built rules are then used to capture the linking knowledge among the involved disciplines. The following knowledge is captured using prebuilt rules:

• The architect uses construction tolerance of the column as an input data to design the connection of the envelope to the structure and to size the encasement of the column (Figure 6-27).

The following knowledge is captured using dynamically built rules:

- The architect sizes the width of the gypsum board column encasement (E) to be the same as the width of the precast concrete panel (A) and based on the following data:
 - A width of 356mm (14") for the flange of the steel column.
 - A thickness of 13mm (0.5") for the sprayed fireproof (Figure 6-28).

• The HVAC engineer sizes the length of the induction unit (G) based on the width of the gypsum board encasement and the column spacing.

The design changes that occurred in the case study are then performed in the model. The structural engineer increases the thickness of the fire proofing from a value of 13mm (0.5") to an adequate value of 36mm (1 7/16"). Consequently, two messages are automatically sent to the architect informing him about the change. The first message (Figure 6-29) shows that the change affects the width of the gypsum board encasement and the second message shows that the change affects the width of the precast concrete panel (A). Accordingly, the architect increases the width of these two components along with the width of the metal pan (F) to enable fixing of the gypsum board encasement. However, as soon as the architect changes the width of the gypsum board encasement, a message is sent automatically to the HVAC engineer informing about the change and indicating that the length of the induction unit (G) is affected. Similarly, when the structural engineer specifies the value of the construction tolerance for the steel columns, a message is automatically sent to the architect. The message shows the correct value of the tolerance, informing that this value affects the design of the connection between the envelope and the structure and also the width of the encasement. The architect then uses the correct tolerance value and redesigns the connection using C-shape instead of the clip angles. He also resizes the width of both the encasement and the metal pan to accommodate the expected construction tolerance. Another message is then sent to the HVAC engineer automatically informing him about the new width of the encasement and

indicating that the length of the induction unit (G) is affected. The HVAC engineer then corrects the length of this unit.

This case study demonstrates how small elements of data can easily be missed in current design practice. These result in incompatibilities in design information and lead to change orders and unnecessary increase in the building cost and project duration. Using the model, design changes are accommodated before starting building construction. The success of the model largely depends on the linking knowledge that is captured from designers. However, the model provides the design team with the ability to record and make use of this knowledge. In current practice, such knowledge is scattered around, unused and/or completely wasted throughout the design process. The case study proves that the use of the model provides a better tool for designers in multi-disciplinary design environment to accommodate design changes. Consequently, the related problems that appear during construction are avoided.

6.4 SUMMARY AND CONCLUSIONS

The model is validated through three levels of validation. Hypothetical design scenarios ensured that the model as it is developed and implemented can provide the expected capabilities in managing design changes. A real case study proved that the model can improve the current design practices and help building design professionals eliminate problems related to incompatibility in design information.

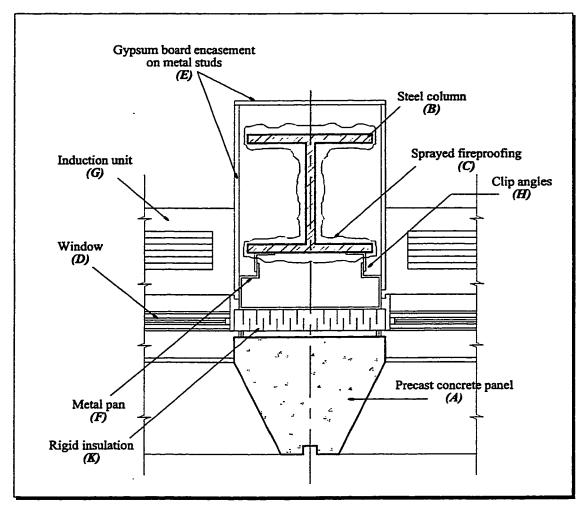


Figure 6-25 Plan view of an exterior column and the surrounding components as designed (Fazio 1990 reproduced)

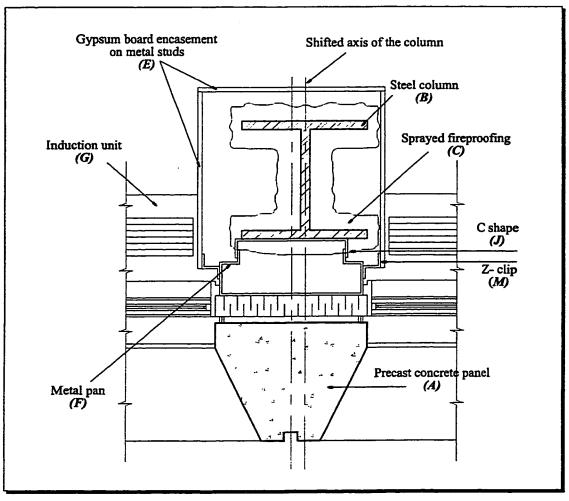


Figure 6-26 Plan view of the exterior column and the surrounding components as constructed (Fazio 1990 reproduced)

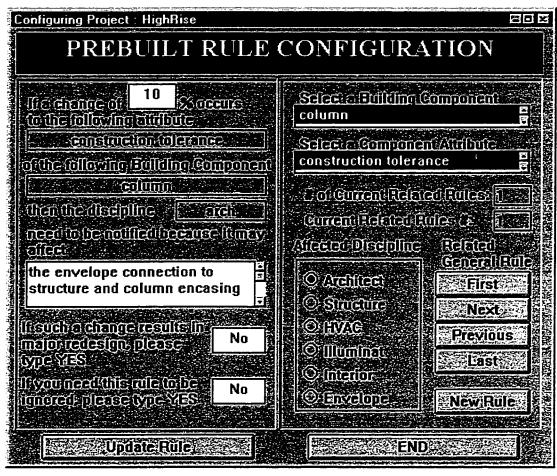


Figure 6-27 Prebuilt rule to capture link between construction tolerance and design of envelope and encasement.

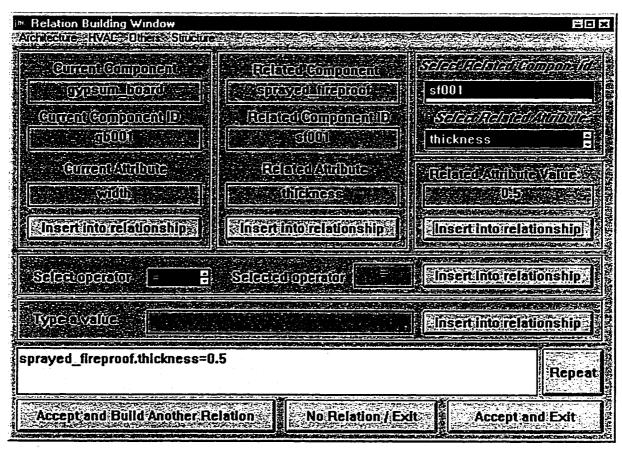


Figure 6-28 Dynamically built rule to capture link between width of the gypsum board encasement and fireproofing thickness¹.

¹ Units in Figure 6-28 and Figure 6-29 are in Inches.

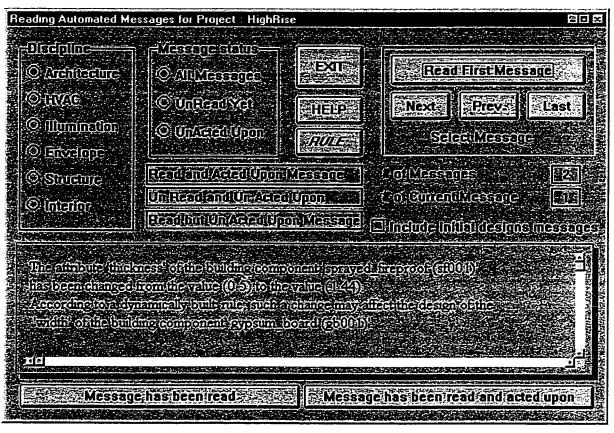


Figure 6-29 Automated message to inform architect about change in thickness of fireproofing.

CHAPTER SEVEN

SUMMARY, CONTRIBUTIONS, AND RECOMMENDATIONS

7.1 SUMMARY

Incompatibilities in construction technical documents represent a major problem for the construction industry. An analysis of the production process of these documents during the detailed design stage reveals that managing design changes is a main cause for incompatibilities. More specifically, propagation of design changes is a principal source of problems. The large amount of data that is generated within a multi-disciplinary design environment makes this task very complex, especially when considering the fact that the involved design disciplines are, in most cases, separated both spatially and educationally.

This thesis presents the development of a model that uses information technology to address the problem. The core concept of the model is a central database that functions as a repository of active building components. Each building component in this database not only carries its design data, but is also capable of recognizing the design disciplines that are affected by any change in these data and automatically send them messages. As such, the complex task of propagating design changes is assigned to the building components themselves. The components efficiently perform the task regardless of the amount of data or the number of disciplines involved in a project. Notification messages are automatically sent by any building component to only affected design disciplines as soon as a design change is made to any attribute of the building components. The messages

clearly describe the design changes and advise each design discipline on how the change may influence its building system. An appropriate reaction is left to the designer who is now well informed. The active building components are able to perform this task because they are equipped with the necessary linking knowledge. The linking knowledge is acquired from the designers and is implemented in the form of rules.

The central database is composed of two parts. The first is the building-components-database which carries the active building components data. The second part is the management-database which carries data that is necessary for the various functions of the model. The concept of management-database allows the building-components-database to adapt easily to any building configuration, an essential requirement because buildings are unique products. The utilization of combined Object-Oriented and Relational Database technologies enables the model to achieve this flexibility.

The developed information model is also equipped with the capability to track past design changes. It can answer the requests of design managers to track past design changes in a variety of ways. It can also show the automated messages that are sent by a building component to each affected discipline as a reaction to a design change. The response of the informed discipline to each of these messages can also be tracked.

The model is also developed to facilitate the planning and scheduling of future design changes. A methodology is incorporated within the model and uses its active components to collect future design change data. This data is organized using especially developed sub-routines, then scheduled and manipulated with some calculations that provide the design manager with a clear view on how to conduct a future design change.

The information model is successfully implemented in a client/server setup. It is subjected to three levels of validation. The validation process ensured not only that the model provides the capabilities that are expected from its development but also that the model is a better tool over current practices for managing design changes in a collaborative multi-disciplinary design environment.

7.2 CONTRIBUTIONS

Aiming to improve the quality of buildings, the contributions of this research towards eliminating the serious problems associated with incompatibility errors in construction technical documents can be summarized as follows:

- A thorough analysis of the production process of construction technical documents with an emphasis on the techniques for coordinating design information so as to accommodate design changes. The analysis diagnoses the main problems that cause incompatibilities in design information. These problems are the management of design changes, the media used to communicate design information, and the source of final documents.
- Review of research work in the area of using computer technology in collaborative design. The review includes an analysis of approaches used in these researches so as

to evaluate their capabilities and limitations to eliminate incompatibilities in design information.

- Development of an information model that establishes the theoretical basis for a computer-aided collaborative multi-disciplinary design environment that facilitates the management of design changes. The model provides remedy for problems with current production processes of construction technical documents and helps ensure the compatibility of design information scattered among these documents. The model, being the first of its kind, not only helps accommodating design changes but also tracks past design changes and helps planning and scheduling of future ones as well. In addition, the model has the flexibility to include any configuration of different building components along with the knowledge that connect them. Hence, it is able to function with any type of building. The model, however, does not cover all the information needed for the design of a building such as codes and design constraints. The model is also intended for sequential type of project delivery systems. It is not intended for other types of delivery systems such as fast track system. Designers who will use the model are expected to collaborate using client-server networks and therefore they should be located in the same building or nearby buildings. Other types of networks such as wide area networks, intranets, or internet are beyond the scope of this model.
- Development and implementation of the concept of active building components. Such a concept enables the management of design changes by assigning the task of propagating

design changes to the building components that are subject to change rather than the designers who perform these changes. The building components are made active by capturing the necessary linking knowledge from the designers. A process is developed and implemented to capture this knowledge in terms of prebuilt and dynamically built rules. The active components, however, do not use a knowledge-based system and therefore no induction is performed by the components.

- Development and implementation of the concept of management database. Such database manages the data-structure of the building-components-database, which carries the design data, and allows it to be customizable to fit any type of building. The management database also keeps track of all design changes as well as the messages that are automatically sent to the designers due to these changes. In addition, it carries the data necessary to plan and schedule future design changes.
- Development and implementation of the concept of planning and scheduling future design changes. To realize this concept, a methodology has been outlined to utilize the capabilities of active building components in collecting design change data from designers so as to enable the planning of extended design changes. Subroutines have been created to translate the collected data into organized design change paths that show the design manager all the possible paths to perform a design change. It also shows the time needed, the design cost, the impact on building cost, and the recommendation level for each of these change paths.

Validation of the developed model not only through hypothetical scenario but also with a real design case. The hypothetical scenario proves that the model provides the capabilities that are expected from its development while the real design case shows that the model improves the current design practices, which is the ultimate objective of developing the model.

7.3 RECOMMENDATIONS FOR FUTURE WORK

Due to the fast technological development in this field of research, many new avenues can be explored to apply and evolve the concepts that are presented in this thesis. At the time of writing these recommendations, we can see potential developments as follows:

- Explore the use of the internet as the carrier of the central project-database and as the medium for automated messages propagation.
- Develop multimedia messages that provide clearer contents with more information.
- Apply the standards of IFCs (IAI 1996) and STEP (Froese 1996) for the structure of the building-components-database.

Some complementary components of the information model have not been developed in this research due to resource limitations. Other researchers can work on these developments as:

• The construction technical documents generation module. This module is responsible for generating, from the project-database, the documents that are used by the

contractors to construct the building. The module needs to have the following features:

- Document formats and classifications need not to be based on disciplines any longer e.g., architectural drawings and structural drawings. Some documents could be prepared on the basis of the requirements of the contractors and the subcontractors so as to make their job easier and with fewer errors (see Mokhtar and Bédard 1994).
- The output medium for construction technical documents needs not to be paper anymore; it can be a laptop connected through a wireless modem to the central database that lies on the internet.
- A drafting interface that facilitates the manipulation of graphical data.
- A mechanism to control the access to the project-database by various users.

More development for some of the model main components is also recommended. The following is advocated:

- More sophisticated relation-acquisition capability that can acquire and incorporate design rationale.
- Automated control of the design changes after an alternative is chosen through the planning and scheduling capability of the model.
- Connect the results of the planning and scheduling to a project management software such as Primavera (1996).

REFERENCES

- Ahmed, S., Sriram, D., Logcher, R., 1992, "Transaction-Management Issues in Collaborative Engineering," <u>Journal of Computing in Civil Engineering</u>, ASCE, Vol.6, No.1, January, pp.85-105.
- Arnold, J., Teicholz, P., 1996, "Data Exchange: File Transfer, Transaction Processing and Application Interoperability," <u>Computing in Civil Engineering</u>, ASCE, Proceedings of the Third Congress, Anaheim, California, June 17-19, pp.438-444.
- Augenbroe, G., 1992, "Integrated Building Performance Evaluation in Early Design Stages," <u>Building and Environment</u>, Vol.27, No.2, pp.149-161.
- Augenbroe, G., 1993, "COMBINE FINAL REPORT," available through anonymous ftp (internet address: 130.161.136.37) under pub/combine/documents.
- Bakkeren, W., Tolman, F., 1995, "Integrated Structural Synthesis and Evaluation using Product Data," <u>Computing in Civil and Building Engineering</u>, Balkema, Rotterdam, pp.291-298.
- Bédard, C., Rivard, H., 1995, "Integration in Building Engineering Design with Database," <u>Computing in Civil Engineering</u>, ASCE, Proceedings of the Second Congress, Atlanta, GA, June 5-8, pp.502-508.
- Björk, B-C., 1989, "Basic Structure of a Proposed Building Product Model," <u>Computer Aided Design</u>, Vol.21, No.2, March, pp.71-78.
- Björk, B-C., Penttilä, H., 1989, "A Scenario for the Development and Implementation of a Building Product Data Model Standard," <u>Advanced Engineering Software</u>, Vol.11, No.4, pp.176-187.
- Björk, B-C., Kurt, L., Arto, K., 1997, "ISO DIS 13567 The Proposed International Standard for Structuring Layers in Computer Aided Building Design," <u>Electronic Journal of Information Technology in Construction ITCon</u>, Vol.2, http://itcon.org
- Bowen, J., Bahler, D., 1993, "Constraint-Based Software for Concurrent Engineering," Computer, January, pp.66-68.
- Ching, F., 1991, "Building Construction Illustrated," Van Nostrand Reinhold, New York, 6 chapters in various pagings.
- Cornick, T., 1990, "Quality Management for Building Design," Butterworth Architecture Management Guides, 218 pages.
- Cutkosky, M.R., Engelmore, R.S., Fikes, R.E., Genesereth, M.R., Gruber, T.R., Mark, W.S., Tenenbaum, J.M., Weber, J.C., 1993, "PACT: An Experiment in Integrating Concurrent Engineering Systems," <u>COMPUTER</u>, January, pp.28-37.
- DeChiara, J., 1980, "Handbook of Architectural Details for Commercial Buildings," McGrow-Hill Company, 508 pages.

- De la Garza, J., Alcantara, P., 1997, "Using Parameter Dependency Network to Represent Design Rationale," <u>Journal of Computing in Civil Engineering</u>, ASCE, Vol.11, No.2, pp.102-112.
- De Waard, M., 1992, "Computer Aided Conformance Checking: checking residential building designs against building regulations with the aid of computers," <u>Doctoral Dissertation</u>, Delft University of Technology, Delft, The Netherlands, 197 pages.
- Dubois, A.M., 1992, "COMBINE Project Task: IDM Conceptual Development," <u>Presentation at the seminar of COMBINE partners hosted by Fraunhofer Institut fur Bauphysik</u>, Stuttgart, Germany, November.
- Dubois, A.M., Parand, F., 1993, "COMBINE Integrated Data Model," <u>CIBSE National Conference</u>, pp.96-108.
- DXF, 1995, AutoCAD User's Guide, Autodesk Inc., 561 pages.
- Eastman, C.M., 1992a, "Modeling of Buildings: Evolution and Concepts," <u>Automation in Construction</u>, Vol.1, pp.99-109.
- Eastman, C.M., 1992b, "A Data Model Analysis of Modularity and Extensibility in Building Databases," <u>Building and Environment</u>, Vol.27, No.2, pp.135-148.
- Eastman, C.M., 1994, "A Data Model for Design Knowledge," <u>Automation in Construction</u>, Vol.3, pp.135-147.
- Eastman, C.M., Cho, M., Jeng, T., Assal, H., 1995, "A Data Model and Database Supporting Integrity Management," <u>Computing in Civil Engineering</u>, ASCE, Proceedings of the Second Congress, Atlanta, GA, June 5-8, pp.517-524.
- El-Bibany, H., 1992, "Architecture for the Human-Computer Design, Management and Coordination in a Collaborative AEC Environment," <u>Ph.D Thesis</u>, Stanford University, Stanford, CA., 111 pages.
- El-Bibany, H., Paulson, B.C., 1994, "Collaborative Knowledge-integration Systems: A Tool for AEC Design, Management, and Coordination," <u>Microcomputers in Civil Engineering</u>, Vol.9, pp.29-40.
- El-Bibany, H., 1996, "Domain Modeling in Generic Parametric Architecture: Issues in Concurrent Representation and Inference," <u>Computing in Civil Engineering</u>, ASCE, Proceedings of the Third Congress, Anaheim, California, June 17-19, pp.522-528.
- Elmasri, R., Navathe, S., 1994, "<u>Fundamentals of Database Systems</u>," Benjamin Cummings, Don Mills, Ontario, 873 pages.
- Enseleit, J., Bitzer, D., Pahl, P., 1995, "Experiences in Using STEP for Distributed Structural Analysis," Computing in Civil and Building Engineering, Balkema, Rotterdam, pp.337-342.
- Fazio, P., 1990, "A Case Study in Integrated Building Design," <u>Canadian Journal of Civil Engineering</u>, Vol.17, pp.636-642.

- Fenves, S.J., Flemming, U., Hendrickson, C., Maher, M.L., Schmitt, G., 1990, "Integrated Software Environment for Building Design and Construction," <u>Computer Aided Design</u>, Vol.22, No.1, pp.27-36.
- Fenves, S.J., Flemming, U., Hendrickson, C., Maher, M.L., Schmitt, G., Hendrickson, C., Quadrel, R., Terk, M., Woodbury, R., 1994, "Concurrent Computer-Integrated Building Design," Prentice Hall, NJ, 242 pages.
- Fischer, M.A., Kunz, J., 1995, "The Circle: Architecture for Integrating Software," <u>Journal of Computing in Civil Engineering</u>, ASCE, Vol.9, No.2, pp.122-133.
- Flemming, U., Woodbury, R., 1995, "Software Environment to Support Early Phase in Building Design (SEED): Overview," <u>Journal of Architectural Engineering</u>, ASCE, Vol.1, No.4, December, pp.147-152.
- Froese, T., 1996, "STEP and the Building Construction Core Model," <u>Computing in Civil Engineering</u>, ASCE, Proceedings of the Third Congress, Anaheim, California, June 17-19, pp.445-451.
- Goodman, R., Chinowsky, P., 1996, "Managing Inter-disciplinary Project Team Through the Web," <u>Computing in Civil Engineering</u>, ASCE, Proceedings of the Third Congress, Anaheim, California, June 17-19, pp.452-458.
- Guenster, R., 1996, "Facilitating Workgroup Activities Through the Internet," Computing in Civil Engineering, ASCE, Proceedings of the Third Congress, Anaheim, California, June 17-19, pp.459-465.
- Hegazy, T., Khalifa, J., 1996, "Expert Rules-of-Thumb for Effective Design Coordination," AACE Transactions, pp.4.1-4.6.
- Herold, K., 1997, "Editorial-Universal Building Language," <u>Journal of Computing in Civil Engineering</u>, ASCE, January, pp.1-3.
- Huyn, P.N., Genesereth, M.R., Letsinger, R., 1993, "Automated Concurrent Engineering in Designworld," COMPUTER, January, pp.74-76.
- IAI, 1996, Industrial Alliance for Interoperability, http://www.interoperability.com
- ISO, 1989, "Industrial Automation Systems Product Data Representation and Exchange," International Organization for Standardization, ISO/TC184/SC4, ISO/DP 10303.
- Jones, B., Riley, M., 1995, "Collaborative Construction Agents," <u>Computing in Civil Engineering</u>, ASCE, Proceedings of the Second Congress, Atlanta, GA, June 5-8, pp.1316-1323.
- Karhu, V., 1997, "Product Model Based Design of Precast Facades," <u>Electronic Journal</u> of Information Technology in Construction ITCon, Vol.2, http://itcon.org
- Khedro, T., 1994, "Future Enterprise Software Integration Through Better Software Interoperation," <u>Bridging the Generations</u>, International Workshop on the Future

- Directions of Computer-Aided Engineering, Carnegie Mellon University, Pennsylvania, June 18-19, pp.117-122.
- Khedro, T., Teicholz, P., Genesereth, M.R., 1994 "A Framework for Collaborative Distributed Facility Engineering," <u>Computing in Civil Engineering</u>, ASCE, Proceedings of the First Congress, Washington D.C., 20-22 June, pp.1489-1496.
- Khedro, T., Teicholz, P., Genesereth, M.R., 1995, "Collaborative Design: Communicating Information through an Agent-Based Network," <u>CIFE newsletter</u>, Centre for Integrated Facility Engineering, Stanford University, No.6, Winter, pp.4,13.
- Khedro, T., Eastman, C., Junge, R., Liebich, T., 1996, "Translation Methods for Integrated Building Engineering," <u>Computing in Civil Engineering</u>, ASCE, Proceedings of the Third Congress, Anaheim, California, June 17-19, pp.579-585.
- Krishnamurthy, K., Law, K., 1995a, "A Data Management Model for Design Change Control," <u>Concurrent Engineering: Research and Applications</u>, Vol.3, No.4, December, pp.329-343.
- Krishnamurthy, K., Law, K., 1995b, "Change Management for Collaborative Engineering," Computing in Civil Engineering, ASCE, Proceedings of the Second Congress, Atlanta, GA, June 5-8, pp.1110-1117.
- Krishnamurthy, K., Law, K., 1996, "A Data Management Model for Change Control in Collaborative Design Environment," <u>Computing in Civil Engineering</u>, ASCE, Proceedings of the Third Congress, Anaheim, California, June 17-19, pp.536-543.
- Krishnamurthy, K., 1996, "A Data Management Model for Change Control in Collaborative Design Environment," Ph.D. Thesis, Department of Civil Engineering, Stanford University, Stanford, CA., 186 pages.
- Lave, C.A., March, J.G., 1975, "An Introduction to Models in the Social Sciences," Harper & Row, Publishers, 421 pages.
- LEVEL5 OBJECT, 1994, <u>Level5 Object for Microsoft Windows Reference Guide</u>, Information Builders, NewYork, 405 pages.
- McGraw, K., Lawrence, P., Morton, J., Heckel, J., 1996, "The Agent Collaboration Environment An Assistant for Architects and Engineers," Computing in Civil Engineering, ASCE, Proceedings of the Third Congress, Anaheim, California, June 17-19, pp.739-745.
- Mokhtar, A., Bédard, C., 1994, "Towards Integrated Construction Technical Documents A New Approach Through Product Modelling," <u>Proceedings of The First European Conference on Product and Process Modelling in the Building Industry</u>, Dresden, Germany, October 5-7, pp.3-9.
- Mokhtar, A., Bédard, C., Fazio, P., 1997a, "Planning and Scheduling Design Changes in a Collaborative Multi-disciplinary Design Environment," Computing in Civil

- Engineering, ASCE, Proceedings of the Fourth Congress, Philadelphia, 16-18 June, pp.97-105.
- Mokhtar, A., Bédard, C., Fazio, P., 1997b, "A Methodology for Dynamically Assembling and Modifying an Active-Repository Building Database," <u>IT Support for Construction Process Re-engineering</u>, Proceeding of the CIB Workshop W78 and TG10, Cairns, Queensland, Australia, July 9-11, pp.285-295.
- Mokhtar, A., Bédard, C., Fazio, P., 1998, "Information Model for Managing Design Changes in a Collaborative Environment," <u>Journal of Computing in Civil Engineering</u>, ASCE, Vol.12, No.2, pp82-92.
- MS-WINDOWS-95, 1995, Microsoft Windows 95 Introduction, Microsoft Corporation, 95 pages.
- MS-ACCESS, 1994, Microsoft Access Getting Started, Microsoft Corporation, 168 pages.
- NOVEL-NETWARE, 1993, Novel Netware 3.12 User's Guide, Novel Inc., 6 chapters in various pagings.
- Paulson, B., 1995, "Computer Applications in Construction," McGraw Hill, New York, 610 pages.
- Peña-Mora, F., Sriram, D., Logcher, R., 1995, "Design Rationale for Computer-Supported Conflict Mitigation," <u>Journal of Computing in Civil Engineering</u>, ASCE, Vol.9, No.1, pp.57-72.
- Platt, D., 1996, "Building Process Models for Design Management," <u>Journal of Computing in Civil Engineering</u>, ASCE, Vol.10, No.3, pp.194-203.
- PREMAVERA, 1996, <u>Primavera Project Planner Reference</u>, Premavera System Inc., 741 pages.
- Ramsey, C., Sleeper, H., 1994, "Architectural Graphic Standards," John Wiley & Sons Inc., 918 pages.
- Reed, K., Harrod, D., Conroy, W., 1990, "The Initial Graphics Exchange Specifications (IGES)," NISTIR 4412, September.
- Rezgui, Y., Debras, P., 1995, "An Integrated Approach for a Model Based Document Production and Management," <u>Electronic Journal of Information Technology in Construction ITCon</u>, Vol.1, http://itcon.org.
- Rivard, H., Bédard, C., Ha, K., Fazio, P., 1995, "An Information Model for the Building Envelope," <u>Computing in Civil Engineering</u>, ASCE, Proceedings of the Second Congress, Atlanta, GA, June 5-8, pp.302-309.
- Rojas, E., 1997, "Developing Web-Centric System for Collaborative Engineering," Computing in Civil Engineering, ASCE, Proceedings of the Fourth Congress, Philadelphia, 16-18 June, pp.232-239.

- Sriram, D., Stephanopoulos, G., Logcher, R., Gossard, D., Groleau, R., Serrano, D., Navinchandndra, N., 1989, "Knowledge-Based System Applications in Engineering Design: Research at MIT," <u>AI Magazine</u>, Vol.10, Fall, pp.79-96.
- Sriram, D., Logcher, R., Wong, A., Ahmed S., 1990, "A Case Study in Computer-Aided Cooperative Product Development," Technical Report No. IESL-90-01, Department of Civil Engineering, Intelligent Engineering System Laboratory, Massachusetts Institute of Technology, Cambridge, 81 pages.
- Sriram, D., Logcher, R., 1993, "The MIT Dice Project," COMPUTER, January, pp.64-65.
- Sriram, D., Logcher, R., Wong, A., 1994, "Computer Supported Collaborative Engineering: Research Issues in Product Modeling," <u>Bridging the Generations</u>, International Workshop on the Future Directions of Computer-Aided Engineering, Carnegie Mellon University, Pennsylvania, June 18-19, pp.87-92.
- Svensson, K.E., 1993, "Maintenance Information Management, Based on a Step-Related Building Product Model," <u>Management of Information Technology for Construction</u>, edited by Mathur, K., Betts, M., Tham, K.W., World Scientific and Global Publications Services, Singapore, pp.509-516.
- Taylor, D., 1992, "Object-Oriented Information Systems: Planning and Implementation," Addison Wesley Publishing Company, 357 pages.
- Tilley, P., Barton, R., 1997, "Design and Documentation Deficiency Causes and Effects," <u>Proceeding of the International Conference on Construction Process Re-engineering</u>, Gold Cost, Queenland, Australia, 14-15 July, pp.703-712.
- Tolman, F.P., Kuiper, P., Luiten, G.T., 1990, "Product Modelling at Work," Computer Integrated Construction Proceedings of the Second CIB W78 & W74 Joint Seminar on Computer Integrated Construction, Tokyo, Japan, 17-19 September, pp.17-26.
- Tsou, J-Y., 1992, "Using Conceptual Modelling and Object-Oriented Environment to Support Building Cost Control during Early Design," <u>Doctoral Dissertation</u>, University of Michigan, Ann Arbor, 345 pages.
- Turner, J.A., 1990, "AEC Building Systems Model," ISO TC184/SC4/WG1 (STEP) working paper, 23 pages.
- Turner, J.A., 1992, "Conceptual Modelling Applied to Computer-aided Architectural Design," <u>Building and Environment</u>, Vol.27, No.2, pp.125-133.
- Vahala, P., 1995, "Prototype User Interface for a Building Product Model in Renovation Design," Computing in Civil and Building Engineering, Balkema, Rotterdam, pp.359-363.
- Voeller, J., 1996, "Editorial Data-centered Thinking," <u>Journal of Computing in Civil Engineering</u>, ASCE, January, pp.1-2.
- Vries, B., Somers, L., 1995, "Message Exchange in the Building Industry," <u>Automation in Construction</u>, Vol.4, pp.91-100.

Wong, Deve	A., Sriram, I lopment," <u>Re</u>	D., 1993, "SI search in En	IARED: Argineering D	n Information esign, Vol.5	on Model for 5, No.1, pp.2	Cooperative 1-39.	e Produc
•							
						-	

DATABASE STRUCTURE IN MS ACCESS

A.1 MANAGEMENT-DATABASE

A.1.1 CONCEPTUAL DATA-STRUCTURE

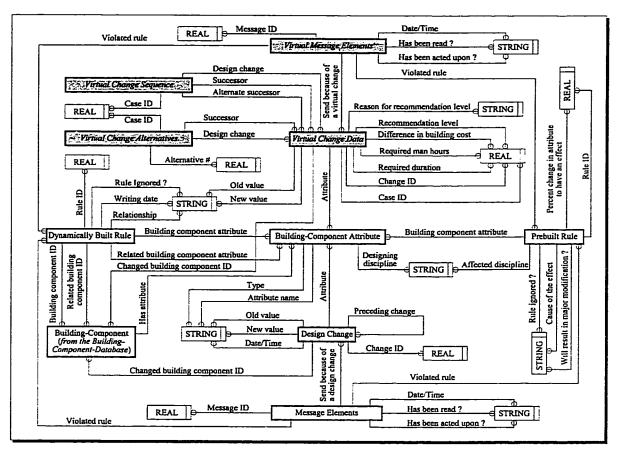


Figure A-1 Conceptual data-structure for management-database presented in EXPRESS-G.

A.1.2 STRUCTURE OF DATA-TABLES

MANAGERANIA MAKERIANIANIKETANAKAN DULANIALA

ID
BUILDING_COMPONENT
ATTRIBUTE_OF_BUILDING_COMPONEN
T
ATTRIBUTE_TYPE
DESIGNING_DISCIPLINE
LINKED_BUILDING_COMPONENT
LINKED_ATTRIBUTE_OF_BUILDING_CO
MPONENT
GRAPH_NAME

MKGANIKAGONIGAKA 11(0); Duganioyez

ID LABEL MAIN_MENU_LABEL

neglijeks Menikanneglijs

PREBUILT_RULE_ID
BUILDING_COMPONENT
ATTRIBUTE_OF_BUILDING_COMPONEN
T
AFFECTED_DISCIPLINE
PERCENT_CHANGE_TO_HAVE_AN_EFF
ECT
RESULT_IN_MAJOR_MODIFICATION_?
RULE_IGNORED_?
CAUSE_OF_THE_EFFECT

Dieleijeka Mielykkaniekaanalairis Kaas

DYNAMICALLY BUILT RULES ID

BUILDING_COMPONENT
BUILDING_COMPONENT_ID
ATTRIBUTE_OF_BUILDING_COMPONEN
T
RELATED_BUILDING_COMPONENT
RELATED_BUILDING_COMPONENT_ID
RELATED_ATTRIBUTE
RELATIONSHIP
WRITING_DATE
RULE_IGNORED_?

incarnezes wkenajekes

CHANGE_ID
DATE_TIME
DISCIPLINE
BUILDING_COMPONENT
BUILDING_COMPONENT_ID
ATTRIBUTE
OLD_VALUE
NEW_VALUE
PRECEEDING CHANGE ID

MAKCEAALGAAYAFGAAYCGA Magagaafaa

ID
CASE_ID
CHANGE
BUILDING_COMPONENT
BUILDING_COMPONENT_ID
ATTRIBUTE
REQUIRED_MAN_HOURS
DIFFERENCE_IN_BUILD_COST
RECOMMENDATION_LEVEL
REASON_FOR_RECOMMENDATION
DURATION
OLD_VALUE
NEW_VALUE

DAGAGIGEASA WKGSYISSAGGAGAGATAKES

ID
CHANGED_ATTRIBUTE
CHANGED_BUILDING_COMPONENT
CHANGED_BUILDING_COMPONENT_ID
OLD_VALUE
NEW_VALUE
AFFECTED ATTRIBUTE

AFFECTED_BUILDING_COMPONENT AFFECTED_BUILDING_COMPONENT_ID

RELATIONSHIP
WRITING_DATE
BEEN_READ_?
BEEN_ACTED_UPON_?
DISCIPLINE TO

DISCIPLINE_FROM
SENT_BECAUSE_CHANGE_ID
PREBUILT OR DYNAMICALLY ?

RULE_WRITING_DATE
VIOLATED RULE ID

RULE_IGNORED_?

CAUSE_OF_THE_EFFECT

Datiebeles Miceographicamessored mangras

ID CHANGED ATTRIBUTE CHANGED BUILDING COMPONENT CHANGED BUILDING COMPONENT_ID OLD_VALUE **NEW VALUE** AFFECTED ATTRIBUTE AFFECTED BUILDING COMPONENT AFFECTED BUILDING COMPONENT_ID RELATIONSHIP WRITING DATE BEEN READ? BEEN ACTED UPON? DISCIPLINE TO DISCIPLINE_FROM SENT BECAUSE CHANGE ID PREBUILT OR DYNAMICALLY? RULE_WRITING_DATE VIOLATED RULE ID CASE ID

CAUSE_OF_THE_EFFECT

SEONIZACE NECESALUITA ESINIZADE DESENDEAS

ID_NUMBER
CASE_ID
DESIGN_CHANGE
SUCCESSOR
ALTERNATE_SUCCESSOR

ID
DESIGN_CHANGE
SUCCESSOR
ALTERNATIVE#
CASE_ID

A.2 GENERAL-DATABASE

A.2.1 CONCEPTUAL DATA-STRUCTURE

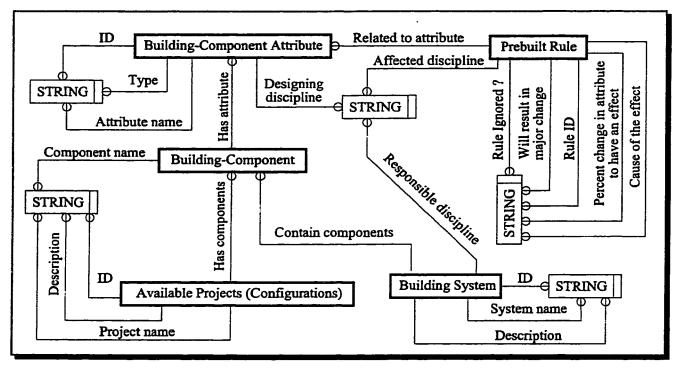


Figure A-2 Conceptual data-structure for general-database presented in EXPRESS-G.

A.2.2 STRUCTURE OF DATA-TABLES

VIARINGIAC Garainadikecanaraka Duraringar

ID
BUILDING_COMPONENT
ATTRIBUTE_OF_BUILDING_COMPONEN
T
ATTRIBUTE_TYPE
DESIGNING_DISCIPLINE
LINKED_BUILDING_COMPONENT
LINKED_ATTRIBUTE_OF_BUILDING_CO
MPONENT
GRAPH_NAME

ॻज़ॿॎॸऻढ़ढ़ढ़ ॴॖढ़ॿॺॹॴॴॿॿॹॹढ़ड़

PREBUILT_RULE_ID
BUILDING_COMPONENT
ATTRIBUTE_OF_BUILDING_COMPONEN
T
AFFECTED_DISCIPLINE
PERCENT_CHANGE_TO_HAVE_AN_EFF
ECT
RESULT_IN_MAJOR_MODIFICATION_?
RULE_IGNORED_?
CAUSE_OF_THE_EFFECT

DTEATHCASE CANSTRUMENTS CONTROL OF THE CONTROL OF T

ID
BUILDING_SYSTEM
DESCRIPTION
DISCIPLINE

Data Fable As 42 GNL & OMEONENTS FOR BUILDING SYSTEMS

ID
BUILDING_SYSTEM
BUILDING_COMPONENT

DAFHADISASS GLISEGURISHISHRISASSISS

ID
PROJECT_NAME
DESCRIPTION

SUBROUTINES FOR PLANNING AND SCHEDULING DESIGN CHANGES

B.1 SUBROUTINE TO GENERATE THE VARIOUS CHANGE-PATHS.

B.1.1 INITIALIZATIONS

- Sub1-A. Connect to Data-Table 5-6 (page 111) and Data-Table 5-8 (page 113) of the related project-database.
- Sub1-B. Limit the reading of the data from Data-Table 5-6 to the rows that have the specified value for the attribute "CASE_ID" (i.e. CASE_ID = 2).
- Sub1-C. Create two lists for "Processed design changes" and "Under processing design changes".

B.1.2 GENERATE THE FIRST CHANGE-PATH

- Sub1-D. Query Data-Table 5-6 (page 111) for the data row that have the value "NULL" in the attribute "DESIGN_CHANGE" and read the value of the attribute "SUCCESSOR" in the found row. This value is the initial design change that has been defined when a designer starts a new case (i.e. A1 in Figure 5-15 on page 111). Add this design change (i.e. A1) to both the list of "Processed design changes" and the list of "Under processing design changes".
- Sub1-E. Save this data in Data-Table 5-8 (page 113) with the attribute "ALTERNATIVE#" given the value "ALTERNATIVE#1". The value for

the attribute "ID" is automatically generated in the database. Sample for the saved data is in the row with ID = 4734 in Figure 5-17 (page 113).

For each design change in the list of "Under processing design changes", perform steps Sub1-F to Sub1-H:

- Sub1-F. Query from Data-Table 5-6 (page 111) all the data rows which have the value of the attribute "DESIGN_CHANGE" equals the under processing design change (i.e. A1). Read the corresponding values in the attribute "SUCCESSOR" (i.e. H1, L1, I1 in Figure 5-15 on page 111).
- Sub1-G. Save the data in Data-Table 5-8 (page 113) for each read successor similar to the saving process in step Sub1-E. Samples are rows with ID = 4735, 4736, 4737 in Figure 5-17 on page 113.
- Sub1-H. For each read successor in Sub1-F, check if the successor exists in the list of "Processed design changes".
 - If it exists then ignore it.
 - If it does not exist then add it to the list of "Processed design changes" and the list of "Under processing design changes". Figure B-1 shows the result of applying steps Sub1-D till Sub1-H on the example given in Figure 5-10 on page 105.

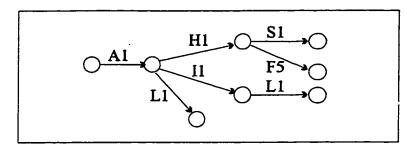


Figure B-1 Result of applying steps Sub1-D till Sub1-H on example given in Figure 5-10.

B.1.3 GENERATE THE OTHER CHANGE-PATHS

- Sub1-I. Read sequentially the design changes in the last generated change-path starting from the end design changes (i.e. F5, S1, or L1 in Figure B-1) to the first design change.
- Sub1-J. For each design change that is read, check using Data-Table 5-6 (page 111) if the design change has an alternate successor.
 - If it does not have an alternate successor (i.e. S1) then ignore this design change.
 - If it has one but the last generated change-path uses the value of the alternate successor then ignore this design change
 - If it has an alternate successor (i.e. A1) and the last generated changepath uses the value of the successor, then repeat steps Sub1-D to Sub1H to generate the next change-path with the use of the

 "ALTERNATE_ SUCCESSOR" for this change (i.e. H2 in Figure
 5-15 on page 111) instead of the "SUCCESSOR".

In case that the second option is the valid one, the value of the

"SUCCESSOR" - not the "ALTERNATE_SUCCESSOR" - is used the next time this design change passes through steps Sub1-D to Sub1-H. Figure B-2 shows the result of applying steps Sub1-I and Sub1-J on the example given in Figure 5-10 on page 105.

Repeat step Sub1-I and Sub1-J till all the possible change-paths are generated as shown in Figure 5-17 (page 113). Figure B-4 shows a flowchart for the subroutine.

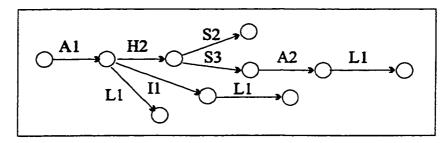


Figure B-2 Result of applying steps Sub1-I and Sub1-J on example given in Figure 5-10.

B.2 SUBROUTINE TO GENERATE THE LOGICAL SEQUENCE OF DESIGN CHANGES.

The logic for developing this subroutine is based on providing a start node value and an end node value for each of the design changes in a certain change-path (Figure 5-18 on page 114). This is to satisfy the requirement for using a CPM algorithm that can schedule the design changes. The following rules apply when defining the values for the start node and the end node:

- The value of the start node of a design change is always less than the value of its end node.
- The value of the start node of a successor to a design change is the same value of the

end node of the design change.

- More than one design change can share the same value for the start node (i.e. H2 and
 I1 in Figure 5-18b on page 114).
- The value of the end node is always unique. This rule is not required for using the CPM algorithm, rather to support the logic of the subroutine.
- Dummy design changes are used (shown as dotted lines with arrow heads in Figure 5-18) to reflect the case where two design changes share the same successor (i.e. I1 and A2 in Figure 5-18b).

Figure B-3 represents a matrix that is used to represent the logic for the developed subroutine and reflects the previous rules. The vertical values of the matrix indicate the start nodes of design changes while the horizontal values indicate the end nodes. Figure B-3 reflects the change-path in Figure 5-18b.

Values of start nodes

Values of end nodes

		5			6.6	
2.	A1					
		H2				
A STATE OF THE STA		I1				
			S2			
			S3			
				_	A2	
						L1

Figure B-3 Matrix to represent start nodes and end nodes for change-path in Figure 5-18b.

B.2.1 INITIALIZATIONS

- Sub2-A Connect to Data-Table 5-8 (page 113) of the related project-database.
- Sub2-B Limit the reading of the data from Data-Table 5-8 to the rows that have the specified value for the attribute "CASE ID" (i.e. CASE ID = 2).
- Sub2-C Create two lists for "Processed design changes" and "Under processing design changes". Each element in these lists has attributes for the value of the start node and the value of the end node.

Repeat the following steps for each group of design changes in Data-Table 5-8 that has the same "ALTERNATIVE#" (i.e. Alternative#1).

B.2.2 Process the initial design change

- Sub2-D Query Data-Table 5-8 (page 113) for the data row that have the value "NULL" in the attribute "DESIGN_CHANGE" and read the value of the attribute "SUCCESSOR" in the found row. This value is the initial design change that has been defined when a designer starts a new case (i.e. A1 in Figure 5-17 on page 113). Add this design change to the list of "Processed design changes" and the list of "Under processing design changes".
- Sub2-E Make the value of the start node = 1 for the initial design change and make a unique consecutive value (i.e. 2) for the end node.

B.2.3 Process the rest of the design changes in the change-path.

For each design change in the list of "Under processing design changes" perform the steps Sub2-F to Sub2-I:

- Sub2-F Read the value of the end node for this design change (i.e. 2 for the design change A1). Assign this value to the variable (e).
- Sub2-G Get from Data-Table 5-8 on page 113 all the successors of the design change (i.e. H1, L1, I1 in Figure 5-17 on page 113).

For each successor, perform steps Sub1-H to Sub1-I:

- Sub2-H Read from Data-Table 5-8 (page 113) all the design changes that have this successor. (i.e. design changes A1 and I1 for the successor L1).
- Sub2-I Check if all the found design changes in Sub2-H (i.e. A1 and I1) exit in the list of "Processed design changes".
 - If any of the found design changes does not exist then ignore this successor. (i.e. L1)
 - If all of the found design changes exit then add this successor to the list of "Processed design changes" and the list of "Under processing design changes". Make the value of the start node for this successor equals the value of the end node of the under processing design change. (i.e. the value of the variable (e) which is defined in step Sub2-F) and use a unique consecutive value for the end node of this successor. Using unique consecutive value for the end nodes makes use of all the rows in the matrix of Figure B-3 and ensures that only one design change exists in every row hence reflecting the stated rules at the beginning of section B.2 on page 182.

B.2.4 GENERATE THE DUMMY DESIGN CHANGES

- Sub2-J For the value of a variable (i) that starts from the value (1) to the highest value used as a start node in the processed design change (i.e. 1 to 7 in Figure B-3 on page 183), check if the value is used as a start node in any of the processed design change. This is similar to searching for the empty columns in the matrix in Figure B-3.
 - If the value of (i) is used (e.g. 1), then check the next value.
 - If the value of (i) is not used (e.g. 4), then continue the following steps.
- Sub2-K Find the design change in the list of processed design changes that have the same value of (i) for its end node (e.g. II). This should be a single design change as the end node is a unique value for every design change.

 This is demonstrated in the matrix in Figure B-3 by having only one design change in any row. The matrix also shows that there are no empty rows.
- Sub2-L Query Data-Table 5-8 (page 113) for all the successors of the found design (e.g. I1) change and determine from the list of processed design changes the successor that has the lowest start node value (i.e. L1). Assign this value (i.e. 7) to the variable (ii).
- Sub2-M Create a dummy design change that has a start node value equals the value of (i) (i.e. 4) and an end node value equals the value of (ii) (i.e. 7). Add the dummy design change to the list of processed design changes.

After generating all the dummy design changes Figure 5-18 on page 114, arrange all the design changes in the processed design change list according to their start node value. The design change of this change-path is ready to be scheduled using a standard CPM algorithm. Figure B-5 shows a flowchart for the described subroutine.

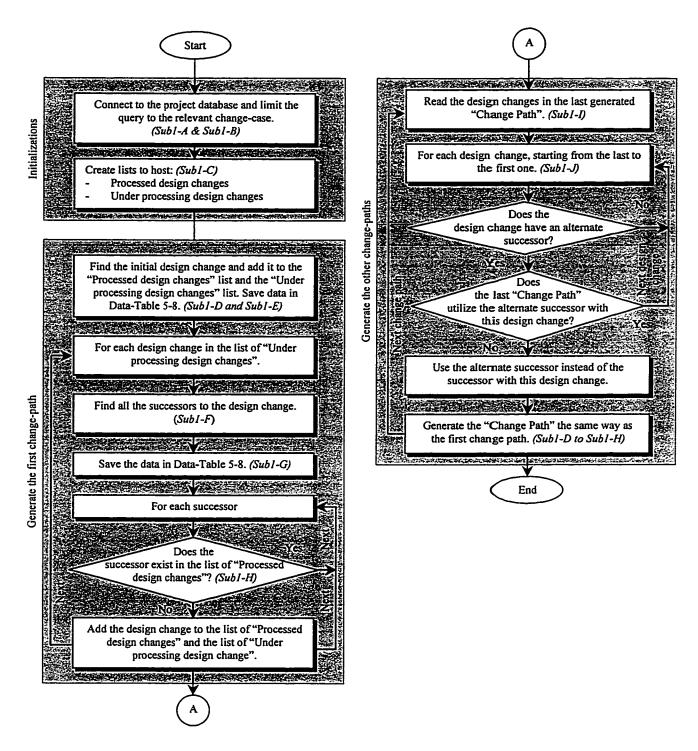


Figure B-4 Flowchart for sub-routine developed to generate various changepaths.

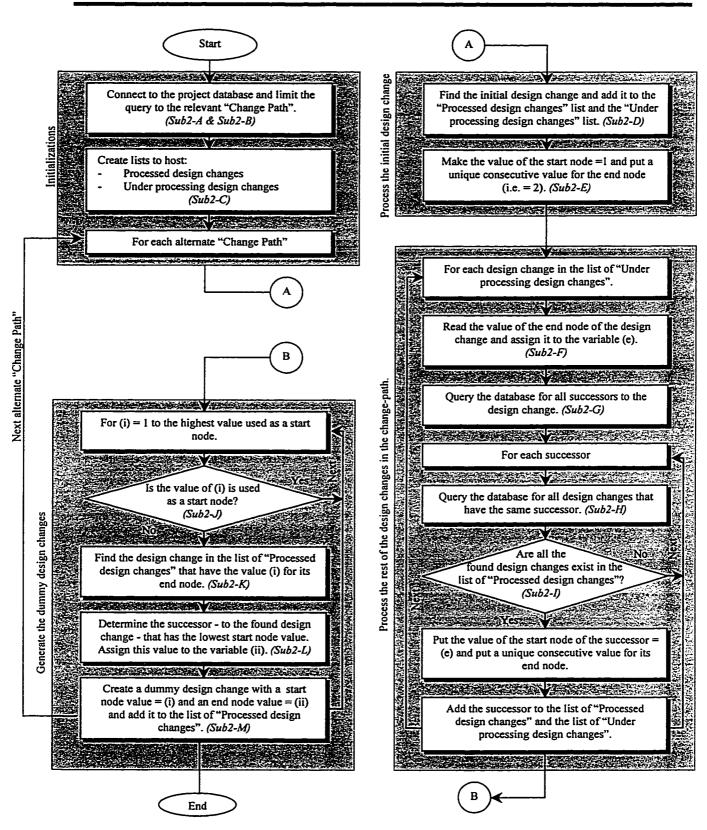


Figure B-5 Flow chart for sub-routine developed to define sequence of performing design changes so as to eliminate redundancy.

LISTINGS OF LEVEL5 OBJECT MODULES

C.1. CONFIGURATION MODULE

building_element_attribute.linked_building_element END SQL INTO domain (tt)

```
SVERSIONIS
                                                                                                                                                     EXEC (Sodbe 1 SQL SELECT * from (AB-
                                                                                   linked_attribute_of_building_element OF
 SLOCATIONS ARE PIXELS
                                                                        building_element_attribute := tz
                                                                                                                                               attribute-designer] where building_element
                                                                                                                                               :Current_rule_data_building_elemen
END_SOL_INTO
                                                                                  END
 CLASS AB_general_rules
                                                                               END
  WITH AB general rules ID NUMERIC
                                                                                                                                               building_element_attribute
  WITH building_element STRING
                                                                              FORGET attribute type OF input data
                                                                                                                                                   END
  WITH attribute of building element STRING
WITH affected discipline STRING
WITH percent change NUMERIC
                                                                                                                                                WITH attribute of building element STRING WHEN CHANGED
                                                                              FORGET discipline OF input data
                                                                                                                                                   BEGIN
  WITH result in major modification STRING
WITH cause of the effect STRING
WITH rule_ignored STRING
                                                                                                                                                    EXEC ISodbe I SQL SELECT . from [AB-
                                                                         WITH attribute of building element STRING WHEN CHANGED
                                                                                                                                                   eral-rules] where building_eleme
                                                                                                                                               Current rule data building element AND
                                                                            BEGIN
                                                                       IF (CONF(attribute_of_building_element OF current_data) >= 0) THEN
                                                                                                                                               attribute_of_building_element =
                                                                                                                                               :Current_rule_data_attribute_of_building_element
END SQL INTO AB_general_rules
CLASS border INHERITS add on, display item
  WITH style COMPOUND
                                                                               BEGIN
   picture frame,
                                                                                attribute_type OF current_data := attribute_type
    edit control,
                                                                        OF building_element_attribute
                                                                                                                                                WITH affected_discipline STRING
                                                                                                                                                WITH result is major modification STRING
WITH cause of the effect STRING
WITH percent_change NUMERIC
WITH rule_ignored STRING
                                                                                designing discipline OF current_data :"
   group,
                                                                       designing discipline OF building element attribute linked building element OF current data:
   INIT picture frame
  WITH perspective COMPOUND
                                                                        linked building element OF
                                                                       building element attribute
linked attribute_of building element OF
current_data := linked_attribute_of_building_elem
                                                                                                                                              CLASS edit table INHERITS add on, display item
WITH attachment CLASS REFERENCE
WITH columns INSTANCE REFERENCE
   INIT is
                                                                       OF building element attribute
FORGET attribute type OF input data
  WITH border width NUMERIC
                                                                                                                                                 COLLECTION
                                                                                                                                                WITH heading SIMPLE
INIT TRUE
  WITH background color COLOR.
                                                                                FORGET discipline OF input data
   INIT 192 192 192
                                                                              END
                                                                                                                                                WITH heading height NUMERIC
WITH row height NUMERIC
WITH fill color COLOR
  WITH fill color COLOR
                                                                         WITH attribute_type STRING
WITH designing_discipline STRING
WITH linked_building_element STRING
   INIT 192,192,192
  WITH highlight color COLOR
INIT 255,255,255
                                                                                                                                                WITH row separator SIMPLE
  WITH shadow color COLOR
                                                                          WHEN CHANGED
                                                                                                                                                 INIT TRUE
   INIT 128,128,128
                                                                                                                                                WITH column separator SIMPLE
                                                                            BEGIN
                                                                             IF ( CONF(linked_building_element OF
                                                                                                                                                 INIT TRUE
                                                                        current_data) >= 0) THEN
                                                                                                                                                WITH max rows NUMERIC
CLASS Building element
  WITH building element STRING
WITH graph name STRING
WHEN CHANGED
                                                                               BEGIN
                                                                                                                                                WITH total rows NUMERIC
                                                                                                                                                WITH current cell NUMERIC
ARRAY SIZE 2
                                                                                EXEC ISodbe | SQL SELECT * from [AB-
                                                                       attribute-designer] where building_element == :current_data.linked_building_element
                                                                                                                                                 INIT[1]1
     BEGIN
                                                                                                                                                 NIT (2 ) 1
      graph_name OF current_data := graph_name OF
                                                                                            END SQL INTO Linked element
Building element
                                                                        attribute
                                                                                                                                                WITH current cell data STRING
    END
                                                                                                                                                WITH highlight text color COLOR
                                                                                FIND Linked building element
                                                                                                                                                WITH highlight fill color COLOR
                                                                                                                                                WITH display only SIMPLE
WITH selected SIMPLE
CLASS building_element_attribute
WITH attribute_of_building_element STRING
                                                                       WHERE building_element OF Linked building element = linked_building_element OF
  WITH attribute_type STRING
                                                                        current_data
                                                                                                                                                WITH double clicked SIMPLE
  WITH designing discipline STRING
WITH linked building element STRING
                                                                                 FIND FND
                                                                                                                                              CLASS edit table column
  WITH linked_attribute_of_building_element STRING
                                                                                linked_building_element OF
                                                                                                                                                WITH attachment ATTRIBUTE REFERENCE
                                                                        building_element_attribute := linked_building_element
                                                                                                                                                WITH heading label STRING
                                                                                                                                                WITH heading style INSTANCE REFERENCE
WITH column style INSTANCE REFERENCE
WITH width NUMERIC
CLASS current_data
                                                                        OF current_data
  WITH building_element STRING
WHEN CHANGED
                                                                              END
                                                                                                                                                WITH max characters NUMERIC
                                                                                                                                                WITH format STRING
WITH justify COMPOUND
                                                                         WITH linked attribute of building element STRING WHEN CHANGED
      EXEC ISodbc 1 SQL SELECT * from [AB-
attribute-designer] where building_element :current_data.building_element
                                                                       IF ( CONF(linked_attribute_of_building_element
OF current_data) >= 0) THEN
                    END SQL INTO
                                                                                                                                                  center.
building_element_attribute
                                                                                                                                                  right
                                                                               BEGÎN
                                                                                                                                                  INIT left
      FOR (i := 1 TO number instances OF navigate
                                                                                linked_attribute_of_building_element OF
                                                                                                                                                WITH pen color COLOR COLLECTION
1)
                                                                        building_element_attribute := linked_building_element
                                                                                                                                                WITH fill color COLOR
                                                                        OF current_data
                                                                              END
                     umber OF anvigate 1 := i
                                                                                                                                                  COLLECTION
IF (CONF(linked_building_element OF building_element_attribute) >= 0) THEN BEGIN
                                                                            END
                                                                          WITH graph_name STRING
                                                                                                                                              CLASS GENERAL_Building_element
                                                                                                                                                WITH building_element STRING
WITH graph_name STRING
           EXEC ISodoc I SQL SELECT
                                                                       CLASS Current_projects
WITH project_name STRING
EARL 19000C 1 SQL SELECT
linked_uttribute_of_building_element from (AB-
attribute-designer] where building_element =
current_data_building_element =
linked_building_element =
                                                                                                                                              CLASS GENERAL_building_element_attribute
WITH attribute_of_building_element STRING
                                                                        CLASS Current_rule_data
                                                                          WITH building element STRING WHEN CHANGED
                                                                                                                                                WITH attribute_type STRING
```

BEGIN

WITH designing_discipline STRING WITH linked_building_element STRING

		•
WITH linked_attribute_of_building_element STRING	BEGIN	EXEC ISodbe I SQL INSERT into [AB-
C7 456 : 4	attribute_type OF current_data := "t"	menu_item] (label, main_menu_label) values
CLASS input data	attribute_type OF building_element_attribute	1 4
WITH discipline COMPOUND Architect	:= ℃ END	:Menus_manipulation_data.current_menu_B_comp,
~ Structure.	i END	:Menus_manipulation_data.current_menu_group) END SOL
HVAC,	IF attribute type OF input data IS Numeric data	1
Duminat	THEN	EXEC ISodbc SQL SELECT label from [AB-
Interior.	BEGIN	menu_item] WHERE main_menu_label =
Envelope	attribute_type OF current_data := "n"	:Menus manipulation data current menu group
WHEN CHANGED	attribute_type OF building_element_attribute	END SQL INTO
BEGIN	:= "a"	Menus_group_items
1	END	
IF (CONF(discipline OF input data) >= 0) THEN	'	FIND Menus_unclassified_B_comp
BEGIN	END	WHERE label OF
IF dissiplies OF insure days IS Applicate TRIFAL	1 EDE	Menus_unclassified_B_comp = current_menu_B_comp
IF discipline OF input data IS Architect THEN BEGIN	END	OF Menus_manipulation_data WHEN FOUND
designing discipline OF	CLASS ISodbe INHERITS add on	FORGET CURRENT
building_element_attribute := "arch"	WITH data source STRING	Menus_unclassified_B_comp
designing discipline OF current data :=	WITH userid STRING	FIND END
"arch"	WITH password STRING	!
affected_discipline OF AB_general_rules :=	WITH connection string STRING	FORGET current_menu_B_comp OF
"arch"	WITH connection prompting SIMPLE	Menus_manipulation_data
END	INIT FALSE	!
	WITH auto commit SIMPLE	! *** end of IF
IF discipline OF input data IS Structure THEN	INIT TRUE	END
BEGIN	WITH action COMPOUND	ELSE BEGIN
designing_discipline OF	connect, disconnect	
building_element_attribute := "stru" designing_discipline OF current_data :=	WITH records NUMERIC	text OF message box 2 := "A selected Menu Group or Building Component is missing"
"stru"	WITH transaction COMPOUND	put up OF message box 2 := TRUE
affected_discipline OF AB_general_rules :=	commit.	END
"stru"	rollback	
END	WITH append SIMPLE	ĺ
1	INIT FALSE	! *** end of method
IF discipline OF input data IS HVAC THEN	WITH status STRING	END
BEGIN	WITH show error SIMPLE	WITH delete build comp SIMPLE
designing_discipline OF	WITH default error handling SIMPLE	WHEN CHANGED
building_element_attribute := "hvac"	INITTRUE	BEGIN
designing_discipline OF current_data :=	WITH error message STRING	# CONF(P OF
"kvac"	WITH native error NUMERIC WITH trace file STRING	IF CONF(current_menu_B_comp OF Menus_manipulation_data) > 0 AND CONF(
affected_discipline OF AB_general_rules := "hvac"	WILL DROE DR 2140140	current_menu_group OF Menus_manipulation_dats) >
END .	CLASS Linked building element	OTHEN
1	WITH building element STRING	BEGIN
IF discipline OF input data IS Illuminat THEN		1
BEGIN	CLASS Linked element attribute	EXEC ISodbe 1 SQL DELETE from [AB-
designing_discipline OF	WITH attribute_of_building_element STRING	menu_item] WHERE label =
building_element_attribute := "illu"		:Menus_manipulation_data.current_menu_B_comp
designing_discipline OF current_data :=	CLASS Menus_classified_B_comp	END SQL
"illu"	WITH label STRING	!
affected_discipline OF AB_general_rules :=	CT 10014	EXEC 15odbc 1 SQL SELECT label from [AB-
"illu" END	CLASS Menus group	menu_item] WHERE main_menu_label =
i	WITH label STRING	:Menus_manipulation_data.current_menu_group END SOL INTO
IF discipline OF input data IS Interior THEN	CLASS Menus_group_items	Menus_group_items
BEGIN	WITH label STRING	!
designing discipline OF		MAKE Menus unclassified B comp
building_element_attribute := "inte"	CLASS Menus_manipulation_data	WITH label := current_menu_B_comp OF
designing_discipline OF current_data :=	WITH current_menu_group STRING	Menus_manipulation_data
"inte"	WHEN CHANGED	!
affocted_discipline OF AB_general_rules :=	BEGIN	FORGET current_menu_B_comp OF
"inte"	F CONT(OF	Menus_manipulation_data
END 1	IF CONF(current_menu_group OF Menus manipulation data) > 0 THEN	! *** end of IF
IF discipline OF input data IS Envelope THEN	BEGIN	END
BEGIN	EXEC ISodbe 1 SQL SELECT label from [AB-	ELSE
designing discipline OF	menu_item] WHERE main_menu_label =	BEGIN
building_element_attribute := "enve"	:Menus manipulation data current menu group	text OF message box 2 := "A selected Menu
designing discipline OF current_data :=	END SQL INTO	Group or Building Component is missing*
"enve"	Menus_group_items	put up OF message box 2 := TRUE
affected_discipline OF AB_general_rules :=	. !	END
"enve"	FORGET current_menu_B_comp OF	1 1
END	Menus_manipulation_data	! *** end of method
! END	, DD	END
!	ÉND '	WITH add menu group SIMPLE WHEN CHANGED
END	! *** end of method	BEGIN
WITH attribute type COMPOUND	END	I BEGIN
Text data,	WITH current_menu_B_comp STRING	IF CONF(current_menu_B_comp OF
Numeric data	WITH add build comp SIMPLE	Menus_manipulation_data) > 0 AND CONF(
WHEN CHANGED	WHEN CHANGED	new_name_of_menu_group OF
BEGIN .	BEGIN	Menus manipulation data) > 0 THEN
Transfer to the second	1	BEGIN
IF (CONF(attribute type OF input data) >= 0)	IF CONF(current_menu_B_comp OF	PACCIC- A- LOOK DICEPT : (40
THEN BEGIN	Menus_manipulation_data) > 0 AND CONF(EXEC ISodbc 1 SQL INSERT into [AB-
!	current_menu_group OF Menus_manipulation_data) > 0 THEN	menu_item] (label, main_menu_label) values
! IF attribute type OF input data IS Text data.	BEGIN	(:Menus_manipulation_data.current_menu_B_comp,
THEN	1	Menus manipulation data new name of menu group

)	Group is missing*	cancel,
END SQL	put up OF message box 2 := TRUE	ignore,
<u> </u>	END	no.
EXEC ISodbe 1 SQL SELECT label from [AB-	†	ok.
menu_item] WHERE main_menu_label =	! *** end of method	retry,
:Menus_manipulation_data.new_name_of_menu_group	END	yes
END SQL INTO	WITH rename menu group SIMPLE	WITH put up SIMPLE
Menus_group_items	WHEN CHANGED	
1	BEGIN	CLASS navigate INHERITS add on
FIND Menus_unclassified_B_comp	Į į	WITH class attachment CLASS REFERENCE
WHERE label OF	IF CONF(new_name_of_menu_group OF	WITH action COMPOUND
Menus_unclassified_B_comp = current_menu_B_comp	Menus manipulation data) > 0 AND CONF(first,
OF Menus_manipulation_data	current_menu_group OF Menus_munipulation_data) >	previous
WHEN FOUND	0 THEN	next.
FORGET CURRENT	BEGIN	last
Menus_unclassified_B_comp	! !	WITH instance number NUMERIC
FIND END "	FOR (i := 1 TO number instances OF navigate	WITH number instances NUMERIC
!	menu group items)	
MAKE Menus_group	BEGIN	CLASS NEW building systems
WITH label := new_name_of_menu_group	instance number OF navigate	WITH building system STRING
OF Menus_manipulation_data	menu group items := i	WITH description STRING
1	EXEC ISodbe 1 SQL update [AB-	WITH discipline STRING
FORGET current_menu_B_comp OF	menu_item] set	William Stare
Menus manipulation data	main menu label =	CLASS NEW_design_disciplines
FORGET current menu_group OF	:Menus_manipulation_data.new_name_of_menu_group	WITH discipline STRING
Menus_manipulation_data	WHERE label =	Ct 400 \ TTM
FORGET new_name_of_menu_group OF	:Menus_group_items.label	CLASS NEW_other_project_components
Menus_manipulation_data	END SQL	WITH building_element STRING
1	END	
! *** end of IF	1	CLASS NEW_to_make_a_new_project
END	FIND Menus_group	WITH from another project SIMPLE
ELZE	WHERE label OF Menus_group *	WITH from systems SIMPLE
BEGIN	current_menu_group OF Menus_manipulation_data	WHEN CHANGED
text OF message box 2 := "Type a New Menu	WHEN FOUND	BEGIN
Group and select a Building Component*	label OF Menus group :=	EXEC 15odbc 2 SQL select distinct (discipline)
put up OF message box 2 := TRUE	new_name_of_menu_group OF	from [building_systems]
END	Menus_manipulation_data	END SQL INTO NEW_design_disciplines
1	FIND END	1
i	1	action OF ISodbc 1 IS connect := TRUE
! *** end of method	current menu group OF	1
END	Menus_manipulation_data :=	visible OF main window := FALSE
WITH delete menu group SIMPLE	new_name_of_menu_group OF	visible OF configuration window := TRUE
WHEN CHANGED	Menus manipulation data	output OF configuration window := NEW from
BEGIN	FORGET new_name_of_menu_group OF	systems display
		END
T 0017/2 2 2 2	Menus_manipulation_data	
IF CONF(current_menu_group OF	! *** end of IF	WITH from components SIMPLE
Menus_manipulation_data) > 0 THEN	END	WHEN CHANGED
BEGIN	ELSE	BEGIN
1	BEGIN	EXEC ISodbc 2 SQL select distinct
EXEC ISodbe SQL DELETE from [AB-	text OF message box 2 := "A selected Menu	(building_element) from [General-AB-attribute-
menu_item} WHERE main_menu_label =	Group or a New name for the Menu Group is missing*	designer]
:Menus_manipulation_data.current_menu_group	put up OF message box 2 := TRUE	END SQL INTO Building element
END SQL	END	1
1	1	action OF ISodbc 1 IS connect := TRUE
FORGET current_menu_B_comp OF	! *** end of method	i e
Menus_manipulation_data	END	visible OF main window := FALSE
FORGET current menu group OF	WITH new_name_of_menu_group STRING	visible OF configuration window := TRUE
Menus manipulation data		output OF configuration window := NEW from
FORGET Menus group_items	CLASS Menus_unclassified_B_comp	components display
!	WITH label STRING	END
EXEC ISodbe I SQL SELECT distinct	***************************************	WITH another_project_name STRING
(main_menu_label) from [AB-menu_item]	CLASS message box INHERITS add on	WHEN CHANGED
END SQL INTO Menus group	WITH title STRING	BEGIN
(label)	WITH text STRING	,
!	WITH variable text STRING	! **** this part will do the following
EXEC ISodbe 1 SQL SELECT label from [AB-	WITH modal style COMPOUND	! **** 1) make connection to the other project database
		! **** 2) read the components from the table AB-
menu_item] END SOL INTO	app modal,	attribute-designer form the database of the
	sys modal,	
Menus_classified_B_comp	not modal	1 0000 Other project
	INIT app modal	! **** 3) create tables in the new project database
FOR (i := 1 TO number instances OF navigate	WITH icon COMPOUND	[**** 4) insert the tables data into the table AB-
Building element)	exclamation,	attribute-designer of the new table
BEGIN	information,	! **** 5) read the general rules and menus from the old
instance number OF navigate Building	question,	project database and put them
element := i	atob agu	in the database of the new project database
FIND Menus_classified_B_comp	INIT exclamation	
WHERE label OF	WITH button COMPOUND	! connect to the new project database
Menus_classified_B_comp = building_element OF	abort retry ignore,	action OF ISodbc 1 IS connect := TRUE
Building element	ok,	Į. Į.
WHEN NONE FOUND	ok cancel,	! make connection with the database of the
MAKE Menus_unclassified_B_comp	retry cancel,	other project
WITH label := building_element OF	yes no.	data source OF ISodbe 3 := another_project_name
Building element	yes no cancel	OF NEW to make a new project
FIND END	INIT ok	action OF ISodoc 3 IS connect := TRUE
END	WITH default button COMPOUND	(
		! read the components from the other project
! *** end of IF	one,	EXEC 15odbc 3 SQL SELECT distinct
END STU	two,	
	three	(building_element), graph_name from [AB-attribute-
ELSE	INIT one	designer]
BEGIN	WITH button selected COMPOUND	END SQL INTO Building element FOR (i := 1 TO number instances OF navigate
text OF message box 2 := "A selected Menu	abort	

```
attribute_of_building_element =
                                                                                                                                                                                  WHEN CHANGED
Building element)
BEGIN
                                                                                        :building_element_attribute.attribute_of_building_elem
                                                                                                                                                                                   BEGIN
           instance number OF navigate Building elen
                                                                                                                                                                                    i = 0
                                                                                                                                                                                     CONF(n) := -2
                                                                                                                END SOL INTO domain (ttt)
       ! *** create the database table with its id attribute
                                                                                                         linked attribute of building element OF
general statement of sql := CONCAT(
"CREATE TABLE [", building_element OF Building element, "] [[", building_element OF Building element of Buildi
                                                                                                                                                                               *** check that there is an inserted name and that this
                                                                                       building_element_attribute := tttt
                                                                                                                                                                             name does not exist
                                                                                                                                                                                     IF ( CONF(building element OF
                                                                                                         EXEC ISodbe 1 SQL INSERT INTO [AB-
                                                                                                                                                                             NEW_to_make_s_new_project) >= 0) THEN
BEGIN
 '_id] TEXT(S0));")

EXEC |Sodbc | SQL WITH general statement
                                                                                       attribute-designer] (building element,
attribute_of_building_element, attribute_type,
designing_discipline, linked_building_element,
of sql END SQL
                                                                                                                                                                                        EXEC ISodbo 1 SQL select distinct
         END
                                                                                       linked attribute of building element) values
                                                                                                                                                                             building element from [AB-attribute-designer]
                                                                                                        ( :domain.tt.
                                                                                        :building_element_attribute.attribute_of_building_elem
                                                                                                                                                                                               where building_element =
              - read the attributes of the components from
                                                                                                                                                                             NEW_to_make_a_new_project.building_el
the other project
                                                                                        ent, :building_element_attribute.attribute_type,
       FOR (i := 1 TO number instances OF navigate
                                                                                        building element attribute designing discipline, build
                                                                                                                                                                                               END SQL INTO domain (tt)
                                                                                       ing element attribute linked building element buildin
         ng ele
        BEGIN
                                                                                       g_element_attribute.linked_attribute_of_building_elem
                                                                                                                                                                                      END
          instance number OF navigate Building element
      ! **** the next step is to overcome the n
                                                                                                                                                                                    IF CONF(tt) = 100 THEN
"Building element" which can not be linked to Access
tt:= building_element OF Building element
EXEC ISodbe 3 SQL SELECT * from [AB-
                                                                                                                                                      END
                                                                                       SOL
                                                                                                                                                                                    ELSE
                                                                                                      END
                                                                                                     ELSE
attribute-designer] where building_element =
                                                                                                       BEGIN
:domain.tt
                         END SQL INTO
                                                                                                        EXEC Isodbc 1 SQL INSERT INTO [AB-
                                                                                       attribute-designer] (building_element,
attribute_of_building_element, attribute_type,
building element attribute
FOR (ii := 1 TO mu
                                                                                                                                                                                    IF i = I THEN
                                             er instances OF
                                                                                                                                                                                      BEGIN
navigate building element attributes)
BEGIN
                                                                                       designing discipline) values
                                                                                                                                                                                       text OF message box 2 := "There is already a
                                                                                                        (:domain.tz.
                                                                                        building element attribute attribute of building elem
                                                                                                                                                                             building component with that name
             instance number OF navigate building
                                                                                                                                                                                        put up OF message box 2 := TRUE
element attributes := ii
! **** ignore the attribute that has the _id of the
                                                                                       ent, :building element_attribute.attribute_type, :building element_attribute.designing_discipline)
                                                                                                                                                                                      END
             IF SEARCHSTR("_id",
attribute of building element OF
building element attribute) = 0 THEN
                                                                                                                                                      END
                                                                                                                                                                             1 ***** If the name was not used before
                                                                                       SOL
                                                                                                      FND
               BEGIN
          ! **** put the attributes in the database
                                                                                                   END
                                                                                                                                                                                    IF i = 2 THEN
general statement of sqi := CONCAT(
"alter table [", building_element OF Building element
                                                                                               END
                                                                                                                                                                                      REGIN
"] add [", attribute_of_building_element OF
building_element_attribute, "] ")
                                                                                                                                                                             1 **** create the database table with its id attribute
                                                                                                     get the general rules from the old database
                                                                                                                                                                             general statement of sql := CONCAT(
"CREATE TABLE [", building_element OF
                                                                                       and save them in the new one (ignore the id attribute)
EXEC ISodbe 3 SQL SELECT * from [AB-
                 IF attribute_type OF
                                                                                                                                                                             NEW_to_make_a_new_project, "] ([", building element OF NEW_to_make_a_new_project, "_id] TEXT(50));")
building_element_attribute = "t" THEN
general statement of sql := CONCAT(
                                                                                             eral-rules]
general statement of sql, "TEXT (50);")
                                                                                                               END SOL INTO AB repent rules
                                                                                                                                                                                       EXEC ISodbe 1 SQL WITH general statement
IF attribute_type OF building_element_attribute = "n" THEN
                                                                                              FOR (i := 1 TO number instances OF rules
                                                                                                                                                                             of sol END SQL
                  general statement of sql := CONCAT(
                                                                                       navigate 2)
general statement of sql, "Number;")

EXEC ISodbe 1 SQL WITH general
                                                                                               BEGIN
                                                                                                                                                                               **** add the rest of attributes to the created table
                                                                                                                                                                             FOR (ii := 1 TO number instances OF navigate building element attributes)
                                                                                                 instance number OF rules navigate 2 := i
statement of sqi END SQL
                                                                                                 EXEC ISodbe | SOL INSERT INTO [AB-
                                                                                                                                                                                        BEGIN
                                                                                       general-rules] (building element,
               END
                                                                                                                                                                                          instance number OF navigate building
                                                                                       attribute of building element, affected discipline,
                                                                                       percent_change, result_in_major_modification,
cause_of_the_effect) values
                                                                                                                                                                             element attributes := ii
            END
                                                                                                                                                                                       ! **** ignore the attribute that has the _id of the
        END
                                                                                                         (:AB_general_rules.building_element,
                                                                                                                                                                                          IF SEARCHSTR("_id",
                                                                                       :AB general rules attribute of building element, :AB general rules affected discipline,
           insert the tables data into the table AB-
                                                                                                                                                                             attribute of building element OF
attribute-designer of the new table
                                                                                                                                                                             building_element_attribute) = 0 THEN
       FOR (i := 1 TO number instances OF navigate
                                                                                        :AB_general_rules.percent_change,
                                                                                       :AB general rules result in major modification,
:AB general rules cause of the effect)
                                                                                                                                                                                            BEGIN
Building element)
                                                                                                                                                                                        ! **** put the attributes in the database
        BEGIN
                                                                                                                                                                             general statement of sql := CONCAT(

"alter table [", building_element OF Building element,
          instance number OF navigate Building element
                                                                                                                                                                              add [". attribute_of_building_element OF
       ! **** the next step is to overcome the name
                                                                                                                                                      END
                                                                                                                                                                             building_element_attribute, "] ")
IF attribute_type OF
"Building element" which can not be linked to Access
                                                                                       SQL
                                                                                               END
          tt := building_element OF Building element
                                                                                                                                                                             building_element_attribute = "t" THEN
general statement of sql := CONCAT(
          EXEC ISodbc 3 SQL SELECT * from [AB-
                                                                                                                                                                             general statement of sql, "TEXT (50);")
attribute-designer] where building_element =
                                                                                                    to activate the push bottons
                                                                                                                                                                                              IF attribute_type OF
:domain.tt
                                                                                                                                                                             building element attribute = "n" THEN
                         END SQL INTO
                                                                                              attribute attachment OF pushbutton 42 := Building
                                                                                                                                                                                               general statement of sql := CONCAT(
building_element_attribute
                                                                                       Components
                                                                                                                                                                                                 ent of sql, "Number,"
                                                                                              attribute attachment OF pushbutton 43 := Menus
                                                                                                                                                                             general staten
                                                                                                                                                                                              EXEC ISodbe 1 SQL WITH general
          FOR (ii := 1 TO number instances OF
                                                                                               attribute attachment OF pushbutton 44 := General
                                                                                                                                                                             statement of sql END SQL
navigate building element attributes)
                                                                                       Rules
                                                                                                                                                                                            END
            BEGIN
                                                                                                                                                                                          END
              instance number OF navigate building
                                                                                          **** end of method
                                                                                            END
element attributes := ii
              IF CONF(linked_building_element OF
                                                                                          WITH building_element STRING
                                                                                                                                                                                *** add the attributes to the table [AB-attribute-
                                                                                           WHEN CHANGED
building_element_attribute) = 100 THEN
                                                                                                                                                                             designer) in the new project
                                                                                                                                                                                       FOR (ii := 1 TO number instances OF
               BEGIN
                                                                                             BEGIN
                                                                                              EXEC ISodbe 2 SQL SELECT * from [General-
                                                                                                                                                                             navigate building element attributes)
                                                                                       AB-attribute-designer] where building_element =
:NEW_to_make_a_new_project.building_element
END SQL_INTO
                                                                                                                                                                                          BEGIN
              ***** the following step is necessary as data
can not be got directly from access to this attribute
                                                                                                                                                                                           instance number OF navigate building
                 EXEC Isodbe 3 SQL SELECT
                                                                                                                                                                             element attributes := ii
                                                                                                                                                                                          IF CONF(linked building element OF
(linked_attribute_of_building_element) from [AB-
                                                                                       building_element_attribute
                                                                                                                                                                             building_element_attribute) = 100 THEN
BEGIN
attribute-designer]
                                                                                            END
```

WITH add component to the new project SIMPLE

where building_element = :domain.tt and

```
AB_general_rules.affected_discipline,
                                                                                                                                                IF SEARCHSTR("_id",
           ***** the following step is necessary as data
                                                                   AB_general_rules.percent_change,
                                                                                                                                   attribute_of_building_element OF
                                                                  :AB general rules result in major modification, :AB general rules cause of the effect)
 can not be got directly from access to this attribute
                                                                                                                                  building_element_attribute) = 0 THEN
              EXEC Bodbe 2 SQL SELECT
                                                                                                                                                 BEĞIN
(linked_attribute_of_building_element) from [GENERAL-AB-attribute-designer]
                                                                                                                                          ! **** put the attributes in the datab
                                                                                                                                   general statement of sql := CONCAT(
"alter table [", building element OF Building element,
             where building elen
                                                                                                                 END
:NEW to make a new project building element and antibute of building element
                                                                  SOL
                                                                                                                                   "] add [", attribute_of_building_element Of
                                                                           END
                                                                                                                                  building_element_attribute, "] ")
 building element attribute attribute of building elem
                                                                                                                                                  IF attribute_type OF
                                                                        END
                                                                                                                                  building_element_attribute = "t" THEN
                   END SQL INTO domain (tttt)
                                                                                                                                                   general statement of sql := CONCAT(
              linked_attribute_of_building_element OF
                                                                    **** move to the
                                                                                                                                  general statement of sql, "TEXT (50);")
                                                                                                                                                 IF attribute_type OF
building_element_attribute := tm
                                                                  CURRENT_project_selection_display to allow more
                                                                                                                                  building element attribute = "n" THEN
                                                                  configuration
              EXEC ISodbe I SQL INSERT INTO [AB-
                                                                       visible OF configuration window := FALSE
                                                                                                                                                  general statement of sql := CONCAT(
                                                                                                                                                 ent of sql, "Number;")
EXEC ISodbe 1 SQL WITH general
                                                                                                                                  general statem
attribute-d
              igner] (building_ele:
                                                                       visible OF main window := TRUE
attribute of building element, attribute type,
                                                                       output OF main window :=
 designing discipline, linked building eler
                                                                  CURENT_project_selection_display
                                                                                                                                  statement of sql END SQL
linked_attribute_of_building_element) values
                                                                                                                                                ĖND
                                                                    end of method
                                                                                                                                              END
                                                                                                                                   **** add the attributes to the table [AB-attribute-
 :NEW_to_make_a_new_project.building_e
                                                                     END
PLEW 10 make a new project outnoing element, building element attribute attribute of building element attribute attribute type, building element attribute designing discipline building element attribute inked building element, building element attribute inked building element, building element.
                                                                   WITH design_discipline STRING
WHEN CHANGED
                                                                                                                                  designer] in the n\
                                                                                                                                  ew project
                                                                                                                                            FOR (ii := 1 TO number instances OF
                                               ent huildin
                                                                                                                                  navigate building element attributes)
g element attribute linked attribute of building elem
                                                                                                                                              BEGIN
                                                                      EXEC (Sodbc 2 SOL select * from
                                                                    ilding_systems where discipli
                                                                                                                                               instance number OF navigate building
                                                                  :NEW 10
                                                                        to make a new_project.design_discipline
END SQL INTO NEW_building_systems
                                                                                                                                       ent attributes := ii
                                                                                                                                               IF CONF(linked building element OF
                                                                                                                                                ment_attribute) = 100 THEN
BEGIN
                                               END
                                                                                                                                  building el
SOL
                                                                     END
            END
                                                                   WITH building_system STRING
WHEN CHANGED
           FI.SE
                                                                                                                                            ***** the following step is necessary as data
            BEGIN
                                                                     BEGIN
                                                                                                                                 can not be got directly from access to this attribute
             EXEC ISodic I SQL INSERT INTO [AB-
                                                                                                                                                 EXEC Bodbe 2 SQL SELECT
                                                                                                                                          attribute_of_building_element) from
attribute
             igner] (building_eleme
                                                                      EXEC ISodbe 2 SQL SELECT * from
attribute_of_building_element, attribute_type,
                                                                 [components_of_building_system] where
                                                                                                                                 GENERAL-AB-attribute-des
                                                                                                                                              AB-attribute-designer]
where building element = :domain.tt and
designing_discipline) values
                                                                 building_system =
                                                                  :NEW_to_make_s_new_project.building_system
                                                                                                                                  attribute_of_building_element =
                                                                                    END SQL INTO Building element
:NEW_to_make_a_new_project.building_element,
:building_element_attribute_attribute_of_building_elem
                                                                                                                                 building element attribute attribute of building elem
ent, :building_element_attribute_attribute_type,
                                                                                                                                                    END SQL INTO domain (ttt)
                                                                  WITH ignore highlighted component SIMPLE WHEN CHANGED
:building_ele
               nent_attribute.designing_discipline)
                                                                                                                                                 linked_attribute_of_building_element
                                                                                                                                 OF building element attribute := HT
                                                                     BEGIN
                                                                      FORGET CURRENT Building element
                                               END
                                                                                                                                                 EXEC Bodbe I SOL INSERT INTO
SQL
                                                                     END
                                                                                                                                 [AB-attribute-designer] (building_element,
           END
                                                                  WITH add system to the new project SIMPLE
                                                                                                                                  attribute of building element, attribute type,
                                                                                                                                 designing discipline, linked building ele
         END
                                                                    WHEN CHANGED
                                                                                                                                 linked_attribute_of_building_element) values
      END
                                                                     BEGIN
                                                                                                                                               (:domain.tt,
! **** end of method
                                                                 ! **** check that there is an inserted name and that this
                                                                                                                                 :building_element_attribute_attribute_of_building_elem
   END
                                                                 name does not exist
                                                                                                                                 ent, building element attribute attribute type,
                                                                                                                                 building element_attribute.designing_discipline_building_element_attribute.linked_building_element_buildin
 WITH finish compone
WHEN CHANGED
                      nents of the new project SIMPLE
                                                                      IF ( CONF(building_system OF
                                                                 NEW_to_make_s_new_project) >= 0) THEN
                                                                       BEGIN
                                                                                                                                 g_element_attribute.linked_attribute_of_building_elem
! **** this method sims to add the general rules of the
chosen components to the new project database
                                                                         FOR (i := 1 TO number instances OF navigate
                                                                                                                                                                                 END
     et the names of the components from the new
                                                                 Building element)
                                                                                                                                 SOL
                                                                          BEGIN
                                                                                                                                               END
project database
                                                                                                                                              FLSE
     EXEC ISodbc 1 SQL select distinct
                                                                           instance number OF navigate Building
(building element) from [AB-attribute-designer]
END SQL INTO Building element
                                                                                                                                               REGIN
                                                                                                                                                 EXEC ISodbc 1 SQL INSERT INTO
                                                                   **** get the attributes of the building element
                                                                                                                                                designer] (building element,
                                                                                                                                 (AB-attribute
     FOR (i := 1 TO number instances OF navigate
                                                                           tt := building_element OF Building element
                                                                                                                                 attribute_of_building_element, attribute_type,
Building element)
                                                                            EXEC ISodbc 2 SQL select * from
                                                                                                                                  designing_discipline) values
      BEGIN
                                                                 [GENERAL-AB-attribute-designer] where
                                                                                                                                               (:domain.tt,
                                                                                                                                 building_element_attribute_attribute_of_building_elem
       instance number OF pavigate Building element
                                                                 building_element = :domain.tt
                                                                          END SQL INTO building_element_attribute
                                                                                                                                 ent, :building_element_attribute.attribute_type,
       tt := building_element OF Building element
EXEC l5odbc 2 SQL select * from
                                                                                                                                 building element attribute designing discipline)
                                                                  **** build the tables and the attributes in the new
[GENERAL-AB-general-rules] where
                                                                 project data

1 **** create the database table with its id attribute
building_element = :domain.tt
                                                                                                                                                                                 END
                                                                 general statement of sql := CONCAT(
"CREATE TABLE [", building_element OF Building
        END SQL INTO AB_general_rules
                                                                                                                                                END
! **** add the rules to the new project database
                                                                 element, "] ([", building_element OF Building element,
                                                                                                                                             END
                                                                                                                                          end of FOR i
                                                                  ' id) TEXT(50));")
                                                                           EXEC ISodbe I SQL WITH general
       FOR (ii := 1 TO number instances OF rules
navigate 2)
                                                                 statement of sql END SQL
                                                                                                                                 I *** end of IF
        BEGIN
                                                                                                                                       END
             stance number OF rules navigate 2 :*
                                                                   **** add the rest of attributes to the created table
          EXEC ISodbe I SQL INSERT INTO [AB-
                                                                           FOR (ii := 1 TO number instances OF
                                                                                                                                 ! **** end of method
general-rules] (building_element,
attribute_of_building_element, affected_discipline,
                                                                 navigate building element attributes)
                                                                                                                                     END
                                                                                                                                   WITH finish systems of the new project SIMPLE
                                                                             BEGIN
percent_change, result_in_major_modification, cause_of_the_effect) values
                                                                              instance number OF navigate building
                                                                                                                                    WHEN CHANGED
                                                                                                                                     BEGIN
                                                                 element attributes := ii
              (:AB_general_rules.building_element,
                                                                          **** ignore the attribute that has the _id of the
                                                                                                                                    *** this method aims to add the general rules of the
:AB_general_rules.attribute_of_building_element,
                                                                 table
                                                                                                                                 components of the chosen system to the new project
```

database	WHEN NONE FOUND	BEGIN
1	i := 2	linked building element OF
! *** get the names of the components from the new	FIND END	building_element_attribute := linked_building_element
project database EXEC ISodbc 1 SQL select distinct	IFi=1 THEN	OF current_data linked_attribute_of_building_element_OF
(building element) from [AB-attribute-designer]	BEGIN	building element attribute:
END SQL INTO Building element	text OF message box 2 := "There is already a	linked_attribute_of_building_element OF current_data
FOR GOT 1 TO symbolications OF assistate	building component with that name"	END
FOR (i := 1 TO number instances OF navigate Building element)	put up OF message box 2 := TRUE FORGET new_component_name	! **** put in the database
BEGIN	END	general statement of sql := CONCAT("alter
instance number OF navigate Building element	l !	table (", building_element OF current_data, ") add [",
:= i tt := building_element OF Building_element	! If the name was not used befor IF i = 2 THEN	attribute_of_building_element OF current_data, "] ") IF attribute_type OF current_data = "t" THEN
EXEC ISodbe 2 SQL select * from	BEGIN	general statement of aql := CONCAT(general
[GENERAL-AB-general-rules] where	[**** hide the temporary put elements and put the	statement of sql, "TEXT (50);")
building_element = :domain.tt	temporary elements for ID data location OF border 6 invisible := 0,0,0,0	iF attribute_type OF current_data = "n" THEN general statement of sql := CONCAT(general
END SQL INTO AB_general_rules	location OF textbox 17 invisible := 0,0,0,0	statement of sql, "Number,")
! **** add the rules to the new project database	location OF promptbox 2 invisible := 0,0,0,0	EXEC ISodbc 1 SQL WITH general statement of
FOR Give 1 TO member insures OF miles	location OF pushbutton 10 invisible := 0,0,0.0	sql END SQL
FOR (ii := 1 TO number instances OF rules navigate 2)	location OF pushbutton 11 invisible := 0,0,0,0 location OF pushbutton 26 invisible2 :=	EXEC ISodbe I SQL INSERT INTO [AB-
BEGIN	25,265,220,285	attribute-designer] (building element,
instance number OF rules navigate 2 := ii	location OF promptbox 3 invisible2 :=	attribute_of_building_element, attribute_type,
EXEC ISodbe 1 SQL INSERT INTO [AB- general-rules] (building_element,	25,195,220,220 location OF border 7 invisible2 :=	designing discipline) values (:current_data_building_element,
attribute_of_building_element, affected_discipline,	15,150,235,290	current data attribute of building element,
percent change, result in major modification,	location OF textbox 19 invisible2:=	:current_data_attribute_type,
cause of the effect) values	25,225,220,260	current_data.designing_discipline) END SOL
(:AB_general_rules.building_element, :AB_general_rules.attribute_of_building_element,	1 **** create the database table with its id attribute	i enusqu
:AB_general_rules.affected_discipline,	general statement of sql := CONCAT(IF i = 1 THEN
:AB_general_rules.percent_change,	"CREATE TABLE [", new_component_name, "] ([",	BEGIN
:AB_general_rules.result_in_major_modification, :AB_general_rules.cause_of the_effect)	new_component_name, "_id] TEXT(50));") EXEC ISodbc 1 SQL WITH general statement	EXEC ISodbe 1 SQL UPDATE [AB-attribute- designer] set
.nb_gens a_larexcasse_or_us_enect/	of sql END SQL	linked_building_element =
	1	:current_data_linked_building_element,
END	! **** add the information to [AB-attribute-designer]	linked attribute of building element =
SQL END	table new_attribute_name := CONCAT(:current_data_linked_attribute_of_building_element WHERE building_element =
1	new_component_name, "_id")	:current_deta.building_element AND
END	EXEC Bodbe 1 SQL INSERT INTO [AB-	attribute_of_building_element =
! ! **** move to the	attribute-designer) (building_element, attribute_of_building_element) values	current data attribute_of_building_element
CURRENT_project_selection_display to allow more	(:domain.new_component_name,	END
configuration	:domain.new_attribute_name)	1
visible OF configuration window := FALSE visible OF main window := TRUE	END SQL	IF CONF(graph_name OF current_data) >= 0 THEN
output OF main window := 1KOE	MAKE Building element	BEGIN
CURENT_project_selection_display	WITH building_element :=	EXEC ISodbc 1 SQL UPDATE [AB-attribute-
! ! **** end of method	new_component_name	designer] set graph_name = :current_data.graph_name WHERE building_element =
END .	MAKE building_element_attribute	:current_data_building_element_AND
	WITH attribute of building element :=	attribute_of_building_element =
ATTRIBUTE Building Components SIMPLE	new_attribute_name	:current_data_attribute_of_building_element
WHEN CHANGED BEGIN	WITH attribute_type := "1"	END SQL END
action OF ISodbc 1 IS connect := TRUE	building_element OF current_data :=	!
t	new_component_name	
EXEC Isodbe 1 SQL SELECT distinct	attribute_of_building_element OF current_data := new_attribute_name	FORGET new_attribute_name FORGET attribute type OF input data
(building_element), graph_name from [AB-attribute- designer]	attribute_type OF current_data := "t"	FORGET discipline OF input data
END SQL INTO Building element	!	location OF pushbutton 12 invisible2 := 0,0,0,0
POTO IC. E - LOOK OFFERT F. days	END	location OF pushbutton 13 invisible2 := 0,0,0,0
EXEC ISodbc 1 SQL SELECT distinct (building element) from [AB-attribute-designer]	END 1	location OF border 7 invisible2 := 0,0,0,0 location OF textbox 18 invisible2 := 0,0,0,0
END SQL INTO Linked building element	END	location OF textbox 19 invisible2 := 0,0,0,0
	ATTRIBUTE new_component_name STRING	location OF pushbutton 16 invisible2 := 0,0,0,0
visible OF main window := FALSE output OF configuration window := config	ATTRIBUTE new attribute_name STRING ATTRIBUTE done of new attribute SIMPLE	location OF promptbox 3 invisible2 := 0,0,0,0
visible OF configuration window := TRUE	WHEN CHANGED	END
1	BEGIN	ELSE -
END ATTRIBUTE IL STRING		BEGIN text OF message box 2 := "Some essential data
ATTRIBUTE I NUMERIC	1:-0	are missing. Check attribute name, type, and designing
ATTRIBUTE general statement of sql STRING	IF (CONF(new_attribute_name) >= 0 AND	discipline.*
ATTRIBUTE done of new component SIMPLE	CONF(attribute_type OF current_data) >= 0 AND CONF(designing discipline OF current_data) >= 0)	put up OF message box 2 := TRUE END
WHEN CHANGED BEGIN	THEN	END
i := 0	BEGIN	ATTRIBUTE next new attribute SIMPLE
	I and the office described	WHEN CHANGED
! **** check that there is an inserted name and that this name does not exist	attribute_of_building_element OF current_data := new_attribute_name	BEGIN IF (CONF(new_attribute_name) >= 0 AND
IF (CONF(new_component_name) >= 0) THEN	attribute_type OF building_element_attribute :=	CONF(attribute_type OF current_data) >= 0 AND
BEGIN	attribute_type OF current_data	CONF(designing_discipline OF current_data) >= 0)
FIND Building element WHERE building element OF Building	designing_discipline OF	THEN
WULLE OFFICIAL CLUMENT OF DOMOTHE	building element applying to designing discipling All	I REGIN
element = new_component_name	building element attribute := designing discipline OF current data	BEGIN !

designing_discipline OF	AB_general_rules) >= 0 AND CONF(designer] where building_element =
building_element_attribute := designing_discipline OF	attribute_of_building_element OF AB_general_rules)	current_data.building_element_END SQL
CUITERE_data	>= 0 AND CONF(affected_discipline OF	EXEC ISodbc SQL SELECT distinct
IF CONF(linked_building_element OF	AB_general_rules) >= 0) THEN	(building_element), graph_name from [AB-attribute-
current_data) >= 0 THEN	BEGIN	designer]
BEGIN	IF CONF(AB_general_rules_ID OF	END SQL INTO Building element
linked_building_element OF	AB_general_rules) >= 0 THEN	EXEC ISodbc 1 SQL SELECT distinct
building element attribute := linked_building_element	BEGIN	(building_element) from [AB-attribute-designer]
OF current data	EXEC (Sodbc 1 SQL update [AB-general-	END SQL INTO Linked building element
linked attribute of building element OF	rules] set	FORGET current_data
building element attribute :-	affected_discipline =	FORGET building_element_attribute
linked_attribute_of_building_element OF current_data	:AB_general_rules.affected_discipline,	END
END	percent_change = :AB_general_rules.percent_change,	ELSE
1	result in major modification =	BEGIN
FORGET new_attribute_name	:AB general_rules.result_in_major_modification,	text OF message box 2 := "No Building
FORGET attribute of building element OF	rule ignored = :AB general rules rule ignored.	Component is currently chosen*
current data	cause of the effect	put up OF message box 2 := TRUE
FORGET attribute_type OF current_data	:AB general rules cause of the effect	END
FORGET designing discipline OF current data	where AB general rules ID =	1
FORGET linked building element OF	:AB_general_rules.AB_general_rules_ID	! **** end of method
CUTTENT data	END SOL	END
FORGET linked attribute of building element	END	ATTRIBUTE done of ID attribute SIMPLE
	ELSE	WHEN CHANGED
OF current_data	BEGIN	BEGIN
FORGET attribute type OF input data		I BEGIN
FORGET discipline OF input data	IF CONF(percent_change OF	? •
location OF pushbutton 16 invisible2 :=	AB_general_rules) < 0 THEN	i:=0
180,195,220,220	percent change OF AB_general rules := 0	IE (CONE/man and the same and same
location OF promptbox 3 invisible2 :=	IF CONF(result_in_major_modification OF	IF (CONF(new_attribute_name) >= 0 AND
25,195,220,220	AB_general_rules) < 0 THEN	CONF(attribute_type OF current_data) >= 0 AND
location OF textbox 18 invisible2 :=	result_in_major_modification OF	CONF(designing_discipline OF current_data) >= 0)
25,155,220,190	AB_general_rules := "No"	THEN
1	IF CONF(rule_ignored OF AB_general_rules)	BEGIN
END	< O THEN	!
ELSE	rule_ignored OF AB_general_rules :** "No"	attribute_type OF building_element_attribute :
BEGIN	IF CONF(cause of the effect OF	attribute_type OF current_data
text OF message box 2 := "Some essential data	AB_general_rules) < 0 THEN	designing_discipline OF
are missing. Check attribute name, type, and designing	cause of the effect OF AB general rules :=	building element attribute := designing discipline O
discipline."		current data
put up OF message box 2 := TRUE	EXEC ISodbe I SQL insert into [AB-general-	IF CONF(linked_building_element OF
END	nules	current data) >= 0 THEN
END	(building element, attribute of building element,	BEGIN
ATTRIBUTE ole of attribute name SIMPLE	affected discipline, percent change,	linked_building_element OF
	result_in_major_modification,	building element attribute := linked_building_eleme
WHEN CHANGED BEGIN	rule_ignored_cause_of_the_effect)	OF current_data
	values	linked_attribute_of_building_element OF
IF (CONF(new_attribute_name) >= 0) THEN	(:AB_general_rules.building_element,	building element attribute :=
BEGIN	I CAD RESERVICIONES CAUSANA CONTRACTOR DE LA CONTRACTOR D	Dunding element attribute
topop to the topop to the state of the state		tinhad and hailding alaman OS assent day
1 **** check if there is an attribute with the same name	:AB_general_rules.attribute_of_building_element,	linked attribute of building element OF current date
FIND building_element_attribute	:AB_general_rules.attribute_of_building_element, :AB_general_rules.affocted_discipline,	i:=t
FIND building_element_attribute WHERE attribute_of_building_element OF	:AB_general_rules.attribute_of_building_element, :AB_general_rules.affected_discipline, :AB_general_rules.percent_change,	i:=1 END
FIND building_element_attribute WHERE attribute_of_building_element OF building_element_attribute = new_attribute_name	:AB_general_rules.attribute_of_building_element, :AB_general_rules.affected_discipline, :AB_general_rules.percent_change, :AB_general_rules.result_in_major_modification,	i:=t END
FIND building_element_attribute WHERE attribute_of building_element Of building_element_attribute = new_attribute_name WHEN FOUND	:AB_general_rules.attribute_of_building_element, :AB_general_rules.affected_discipline, :AB_general_rules.percent_change, :AB_general_rules.result_in_major_modification, :AB_general_rules.rule_ignored,	i:= END ! ! **** put in the database
FIND building_element_attribute WHERE attribute_of_building_element OF building_element_attribute = new_attribute_name WHEN FOUND i := 1	:AB_general_rules.attribute_of_building_element, :AB_general_rules.affected_discipline, :AB_general_rules.percent_change, :AB_general_rules.result_in_major_modification, :AB_general_rules.rule_ignored, :AB_general_rules.cause_of_the_effect)	i:= END **** put in the database EXEC Sodbc SQL UPDATE [AB-attribute-
FIND building_element_attribute WHERE attribute_of_building_element OF building_element attribute = new_attribute_name WHEN FOUND i := 1 WHEN NONE FOUND	:AB_general_rules.attribute_of_building_element, :AB_general_rules.affected_discipline, :AB_general_rules.percent_change, :AB_general_rules.result_in_major_modification, :AB_general_rules.rule_ignored, :AB_general_rules.cause_of_the_effect) END SQL	i := 1 END ! !*** put in the database EXEC 15odbc 1 SQL UPDATE (AB-attribute-designer) set
FIND building_element_attribute WHERE attribute_0 building_element OF building_element OF building_element OF building_element of the service	:AB_general_rules.attribute_of_building_element, :AB_general_rules.affected_discipline, :AB_general_rules.percent_change, :AB_general_rules.result_in_major_modification, :AB_general_rules.rule_ignored, :AB_general_rules.cause_of_the_effect)	i := 1 END ! ! **** put in the database EXEC ISOdbc 1 SQL UPDATE [AB-attribute-designer] set attribute_type = :current_data_attribute_type,
FIND building_element_attribute WHERE attribute_of_building_element OF building_element attribute = new_attribute_name WHEN FOUND i := 1 WHEN NONE FOUND	:AB_general_rules_attribute_of_building_element, :AB_general_rules_affected_discipline, :AB_general_rules_percent_change, :AB_general_rules_result_in_major_modification, :AB_general_rules_rule_ignored, :AB_general_rules_cause_of_the_effect) END SQL END	i:= END ! ! **** put in the database EXEC Sodbc SQL UPDATE (AB-attribute- designer) set attribute_type = :current_data_attribute_type, designing_discipline =
FIND building_element_attribute WHERE attribute_0 building_element OF building_element OF building_element OF building_element of the service	:AB_general_rules_attribute_of_building_element, :AB_general_rules_affected_discipline, :AB_general_rules_precent_change, :AB_general_rules_result_in_major_modification, :AB_general_rules_rule_ignored, :AB_general_rules_cause_of_the_effect) END SQL END [END] [END]	i := 1 END ! ! *** put in the database EXEC !Sodbe ! SQL UPDATE [AB-attribute-designer] set attribute_type = :current_data_attribute_type, designing_discipline = :current_data_designing_discipline
FIND building_element_attribute WHERE attribute_0 building_element OF building_element attribute = new_attribute_name WHEN FOUND i := 1 WHEN NONE FOUND i := 2 FIND END	:AB_general_rules_attribute_of_building_element, :AB_general_rules_affected_discipline, :AB_general_rules_percent_change, :AB_general_rules_result_in_major_modification, :AB_general_rules_rule_ignored, :AB_general_rules_cause_of_the_effect) END SQL END	i:= END ! !**** put in the database EXEC ISOdbc SQL UPDATE (AB-attribute- designer) set attribute_type = :current_data_attribute_type, designing_discipline = :current_data_designing_discipline WHERE building_element =
FIND building_element_attribute WHERE attribute_of_building_element OF building_element_attribute = new_attribute_name WHEN FOUND i = 1 WHEN NONE FOUND i = 2 FIND END	:AB_general_rules_attribute_of_building_element, :AB_general_rules_affected_discipline, :AB_general_rules_precent_change, :AB_general_rules_result_in_major_modification, :AB_general_rules_rule_ignored, :AB_general_rules_cause_of_the_effect) END SQL END [END] [END]	i:= END ! ! **** put in the database EXEC ISodbc 1 SQL UPDATE (AB-attribute-designer) set attribute_type = :current_data_attribute_type, designing_discipline = :current_data_designing_discipline WHERE building_element = :current_data_building_element AND
FIND building_element_attribute WHERE attribute_0 building_element OF building_element	:AB_general_rules_attribute_of_building_element, :AB_general_rules_affected_discipline, :AB_general_rules_percent_change, :AB_general_rules_result_in_major_modification, :AB_general_rules_rule_ignored, :AB_general_rules_cause_of_the_effect) END SQL END END END END ELSE	i:= END ! ! **** put in the database EXEC Sodbc SQL UPDATE AB-attribute- designer set attribute type = :current_data_attribute_type, designing_discipline = :current_data_designing_discipline WHERE building_element = :current_data_building_element AND attribute_of_building_element =
FIND building_element_attribute WHERE attribute_of building_element OF building_element attribute = new_attribute_name WHEN FOUND i = 1 WHEN NONE FOUND i = 2 FIND END	:AB_general_rules_attribute_of_building_element, :AB_general_rules_affected_discipline, :AB_general_rules_percent_change, :AB_general_rules_result_in_major_modification, :AB_general_rules_rule_ignored, :AB_general_rules_cause_of_the_effect) END SQL END END END END ESE BEGIN	i:= END ! ! **** put in the database EXEC ISodbc 1 SQL UPDATE (AB-attribute-designer) set attribute_type = :current_data_attribute_type, designing_discipline = :current_data_designing_discipline WHERE building_element = :current_data_building_element AND
FIND building_element_attribute WHERE attribute_of_building_element Of building_element attribute = new_attribute_name WHEN FOUND i := 1 WHEN NONE FOUND i := 2 FIND END ! IF i = 2 THEN BEGIN MAKE building_element_attribute	:AB_general_rules_attribute_of_building_element, :AB_general_rules_affected_discipline, :AB_general_rules_precent_change, :AB_general_rules_result_in_major_modification, :AB_general_rules_rule_ignored, :AB_general_rules_cause_of_the_effect) END SQL END END END END END END ELSE BEGIN text OF message box 2 := "Some important data	i:= END ! ! **** put in the database EXEC Sodbc SQL UPDATE AB-attribute- designer set attribute type = :current_data_attribute_type, designing_discipline = :current_data_designing_discipline WHERE building_element = :current_data_building_element AND attribute_of_building_element =
FIND building_element_attribute WHERE attribute_of building_element OF building_element attribute = new_attribute_name WHEN FOUND i = 1 WHEN NONE FOUND i = 2 FIND END IF i = 2 THEN BEGIN MAKE building_element_attribute WITH attribute_of_building_element := new_attribute_name	:AB_general_rules_attribute_of_building_element, :AB_general_rules_affected_discipline, :AB_general_rules_percent_change, :AB_general_rules_result_in_major_modification, :AB_general_rules_rule_ignored, :AB_general_rules_cause_of_the_effect) END SQL END [END ELSE BEGIN Lext OF message box 2 := "Some important data are missing.	i:= END ! !**** put in the database EXEC ISOdbc 1 SQL UPDATE (AB-attribute- designer) set attribute_type = :current_data_attribute_type, designing_discipline = :current_data_designing_discipline WHERE building_element = :current_data_building_element AND attribute_of_building_element = :current_data_attribute_of_building_element
FIND building_element_attribute WHERE attribute_of building_element OF building_element Fibute = new_attribute_name WHEN FOUND i = 1 WHEN NONE FOUND i = 2 FIND END IF i = 2 THEN BEGIN MAKE building_element_attribute WITH attribute_of_building_element = new_attribute_name location OF pushbutton 12 invisible2 := 25.265.225.285	:AB_general_rules_attribute_of_building_element, :AB_general_rules_affected_discipline, :AB_general_rules_percent_change, :AB_general_rules_result_in_major_modification, :AB_general_rules_rule_ignored, :AB_general_rules_cause_of_the_effect) END SQL END [END ELSE BEGIN text OF message box 2 := "Some important data are missing. You may need to puch New Rule botton"	i:= END ! !**** put in the database
FIND building_element_attribute WHERE attribute_of building_element OF building_element Fibute = new_attribute_name WHEN FOUND i = 1 WHEN NONE FOUND i = 2 FIND END IF i = 2 THEN BEGIN MAKE building_element_attribute WITH attribute_of_building_element = new_attribute_name location OF pushbutton 12 invisible2 := 25.265.225.285	:AB_general_rules_attribute_of_building_element, :AB_general_rules_affected_discipline, :AB_general_rules_precent_change, :AB_general_rules_result_in_major_modification, :AB_general_rules_rule_ignored, :AB_general_rules_cause_of_the_effect) END SQL END END END END END END END END	i:= END ! ! **** put in the database
FIND building_element_attribute WHERE attribute_of building_element OF building_element attribute = new_attribute_name WHEN FOUND i = 1 WHEN NONE FOUND i = 2 FIND END IF i = 2 THEN BEGIN MAKE building_element_attribute WITH attribute_of_building_element = new_attribute_name location OF pushbutton 12 invisible2 := 25.265,225,285 location OF pushbutton 16 invisible2 :=	:AB_general_rules_attribute_of_building_element, :AB_general_rules_affected_discipline, :AB_general_rules_percent_change, :AB_general_rules_result_in_major_modification, :AB_general_rules_rule_ignored, :AB_general_rules_cause_of_the_effect) END SQL END ! END ELSE BEGIN text OF message box 2 := "Some important data are missing. You may need to puch New Rule botton" put up OF message box 2 := TRUE END END END END	i:= END
FIND building_element_attribute WHERE attribute_of building_element OF building_element Fibute = new_attribute_name WHEN FOUND i = 1 WHEN NONE FOUND i = 2 FIND END IF i = 2 THEN BEGIN MAKE building_element_attribute WITH attribute_of_building_element = new_attribute_name location OF pushbutton 12 invisible2 := 25.265.225.285	:AB_general_rules_attribute_of_building_element, :AB_general_rules_affected_discipline, :AB_general_rules_percent_change, :AB_general_rules_result_in_major_modification, :AB_general_rules_rule_ignored, :AB_general_rules_cause_of_the_effect) END SQL END ! END ELSE BEGIN text OF message box 2 := "Some important data are missing. You may need to puch New Rule botton" put up OF message box 2 := TRUE END	i:= END ! !**** put in the database
FIND building_element_attribute WHERE attribute_of building_element OF building_element attribute = new_attribute_name WHEN FOUND i = 1 WHEN NONE FOUND i = 2 FIND END IF i = 2 THEN BEGIN MAKE building_element_attribute WITH attribute_of_building_element := new_attribute_name location OF pushbutton 12 invisible2 := 25,265,225,285 location OF pushbutton 16 invisible2 := 0,0,0,0 END	:AB_general_rules_attribute_of_building_element, :AB_general_rules_affected_discipline, :AB_general_rules_precent_change, :AB_general_rules_result_in_major_modification, :AB_general_rules_rule_ignored, :AB_general_rules_cause_of_the_effect) END SQL END END END END END END END END	i:= END !**** put in the database EXEC ISodbc 1 SQL UPDATE (AB-attribute- designer) set sattribute_type = :current_data_attribute_type, designing_discipline = :current_data_designing_discipline WHERE building_element = :current_data_building_element = :current_data_building_element = :current_data_sattribute_of_building_element END SQL ! IF i = 1 THEN BEGIN EXEC ISodbc 1 SQL UPDATE (AB-attribute_IND ISOL ISOL ISOL ISOL ISOL ISOL ISOL ISOL
FIND building_element_attribute WHERE attribute_of building_element OF building_element_attribute = new_attribute_name WHEN FOUND i = 1 WHEN NONE FOUND i = 2 FIND END IF i = 2 THEN BEGIN MAKE building_element_attribute WITH attribute_of_building_element := new_attribute_name location OF pushbutton 12 invisible2 := 25,265,225,285 location OF pushbutton 16 invisible2 := 0,0,0,0	:AB_general_rules_attribute_of_building_element, :AB_general_rules_affected_discipline, :AB_general_rules_percent_change, :AB_general_rules_result_in_major_modification, :AB_general_rules_rule_ignored, :AB_general_rules_cause_of_the_effect) END SQL END ! END ELSE BEGIN text OF message box 2 := "Some important data are missing, You may need to puch New Rule botton" put up OF message box 2 := TRUE END END ATTRIBUTE new rule SIMPLE WHEN CHANGED BEGIN	i:= END
FIND building_element_attribute WHERE attribute_of building_element OF building_element_attribute = new_attribute_name WHEN FOUND i = 1 WHEN NONE FOUND i = 2 FIND END IF i = 2 THEN BEGIN MAKE building_element_attribute WITH attribute_of_building_element := new_attribute_name location OF pushbutton 12 invisible2 := 25,265,225,285 location OF pushbutton 16 invisible2 := 0,0,0,0 END END	:AB general rules attribute of building element, :AB general rules affected discipline, :AB general rules percent change, :AB general rules result in major modification, :AB general rules result in major modification, :AB general rules cause of the effect) END SQL END END END END END END END EN	i:= END **** put in the database
FIND building_element_attribute WHERE attribute_of_building_element OF building_element attribute = new_attribute_name WHEN FOUND i = 1 WHEN NONE FOUND i = 2 FIND END IF i = 2 THEN BEGIN MAKE building_element_attribute WITH stribute_of_building_element := new_attribute_name location OF pushbutton 12 invisible2 := 25,265,225,235 location OF pushbutton 16 invisible2 := 0,0,0,0 END END END !	:AB_general_rules_attribute_of_building_element, :AB_general_rules_atflected_discipline, :AB_general_rules_precent_change, :AB_general_rules_result_in_major_modification, :AB_general_rules_result_in_major_modification, :AB_general_rules_rule_ignored, :AB_general_rules_cause_of_the_effect) END SQL END END END ELSE BEGIN text OF message box 2 := "Some important data are missing. You may need to puch New Rule botton" put up OF message box 2 := TRUE END END ATTRIBUTE new rule SIMPLE WHEN CHANGED BEGIN IF (CONF(building_element OF Current_rule_data) >= 0 AND CONF(i:= END ! !**** put in the database
FIND building_element_attribute WHERE attribute_of building_element OF building_element attribute = new_attribute_name WHEN FOUND i = 1 WHEN NONE FOUND i = 2 FIND END IF i = 2 THEN BEGIN MAKE building_element_attribute WITH attribute_of_building_element = new_attribute_name location OF pushbutton 12 invisible2 := 0,0,0,0 END END ! *** end of method END	:AB_general_rules_attribute_of_building_element, :AB_general_rules_affected_discipline, :AB_general_rules_result_in_major_modification, :AB_general_rules_result_in_major_modification, :AB_general_rules_rule_ignored, :AB_general_rules_cause_of_the_effect) END SQL END END END ELSE BEGIN text OF message box 2 := "Some important data are missing, You may need to puch New Rule botton" put up OF message box 2 := TRUE END END ATTRIBUTE new rule SIMPLE WHEN CHANGED BEGIN IF (CONF(building_element OF Current_rule_data) >= O ANID CONF(attribute_of_building_element OF Current_rule_data)	i:= END ! !**** put in the database
FIND building_element_attribute WHERE attribute_of building_element OF building_element attribute = new_attribute_name WHEN FOUND i = 1 WHEN NONE FOUND i = 2 FIND END IF i = 2 THEN BEGIN MAKE building_element_attribute WITH attribute_of_building_element := new_attribute_name location OF pushbutton 12 invisible2 := 25,265,225,285 location OF pushbutton 16 invisible2 := 0,0,0,0 END END ! ***Proceedings** ***Proceedings** *	:AB general rules attribute of building element, :AB general rules affected discipline, :AB general rules percent change, :AB general rules result in major modification, :AB general rules result in major modification, :AB general rules cause of the effect) END SQL END END END END END END END EN	i:= END **** put in the database
FIND building_element_attribute WHERE attribute_of building_element OF building_element attribute = new_attribute_name WHEN FOUND i = 1 WHEN NONE FOUND i = 2 FIND END IF i = 2 THEN BEGIN MAKE building_element_attribute WITH attribute_of_building_element := new_attribute_name location OF pushbutton 12 invisible2 := 25,265,225,285 location OF pushbutton 16 invisible2 := 0,0,0,0 END END END END ATTRIBUTE General Rules SIMPLE WHEN CHANGED	:AB_general_rules_attribute_of_building_element, :AB_general_rules_affected_discipline, :AB_general_rules_precent_change, :AB_general_rules_result_in_major_modification, :AB_general_rules_rule_ignored, :AB_general_rules_cause_of_the_effect) END SQL END END ELSE BEGIN text OF message box 2 := "Some important data are missing. You may need to puch New Rule botton" put up OF message box 2 := TRUE END END ATTRIBUTE new rule SIMPLE WHEN CHANGED BEGIN IF (CONF(building_element OF Current_rule_data) >= 0 AND CONF(attribute_of_building_element OF Current_rule_data) >= 0) THEN BEGIN	i:= END
FIND building_element_attribute WHERE attribute_of building_element OF building_element attribute = new_attribute_name WHEN FOUND i = 1 WHEN NONE FOUND i = 2 FIND END IF i = 2 THEN BEGIN MAKE building_element_attribute WITH attribute_of_building_element := new_attribute_name location OF pushbutton 12 invisible2 := 25,265,225,285 location OF pushbutton 16 invisible2 := 0,0,0,0 END END END ! *** end of method END ATTRIBUTE General Rules SIMPLE WHEN CHANGED BEGIN	:AB_general_rules_attribute_of_building_element, :AB_general_rules_affected_discipline, :AB_general_rules_percent_change, :AB_general_rules_result_in_major_modification, :AB_general_rules_rule_ignored, :AB_general_rules_cause_of_the_effect) END SQL END END ELSE BEGIN text OF message box 2 := "Some important data are missing. You may need to puch New Rule botton" put up OF message box 2 := TRUE END END ATTRIBUTE new rule SIMPLE WHEN CHANGED BEGIN IF (CONF(building_element OF Current_rule_data) >= 0 ANID CONF(attribute_of_building_element OF Current_rule_data) >= 0) THEN BEGIN FORGET_discipline OF input data	i:= END ! !**** put in the database
FIND building_element_attribute WHENE attribute_of building_element OF building_element attribute = new_attribute_name WHEN FOUND i = 1 WHEN NONE FOUND i = 2 FIND END IF i = 2 THEN BEGIN MAKE building_element_attribute WITH attribute_of_building_element := new_attribute_name location OF pushbutton 12 invisible2 := 25.265.225.285 location OF pushbutton 16 invisible2 := 0,0,0,0 END END	:AB general rules attribute of building element, :AB general rules affected discipline, :AB general rules percent change, :AB general rules percent change, :AB general rules result in major modification, :AB general rules rule ignored, :AB general rules cause of the effect) END SQL END END END END END END END EN	i := END **** put in the database
FIND building_element_attribute WHERE attribute = new_attribute_name WHEN FOUND i = 1 WHEN NONE FOUND i = 2 FIND END IF i = 2 THEN BEGIN MAKE building_element_attribute WITH attribute_of_building_element := new_attribute_name location OF pushbutton 12 invisible2 := 25,265,225,285 location OF pushbutton 16 invisible2 := 0,0,0,0 END END END END ATTRIBUTE General Rules SIMPLE WHEN CHANGED BEGIN action OF ISodbe 1 IS connect := TRUE	:AB general rules attribute of building element, :AB general rules affected discipline, :AB general rules percent change, :AB general rules result in major modification, :AB general rules result in major modification, :AB general rules cause of the effect) END SQL END END END END END END END END	i := END **** put in the database EXEC ISodbe I SQL UPDATE [AB-attribute-designer] set attribute_type = :current_data_attribute_type, designing_discipline = :current_data_dats_designing_discipline WHERE building_element = :current_data_building_element AND attribute_of_building_element = :current_data_attribute_of_building_element END SQL IF i = I THEN BEGIN EXEC ISodbe I SQL UPDATE [AB-attribute_designer] set linked_building_element = :current_data_linked_building_element, linked_attribute_of_building_element :current_data_linked_building_element = :current_data_building_element = :current_data_building_element = :current_data_building_element = :current_data_building_element = :current_data_building_element = :current_data_building_element = :current_data_attribute_of_building_element END SQL
FIND building_element_attribute WHERE attribute_of building_element OF building_element attribute = new_attribute_name WHEN FOUND i = 1 WHEN NONE FOUND i = 2 FIND END IF i = 2 THEN BEGIN MAKE building_element_attribute WITH stribute_of_building_element := new_attribute_name location OF pushbutton 12 invisible2 := 25,265,225,285 location OF pushbutton 16 invisible2 := 0,0,0,0 END END END **** end of method END ATTRIBUTE General Rules SIMPLE WHEN CHANGED BEGIN action OF ISodbe 1 IS connect := TRUE	:AB general rules attribute of building element, :AB general rules affected discipline, :AB general rules percent change, :AB general rules result in major modification, :AB general rules result in major modification, :AB general rules cause of the effect) END SQL END END ELSE BEGIN text OF message box 2 := "Some important data are missing. You may need to puch New Rule botton" put up OF message box 2 := TRUE END END ATTRIBUTE new rule SIMPLE WHEN CHANGED BEGIN IF (CONF(building element OF Current rule data) >= 0 AND CONF(attribute of building element OF Current rule data) >= 0) THEN BEGIN FORGET discipline OF input data MAKE AB general rules WITH building element := building element OF Current rule data	i := END **** put in the database
FIND building_element_attribute WHENE attribute_of building_element OF building_element attribute = new_attribute_name WHEN FOUND i = 1 WHEN NONE FOUND i = 2 FIND END IF i = 2 THEN BEGIN MAKE building_element_attribute WITH attribute_of_building_element := new_attribute_name location OF pushbutton 12 invisible2 := 25.265.225.285 location OF pushbutton 16 invisible2 := 0.0.0.0 END END *** end of method END ATTRIBUTE General Rules SIMPLE WHEN CHANGED BEGIN action OF ISodbe 1 IS connect := TRUE EXEC ISodbe 1 SQL SELECT distinct	:AB general rules attribute of building element, :AB general rules affected discipline, :AB general rules percent change, :AB general rules result in major modification, :AB general rules result in major modification, :AB general rules cause of the effect) END SQL END END END END END END END END	i := END **** put in the database
FIND building_element_attribute WHERE attribute_of building_element OF building_element Stribute = new_attribute_name WHEN FOUND i := 1 WHEN NONE FOUND i := 2 FIND END IF i = 2 THEN BEGIN MAKE building_element attribute WITH attribute_of_building_element := new_attribute_name location OF pushbutton 12 invisible2 := 25,265,225,285 location OF pushbutton 16 invisible2 := 0,0,0,0 END END END **** end of method END ATTRIBUTE General Rules SIMPLE WHEN CHANGED BEGIN **** action OF ISodbe 1 IS connect := TRUE EXEC ISodbe 1 SQL SELECT distinct (building_element) from [AB-attribute-designer]	:AB general rules attribute of building element, :AB general rules affected discipline, :AB general rules percent change, :AB general rules percent change, :AB general rules result in major modification, :AB general rules cause of the effect) END SQL END END END END END END END END	i := 1 END ! !**** put in the database EXEC ISodbe I SQL UPDATE [AB-attribute-designer] set attribute_type = :current_data_attribute_type, designing_discipline = :current_data_dats_designing_discipline WHERE building_element = :current_data_building_element AND attribute_of_building_element = :current_data_attribute_of_building_element END SQL IF i = 1 THEN BEGIN EXEC ISodbe I SQL UPDATE [AB-attribute] designer] set linked_building_element = :current_data_linked_building_element, linked_attribute_of_building_element WHERE building_element = :current_data_building_element = :current_data_building_element = :current_data_building_element = :current_data_building_element = :current_data_building_element = :current_data_building_element = :current_data_attribute_of_building_element END SQL END IF CONF(graph_name_OF current_data) >= 0
FIND building_element_attribute WHERE attribute_of building_element OF building_element attribute = new_attribute_name WHEN FOUND i = 1 WHEN NONE FOUND i = 2 FIND END IF i = 2 THEN BEGIN MAKE building_element_attribute WITH attribute_of_building_element := new_attribute_name location OF pushbutton 12 invisible2 := 25,265,225,285 location OF pushbutton 16 invisible2 := 0,0,0,0 END END END ! **** end of method END ATTRIBUTE General Rules SIMPLE WHEN CHANGED BEGIN action OF ISodbe 1 IS connect := TRUE ! EXEC ISodbe 1 SQL SELECT distinct (building_element) from [AB-attribute-designer] END SQL INTO Building element	:AB general rules attribute of building element, :AB general rules affected discipline, :AB general rules percent change, :AB general rules result in major modification, :AB general rules result in major modification, :AB general rules cause of the effect) END SQL END END ELSE BEGIN text OF message box 2 := "Some important data are missing. You may need to puch New Rule botton" put up OF message box 2 := TRUE END END ATTRIBUTE new rule SIMPLE WHEN CHANGED BEGIN IF (CONF(building element OF Current rule data) >= 0 AND CONF(attribute of building element OF Current rule data) >= 0) THEN BEGIN FORGET discipline OF input data MAKE AB general rules WITH building element := building element OF Current rule data WITH stribute of building element := stribute of building element OF Current rule data END	i := END
FIND building_element_attribute WHERE attribute_of building_element OF building_element attribute = new_attribute_name WHEN FOUND i = 1 WHEN NONE FOUND i = 2 FIND END IF i = 2 THEN BEGIN MAKE building_element_attribute WITH attribute_of_building_element := new_attribute_name location OF pushbutton 12 invisible2 := 0.0.0.0 END END END ! *** end of method END ATTRIBUTE General Rules SIMPLE WHEN CHANGED BEGIN action OF ISodbe 1 IS connect := TRUE ! EXEC ISodbe 1 SQL SELECT distinct (building_element) from [AB-attribute-designer] END SQL INTO Building element !	:AB general rules attribute of building element, :AB general rules affected discipline, :AB general rules percent change, :AB general rules result in major modification, :AB general rules rule ignored, :AB general rules cause of the effect) END SQL END END END END END END END END	i := END **** put in the database
FIND building_element_attribute WHERE attribute_of building_element OF building_element Stribute = new_attribute_name WHEN FOUND i = 1 WHEN NONE FOUND i = 2 FIND END IF i = 2 THEN BEGIN MAKE building_element_attribute WITH attribute_of_building_element := new_attribute_name location OF pushbutton 12 invisible2 := 25,265,225,285 location OF pushbutton 16 invisible2 := 0,0,0,0 END END END END	:AB general rules attribute of building element, :AB general rules affected discipline, :AB general rules percent change, :AB general rules result in major modification, :AB general rules result in major modification, :AB general rules cause of the effect) END SQL END END END END END END END END	i := END
FIND building_element_attribute WHERE attribute_of building_element OF building_element attribute = new_attribute_name WHEN FOUND i = 1 WHEN NONE FOUND i = 2 FIND END IF i = 2 THEN BEGIN MAKE building_element_attribute WITH attribute_of_building_element := new_attribute_name location OF pushbutton 12 invisible2 := 25,265,225,285 location OF pushbutton 16 invisible2 := 0,0,0,0 END END END ! ! **** end of method END ATTRIBUTE General Rules SIMPLE WHEN CHANGED BEGIN ! action OF ISodbe 1 IS connect := TRUE ! EXEC ISodbe 1 SQL SELECT distinct (building_element) from [AB-attribute-designer] END SQL INTO Building element ! visible OF main window := FALSE output OF configuration window := General rules	:AB general rules attribute of building element, :AB general rules affected discipline, :AB general rules percent change, :AB general rules result in major modification, :AB general rules result in major modification, :AB general rules cause of the effect) END SQL END END END ELSE BEGIN text OF message box 2 := "Some important data are missing. You may need to puch New Rule botton" put up OF message box 2 := TRUE END END ATTRIBUTE new rule SIMPLE WHEN CHANGED BEGIN IF (CONF(building element OF Current rule data) >= 0 AND CONF(attribute of building element OF Current rule data) >= 0 AND CONF(attribute of building element := building element OF Current rule data WITH stribute of building element := building element OF Current rule data WITH stribute of building element := stribute of building element oF Current rule data WITH stribute of building element := building element END ATTRIBUTE delete a component SIMPLE WHEN CHANGED	i := END
FIND building_element_attribute WHERE attribute_of building_element OF building_element attribute = new_attribute_name WHEN FOUND i = 1 WHEN NONE FOUND i = 2 FIND END IF i = 2 THEN BEGIN MAKE building_element_attribute WITH attribute_of_building_element := new_attribute_name location OF pushbutton 12 invisible2 := 0,0,0,0 END END	:AB general rules attribute of building element, :AB general rules affected discipline, :AB general rules percent change, :AB general rules result in major modification, :AB general rules rule ignored, :AB general rules cause of the effect) END SQL END END END END END END END END	i := END **** put in the database
FIND building_element_attribute WHERE attribute_of building_element OF building_element attribute = new_attribute_name WHEN FOUND i = 1 WHEN NONE FOUND i = 2 FIND END IF i = 2 THEN BEGIN MAKE building_element_attribute WITH attribute_of_building_element := new_attribute_name location OF pushbutton 12 invisible2 := 25,265,225,285 location OF pushbutton 16 invisible2 := 0,0,0,0 END END END ! ! **** end of method END ATTRIBUTE General Rules SIMPLE WHEN CHANGED BEGIN ! action OF ISodbe 1 IS connect := TRUE ! EXEC ISodbe 1 SQL SELECT distinct (building_element) from [AB-attribute-designer] END SQL INTO Building element ! visible OF main window := FALSE output OF configuration window := General rules	:AB general rules attribute of building element. :AB general rules affected discipline. :AB general rules percent change. :AB general rules result in major modification. :AB general rules result in major modification. :AB general rules cause of the effect) END SQL END END END END END END END END	i := 1 END ! !**** put in the database EXEC ISodbe I SQL UPDATE [AB-attribute-designer] set attribute_type = :current_data_attribute_type, designing_discipline = :current_data_tatribute_type, designing_discipline = :current_data_tatribute_type, designing_discipline = :current_data_tatribute_type, designing_discipline = :current_data_tatribute_of_building_element = :current_data_attribute_of_building_element END SQL IF i = 1 THEN BEGIN EXEC ISodbe I SQL UPDATE [AB-attribute_designer] set linked_building_element = :current_data_linked_building_element, linked_attribute_of_building_element WHERE building_element = :current_data_building_element = :current_data_building_element = :current_data_building_element = :current_data_tatribute_of_building_element END SQL END IF CONF(graph_name_OF current_data) >= 0 THEN BEGIN EXEC ISodbe I SQL UPDATE [AB-attribute_designer] set graph_name = :current_data_graph_name WHERE building_element = :current_data_graph_name unrent_data_building_element = :current_data_graph_name UNTERE_toulding_element = :current_data_tata_building_element = :current_data_tata_building_element = :current_data_tata_building_element = :current_data_tata_building_element_aND
FIND building_element_attribute WHERE attribute_of building_element OF building_element attribute = new_attribute_name WHEN FOUND i = 1 WHEN NONE FOUND i = 2 FIND END IF i = 2 THEN BEGIN MAKE building_element_attribute WITH attribute_of_building_element := new_attribute_name location OF pushbutton 12 invisible2 := 25,265,225,285 location OF pushbutton 16 invisible2 := 0,0,0,0 END END **** end of method END ATTRIBUTE General Rules SIMPLE WHEN CHANGED BEGIN **** action OF ISodbe 1 IS connect := TRUE EXEC ISodbe 1 SQL SELECT distinct (building_element) from [AB-attribute-designer] END SQL INTO Building element visible OF main window := FALSE output OF configuration window := TRUE ***** isode of the suit of t	:AB general rules attribute of building element, :AB general rules affected discipline, :AB general rules percent change, :AB general rules result in major modification, :AB general rules result in major modification, :AB general rules cause of the effect) END SQL END END ELSE BEGIN text OF message box 2 := "Some important data are missing. You may need to puch New Rule botton" put up OF message box 2 := TRUE END END ATTRIBUTE new rule SIMPLE WHEN CHANGED BEGIN IF (CONF(building element OF Current rule data) >= 0 AND CONF(attribute of building element OF Current_rule data) >= 0) THEN BEGIN FORGET discipline OF input data MAKE AB general rules WITH building element := building element OF Current rule_data WITH stribute of building element := attribute of building element of Current_rule_data END END ATTRIBUTE delete a component SIMPLE WHEN CHANGED BEGIN IF CONF(building_element OF current_data) >= 0 THEN ATTRIBUTE delete a component SIMPLE WHEN CHANGED BEGIN IF CONF(building_element OF current_data) >= 0 THEN	i := END **** put in the database
FIND building_element_attribute WHERE attribute_of building_element OF building_element attribute = new_attribute_name WHEN FOUND i = 1 WHEN NONE FOUND i = 2 FIND END IF i = 2 THEN BEGIN MAKE building_element_attribute WITH attribute_of_building_element := new_attribute_name location OF pushbutton 12 invisible2 := 125,265,225,245 location OF pushbutton 16 invisible2 := 10,0,0,0 END END	:AB general rules attribute of building element, :AB general rules affected discipline, :AB general rules percent change, :AB general rules result in major modification, :AB general rules rule ignored, :AB general rules cause of the effect) END SQL END END END END END END END ATTRIBUTE new rule SIMPLE WHEN CHANGED BEGIN IF (CONF(building element OF Current rule data) >= 0 AND CONF(attribute of building element := building element OF Current rule data WITH building element := building element OF Current rule data WITH stribute of building element := attribute of building element of Current rule data WITH building element of Current rule data WITH stribute of building element := building element OF Current rule data WITH stribute of December of Current rule data END ATTRIBUTE delete a component SIMPLE WHEN CHANGED BEGIN IF CONF(building element OF current rule data END ATTRIBUTE delete a component SIMPLE WHEN CHANGED BEGIN IF CONF(building element OF current data) >= 0 THEN BEGIN	i := END **** put in the database
FIND building_element_attribute WHERE attribute = new_attribute_name WHEN FOUND i = 1 WHEN FOUND i = 2 FIND END IF i = 2 THEN BEGIN MAKE building_element_attribute WITH attribute_of_building_element := new_attribute_name location OF pushbutton 12 invisible2 := 25,265,225,285 location OF pushbutton 16 invisible2 := 0,0,0,0 END	:AB general rules attribute of building element. :AB general rules affected discipline. :AB general rules percent change. :AB general rules result in major modification. :AB general rules result in major modification. :AB general rules cause of the effect) END SQL END END END END ELSE BEGIN text OF message box 2 := "Some important data are missing. You may need to puch New Rule botton" put up OF message box 2 := TRUE END END ATTRIBUTE new rule SIMPLE WHEN CHANGED BEGIN IF (CONF(building element OF Current rule data) >= 0 AND CONF(attribute of building element OF Current rule data) >= 0) THEN BEGIN FORGET discipline OF input data MAKE AB general rules WITH building element := building element OF Current rule data WITH stribute of building element := attribute of building element of Current rule data END ATTRIBUTE delete a component SIMPLE WHEN CHANGED BEGIN IF CONF(building element OF current rule data) >= 0 THEN BEGIN general statement of sql := CONCAT("drop	i = 1
FIND building_element_attribute WHERE attribute_of building_element OF building_element attribute = new_attribute_name WHEN FOUND i = 1 WHEN NONE FOUND i = 2 FIND END IF i = 2 THEN BEGIN MAKE building_element_attribute WITH attribute_of_building_element := new_attribute_name location OF pushbutton 12 invisible2 := 25,265,225,285 location OF pushbutton 16 invisible2 := 0,0,0,0 END END END **** end of method END ATTRIBUTE General Rules SIMPLE WHEN CHANGED BEGIN **** action OF ISodbe 1 IS connect := TRUE EXEC ISodbe 1 SQL SELECT distinct (building_element) from [AB-attribute-designer] END SQL INTO Building element visible OF main window := FALSE output OF configuration window := TRUE **** end of method END ATTRIBUTE update rule SIMPLE	:AB general rules attribute of building element, :AB general rules affected discipline, :AB general rules percent change, :AB general rules result in major modification, :AB general rules result in major modification, :AB general rules cause of the effect) END SQL END END END ELSE BEGIN text OF message box 2 := "Some important data are missing. You may need to puch New Rule botton" put up OF message box 2 := TRUE END END ATTRIBUTE new rule SIMPLE WHEN CHANGED BEGIN IF (CONF(building element OF Current rule data) >= 0 AND CONF(attribute of building element OF Current rule data) >= 0 AND CONF(attribute of building element := building element OF Current rule data WITH building element := building element OF Current rule data WITH stribute of building element := attribute of building element oF Current rule data ATTRIBUTE delete a component SIMPLE WHEN CHANGED BEGIN IF CONF(building element OF current data) >= 0 THEN BEGIN GENERAL STRIBUTE delete a component SIMPLE WHEN CHANGED BEGIN IF CONF(building element OF current data) >= 0 THEN BEGIN general statement of sql := CONCAT("drop table ", building element OF current data)	i := 1 END
FIND building_element_attribute WHERE attribute = new_attribute_name WHEN FOUND i = 1 WHEN FOUND i = 2 FIND END IF i = 2 THEN BEGIN MAKE building_element_attribute WITH attribute_of_building_element := new_attribute_name location OF pushbutton 12 invisible2 := 25,265,225,285 location OF pushbutton 16 invisible2 := 0,0,0,0 END	:AB general rules attribute of building element, :AB general rules affected discipline, :AB general rules percent change, :AB general rules result in major modification, :AB general rules rule ignored, :AB general rules cause of the effect) END SQL END END END END END END END ATTRIBUTE new rule SIMPLE WHEN CHANGED BEGIN IF (CONF(building element OF Current rule data) >= 0 AND CONF(attribute of building element := building element OF Current rule data WITH building element == building element OF Current rule data WITH stribute of building element == attribute of building element of Current rule data WITH building element of Current rule data END ATTRIBUTE delete a component SIMPLE WHEN CHANGED BEGIN IF (CONF(building element of Current rule data WITH stribute of building element == building element OF Current rule data WITH SIMPLE delete a component SIMPLE WHEN CHANGED BEGIN IF CONF(building element OF current data) >= 0 THEN BEGIN general statement of sql := CONCAT("drop table ", building element of SQL WITH general statement of	i = 1
FIND building_element_attribute WHERE attribute_of building_element OF building_element attribute = new_attribute_name WHEN FOUND i = 1 WHEN NONE FOUND i = 2 FIND END IF i = 2 THEN BEGIN MAKE building_element_attribute WITH attribute_of_building_element := new_attribute_name location OF pushbutton 12 invisible2 := 25,265,225,285 location OF pushbutton 16 invisible2 := 0,0,0,0 END END END **** end of method END ATTRIBUTE General Rules SIMPLE WHEN CHANGED BEGIN **** action OF ISodbe 1 IS connect := TRUE EXEC ISodbe 1 SQL SELECT distinct (building_element) from [AB-attribute-designer] END SQL INTO Building element visible OF main window := FALSE output OF configuration window := TRUE **** end of method END ATTRIBUTE update rule SIMPLE	:AB general rules attribute of building element, :AB general rules affected discipline, :AB general rules percent change, :AB general rules result in major modification, :AB general rules result in major modification, :AB general rules cause of the effect) END SQL END END END ELSE BEGIN text OF message box 2 := "Some important data are missing. You may need to puch New Rule botton" put up OF message box 2 := TRUE END END ATTRIBUTE new rule SIMPLE WHEN CHANGED BEGIN IF (CONF(building element OF Current rule data) >= 0 AND CONF(attribute of building element OF Current rule data) >= 0 AND CONF(attribute of building element := building element OF Current rule data WITH building element := building element OF Current rule data WITH stribute of building element := attribute of building element oF Current rule data ATTRIBUTE delete a component SIMPLE WHEN CHANGED BEGIN IF CONF(building element OF current data) >= 0 THEN BEGIN GENERAL STRIBUTE delete a component SIMPLE WHEN CHANGED BEGIN IF CONF(building element OF current data) >= 0 THEN BEGIN general statement of sql := CONCAT("drop table ", building element OF current data)	i := 1 END

```
FORGET discipline OF input data location OF pushbutton 12 invisible2 := 0,0,0,0 location OF pushbutton 13 invisible2 := 0,0,0,0
      location OF prompthox 3 invisible2 := 0,0,0,0
       location OF border 7 invisible2 := 0,0,0,0
      location OF textbox 18 invisible2 := 0,0,0,0
                                                                Building element
      location OF textbox 19 invisible2 := 0,0,0,0
                                                                      FIND END
      location OF pushbutton 16 invisible2 := 0,0,0,0 location OF pushbutton 26 invisible2 := 0,0,0,0
                                                                     END
    FLSE
     BEGIN
      text OF message box 2 := "Some essential data
are missing. Check attribute name, type, and designing discipline."
                                                                  END
     put up OF message box 2 := TRUE
END
                                                                  REGIN
  END
ATTRIBUTE delete of attribute SIMPLE
 WHEN CHANGED
   BEGIN
IF CONF(attribute_of_building_element OF current_data) >= 0 THEN
     REGIN
                                                                  END
 *** delete from its table and from table [AB-attribute-
                                                                 WHEN CHANGED
designer)
          eral statement of aql := CONCAT( "alter
                                                                  REGIN
table [", building element OF current data, "] drop [", attribute of building element OF current data, "];")
      EXEC ISodbe 1 SQL WITH general states
sal END SOL
      EXEC (Sodbe | SQL delete from [AB-attribute
designer] where building eleme
current_data.building_element AND
                                                                 WHEN CHANGED
attribute of building element
current data attribute of building element END SQL
                                                                Current_projects
       delete the related data from Level 5
      FIND building_element_attrib
       WHERE attribute of building element OF
                                                                  END
building element attribute attribute of building element OF current data
                                                                SIMPLE
                                                                 WHEN CHANGED
       WHEN FOUND
        FORGET CURRENT
                                                                  BEGIN
building_element_attribute
      FIND END
      FORGET attribute_of_building_element OF
      FORGET attribute_type OF current_data
FORGET designing_discipline OF current_data
                                                                  END
                                                                ATTRIBUTE
      FORGET linked building element OF
current_data
      FORGET linked_attribute_of_building_element
                                                                  BEGIN
OF current_data
     END
    ELSE
     REGIN
      text OF message box 2 := "No Building
Component Attribute is currently choses
    put up OF message box 2 := TRUE
END
! **** end of method
  END
ATTRIBUTE Menus SIMPLE
                                                                  FND
 WHEN CHANGED
  BEGIN
                                                                 WHEN CHANGED
    action OF 15odbc 1 IS connect := TRUE
                                                                  BEGIN
    EXEC ISodbe 1 SQL SELECT distinct
(building_element) from [AB-attribute-designer]
END SOL INTO Building element
                                                                path of the directorey
                                                                'empty" database
    EXEC ISodbc 1 SQL SELECT distinct
(main_menu_label) from [AB-menu_item]
END SQL INTO Menus_group (label)
    EXEC ISodbe 1 SQL SELECT label from [AB-
menu_item]
                END SOLINTO
Menus_classified_B_comp
                                                                database to a file with
    FOR (i:= 1 TO number instances OF navigate
Building element)
     BEGIN
      instance number OF navigate Building element
                                                               project.ini
```

FIND Menus_classified_B_comp

```
WHERE label OF Menus_classified_B_comp
        ing_element OF Building element
WHEN NONE FOUND
                                                                        read line OF file 1 := TRUE
                                                                         IF SUBSTR(current line OF file 1, 1, 16) =
         MAKE Menus unclassified B comp
                                                                  *path of ODBC.ini* THEN
                                                                  path_of_odbc := SUBSTR( current line OF file 1, 36, 11)
          WITH label := building_ele
                                                                        IF SUBSTR(current line OF file 1, 1, 29) =
                                                                  *path of the developed program* THEN
current_path := SUBSTR( current line OF file
                                                                  1, 36, 16)
    visible OF main window := FALSE
    output OF configuration window := Menus display visible OF configuration window := TRUE
                                                                      END
                                                                      action OF file 1 IS close := TRUE

    modify the ODBC.ini

                                                                      filename OF file 3 := CONCAT( path_of_odbc,
ATTRIBUTE project_name STRING
WHEN CHANGED
                                                                  "ODBC.INI")
action OF file 3 IS open old := TRUE
                                                                      WHILE (NOT of OF file 3)
    data source OF ISodbc 1 := project_name
title OF configuration window := CONCAT(
                                                                       BEGIN
                                                                        read line OF file 3 := TRUE
*Configuring Project : *, project name)
attribute attachment OF pushoutton 38 :=
                                                                       END
                                                                      write line OF file 3 := CONCAT( "[".
   eck_the_existance_of_new_project_name
display attachment OF pushbutton 37 :=
                                                                 project name, "]")
                                                                      write line OF file 3 := CONCAT( "Driver=",
CURENT_project_selection_display
                                                                 path_of_odbc, "SYSTEM/ODBCJT16.DLL")
write line OF file 3 := CONCAT( "DBQ=",
                                                                 project_name, *.MDB*)
write line OF file 3 := CONCAT( "DefaultDir=",
ATTRIBUTE New_project SIMPLE
                                                                 current path)
                                                                      write line OF file 3 := "FIL=Microsoft Access"
    action OF ISodbc 2 IS connect := TRUE
                                                                      write line OF file 3 := "JetIniPath=msacc20.ini"
    EXEC !Sodbc 2 SQL select Project_name from
                                                                     write line OF file 3 := "UID=Admin"
write line OF file 3 := " "
Current_projects
END SQL INTO Current_projects
                                                                      action OF file 3 IS close := TRUE
ATTRIBUTE Current_project SIMPLE
                                                                          add the new file to the general database
                                                                     EXEC ISodbe 2 SQL insert into Current_projects
    action OF ISodbc 2 IS connect := TRUE
                                                                 (Project_name) values ( :domain.project_name )
   EXEC ISodbc 2 SQL select Project_name from
                                                                         make a batch file to have a new database
       END SQL INTO Current_projects
                                                                      action OF file 2 IS delete := TRUE
                                                                     action OF file 2 IS open new := TRUE write line OF file 2 := CONCAT( "copy",
ATTRIBUTE back_to_current_project_display
                                                                 current_path, "empty.mdb ", current_path, project_name, ".mdb ")
                                                                      write line OF file 2 := CONCAT( "copy ",
   visible OF configuration window := FALSE visible OF main window := TRUE
                                                                   urrent_path, "empty.kdb ", current_path, project_name,
   output OF main window :=
                                                                  ' Нь ")
                                                                     action OF file 2 IS close := TRUE
CURENT_project_selection_display
                                                                     check the existance of new project name SIMPLE WHEN CHANGED
                                                                         - connect the display to the window
                                                                      output OF main wind
   FIND Current_projects
                                                                 NEW_project_selection_display
     WHERE project_name OF Current_projects =
     WHEN FOUND
                                                                   *** end of method
      text OF message box 2 := "This name already
                                                                    END
                                                                 ATTRIBUTE path of odbe STRING
ATTRIBUTE current path STRING
ATTRIBUTE ii NUMERIC
exists for a current project.
Type another name please. *
      put up OF message box 2 := TRUE
                                                                  ATTRIBUTE un STRING
     WHEN NONE FOUND
      setup a new project :- TRUE
                                                                  ATTRIBUTE get from general database SIMPLE
                                                                   WHEN CHANGED
                                                                    BEGIN
      end of method
ATTRIBUTE setup a new project SIMPLE
                                                                     EXEC ISodhe 2 SOL select distinct
                                                                  (building_element) from [General-AB-attribute-
                                                                  designer
                                                                          END SOL INTO
! ***** this method does the following:
! ***** 1) Read the paths of the ODBC.ini file and the
                                                                  GENERAL_Building_elen
         that is used for this program and contains an
                                                                    **** remove the building elements that are already in
       2) Modify the ODBC.[NI file so it links to the
                                                                      FOR (i := 1 TO number instances OF navigate
new database [ ***** 3] Add the file name to the General database so
                                                                  Building element)
                                                                       BEGIN
                                                                        instance number OF navigate Building element
as to be considered existing project
! ***** 4) Prepare a batch file to exceute the next step ! **** 5) Make a new database by copy the empty
                                                                        FIND GENERAL_Building_element
                                                                  WHERE building element OF
GENERAL Building element = building element OF
          the new name
                                                                  Building element
                                                                          WHEN FOUND
                                                                           FORGET CURRENT
        read the path of the files from the file
                                                                  GENERAL_Building_element
    action OF file 1 IS open old := TRUE
WHILE (NOT cof OF file 1)
                                                                        FIND END
```

```
GENERAL_building_element_attribute) = 100 THEN
     visible OF get from general database window :=
                                                                                    BEGIN
                                                                                                                                                    IF i = I THEN
                                                                                                                                                      BEGIN
TRUE
                                                                                   ***** the following step is necessary as data
                                                                                                                                                       text OF n
                                                                                                                                                                      age box 2 := "This Compo
    output OF get from general database window := add
                                                                                                                                                     put up OF message box 2 := TRUE END
 from general database display
                                                                        can not be got directly from access to this attribute
EXEC ISodbe 2 SOL SELECT
                                                                                                                                                Already exists*
! *** end of method
                                                                        (linked_attribute_of_building_element) from
                                                                        [GENERAL-AB-attribute-designer]
   END
                                                                        where building_element =
:domain.GENERAL_Building_component and
attribute_of_building_element =
:GENERAL_building_element_attribute.attribute_of_b
 ATTRIBUTE GENERAL_Building_component
                                                                                                                                                    IF i = 2 THEN
STRING
  WHEN CHANGED
                                                                                                                                                     BEGIN
   BEGIN
     EXEC ISodbe 2 SQL SELECT * from [General-
                                                                        uilding element
                                                                                                                                                ! **** create the database table with its id attribute
                                                                                                                                               general statement of sql := CONCAT("CREATE
TABLE [", GENERAL_Building_component, "] [[",
GENERAL_Building_component, "_id] TEXT(50));")
EXEC [Sodbc 1 SQL WITH general statement of
AB-attribute-designer) where building_elen:domain.GENERAL_Building_component
                                                                                           END SOL INTO domain (tttt)
                                                                                    linked attribute of building element OF
                   END SQL INTO
                                                                        GENERAL building element_attribute := ttt
GENERAL building element attribute
                                                                                     EXEC ISodbe I SQL INSERT INTO [AB-
                                                                                                                                                sql END SQL
   END
                                                                        attribute-designer] (building element,
attribute_of_building element, attribute_type,
designing_discipline, linked_building_element,
ATTRIBUTE add component from _get from general
                                                                                                                                                1 **** add the rest of attributes to the created table
 datab SIMPLE
                                                                                                                                                      FOR (ii := 1 TO number instances OF navigate
  WHEN CHANGED
                                                                        linked_attribute_of_building_element) values
                                                                                                                                                GENERAL building element attributes)
   BEGIN
                                                                        :domain.GENERAL_Building_component,
:GENERAL_building_element_attribute_attribute_of_b
                                                                                                                                                        BEGIN
     FIND Building element
                                                                                                                                                          instance number OF navigate GENERAL
                                                                                                                                               building element attributes := ii

! **** ignore the attribute that has the _id of the
      WHERE LOWCASE( building_element OF
                                                                        uilding element,
GENERAL building element attribute attribute type,
Building element) = LOWCASE(
GENERAL Building component)
WHEN FOUND
                                                                        GENERAL building element attribute designing dis-
cipline: GENERAL building element attribute linked
building element; GENERAL building element attri
                                                                                                                                                         IF SEARCHSTR("_id".
                                                                                                                                               attribute of building element OF
GENERAL building element attribute) = 0 THEN
      WHEN NONE FOUND
                                                                        bute.linked_attribute_of_building_element)
                                                                                                                                                          BEGIN
                                                                                                                                                           **** put the attributes in the database
    FIND END
                                                                                                                                                            general statement of sql := CONCAT( "alter
                                                                                                                                               table [", building_element OF
GENERAL_Building_element, "] add [",
                                                                                                                            END
    IF i = I THEN
                                                                        SOL
                                                                                                                                               GENERAL_Building_element of GENERAL_building_element of GENERAL_building_element attribute, "] ")

If attribute_type Of GENERAL_building_element_attribute = "\" THEN
                                                                                   END
      BEGIN
       text OF message box 2 := "This Component
                                                                                   BEGIN
Already exists'
       put up OF message box 2 := TRUE
                                                                                    EXEC ISodbe I SQL INSERT INTO [AB-
                                                                        attribute-designer] (building_element, attribute_type, designing_discipline) values
                                                                                                                                                             general statement of sql := CONCAT(
      END
                                                                                                                                                      al statement of sql, "TEXT (50);")
                                                                                                                                                            IF attribute_type OF
                                                                                                                                               GENERAL building element annibute = "n" THEN
general statement of sql := CONCAT(
    IF i = 2 THEN
                                                                        :domain.GENERAL_Building_component,
:GENERAL_building_element_attribute_attribute_of_b
     BEGIN
                                                                                                                                                   neral statement of sql, "Number;")

EXEC ISodbc 1 SQL WITH general
1 **** create the database table with its id attribute
general statement of sqt := CONCAT( "CREATE
TABLE [", GENERAL Building component, "] ([",
GENERAL Building component, "_id] TEXT(50));")
EXEC ISodbe 1 SQL WITH general statement of
                                                                        GENERAL building element attribute attribute type, GENERAL building element attribute designing dis
                                                                                                                                               statement of soi END SOL
                                                                                                                                                        END
                                                                                                                                                 **** add the attributes to the table [AB-attribute-
sql END SOL
                                                                                                                                               designer] in the new project

FOR (ii := 1 TO number instances OF navigate
                                                                                                                            END
! **** add the rest of attributes to the created table
                                                                        SOL
                                                                                   END
                                                                                                                                               GENERAL building element attributes)
      FOR (ii := 1 TO number instances OF navistate
GENERAL building element attributes)
                                                                                                                                                        BEGIN
                                                                                END
         RECIN
                                                                             FND
                                                                                                                                                         instance number OF navigate GENERAL.
          instance number OF navigate GENERAL
                                                                                                                                               building element attributes := ii
                                                                        1 *** end of method
                                                                                                                                                         IF CONF(linked_building_element OF
building element attributes := ii
                                                                                                                                               GENERAL_building_element_attribute) = 100 THEN
        I **** ignore the attribute that has the _id of the
                                                                          END
                                                                        ATTRIBUTE return from _get from general datab
                                                                                                                                                           BEGIN
          IF SEARCHSTR("_id",
                                                                        SIMPLE
                                                                          WHEN CHANGED
                                                                                                                                                           ***** the following step is necessary as data
attribute_of_building_element_oF
GENERAL_building_element_attribute) = 0 THEN
                                                                           BEGIN
                                                                                                                                               can not be got directly from access to this attribute
           BEGIN
                                                                                                                                                            EXEC Isodbe 2 SOL SELECT
         ! **** put the attributes in the database
                                                                            EXEC (Sodbe 1 SOL SELECT distinct
                                                                                                                                                (linked attribute of building element) from
            general statement of sql := CONCAT( "alter
                                                                                                                                               [GENERAL-AB-attribute-designer]
                                                                        (building_element), graph_name from [AB-attribute-
table [", building_element OF
                                                                                                                                                           where building_element
                                                                        designer]
                                                                               END SQL INTO Building element
                                                                                                                                                domain.GENERAL Building component and
GENERAL Building element, "] add [", attribute of building element OF
                                                                                                                                                attribute_of_building_element
GENERAL building element attribute, "] ")

IF stribute type OF
                                                                                                                                                :GENERAL_building_element_attribute.attribute_of_b
                                                                            EXEC ISodbe 1 SQL SELECT distinct
                                                                        (building_element) from [AB-attribute-designer]
END SQL INTO Linked building element
                                                                                                                                                uilding element
                                                                                                                                                                   END SQL INTO domain (tm)
GENERAL building element attribute = "t" THEN
                                                                                                                                                linked_attribute_of_building_element OF
GENERAL_building_element_attribute := tmt
              general statement of sqi := CONCAT(
general statement of sql, "TEXT (50);")

IF sttribute_type OF
                                                                             visible OF get from general database window:
                                                                                                                                                            EXEC Isodbe 1 SOL INSERT INTO (AB-
GENERAL_building_element_attrib
                                                                           END
                                                                                                                                                attribute-designer] (building element,
attribute of building element, attribute type,
designing discipline, linked building element,
              general statement of sql := CONCAT(
                                                                         ATTRIBUTE add component and general rule
general statement of sql, "Number,")

EXEC Boobe 1 SQL WITH general
                                                                          WHEN CHANGED
                                                                                                                                                linked attribute of building element) values
statement of sol END SOL
                                                                           BEGIN
                                                                                                                                                domain.GENERAL_Building_component,
GENERAL_building_element_attribute.attribute_of_b
                                                                            FIND Building element
         END
                                                                              WHERE LOWCASE( building element OF
                                                                                                                                                uilding_element,
:GENERAL_building_element_attribute.attribute_type,
  *** add the attributes to the table [AB-attribute-
                                                                        Building element) = LOWCASE(
                                                                        GENERAL_Building_component)
WHEN FOUND
designer] in the new project
FOR (ii := 1 TO number instances OF navigate
                                                                                                                                                GENERAL building element auribute designing dis
cipline, GENERAL building element auribute linked
building element, GENERAL building element auri-
GENERAL building element attributes)
         BEGIN
                                                                              WHEN NONE FOUND
                                                                                                                                                bute linked attribute of building element )
          instance number OF navigate GENERAL
                                                                             FIND END
building element attributes := ii
```

IF CONF(linked_building_element OF

WITH style IS group WITH perspective IS in WITH border width := 1 END WITH perspective IS in WITH border width := 4 SQL END WITH background color := 192,192,192 WITH fill color := 0,0,128 **ELSE** WITH background color := 192,192,192 WITH fill color := 192,192,192 WITH highlight color := 0,0,0 WITH shadow color := 0,0,0 BEGIN EXEC Isodbe 1 SQL INSERT INTO [AB-attribute-designer] (building element, attribute_of_building_element, attribute_type, WITH highlight color := 255,255,255 WITH shadow color := 128,128,128 WITH location := 15,295,490,400 WITH location := 5,5,165,75 designing discipline) values INSTANCE border 14 ISA border :domain.GENERAL Building component INSTANCE border 5 ISA border WITH style IS shade WITH style IS picture frame WITH perspective IS in WITH perspective IS in
WITH border width = 4
WITH background color = 192,192,192 GENERAL building element attribute attribute of b WITH border width := 4 uikling element, GENERAL buikling element attribute attribute type, GENERAL buikling element attribute designing dis WITH background color := 192,192,192 WITH fill color := 0,0,12\$ WITH highlight color := 0,0,0 WITH shadow color := 0,0,0 WITH fill color := 192,192,192 WITH highlight color := 255,255,255 WITH shadow color := 128,128,128 cioline) WITH location := 10,5,170,60 WITH location := 5,40,595,245 END INSTANCE border 15 ISA border SQL INSTANCE border 6 invisible ISA border WITH style IS group WITH perspective IS in END WITH style IS picture frame WITH border width := 1 END WITH perspective IS in WITH border width := 1 WITH background color := 192,192,192 **** Add the General Rules WITH background color := 192,192,192 WITH fill color := 192,192,192 WITH fill color := 192,192,192
WITH highlight color := 255,255,255
WITH shadow color := 128,128,128 WITH highlight color := 255,255,255 WITH shadow color := 128,128,128 EXEC ISodbe 2 SQL select * from [GENERAL-AB-general-rules] where building element domain.GENERAL_Building component WITH location := 5,40,490,400 WITH location := 0,0,0,0 END SQL INTO AB general rules INSTANCE border 16 ISA border WITH style IS group
WITH perspective IS in
WITH border width := 1 INSTANCE border 7 invisible 2 ISA border WITH style IS picture frame WITH perspective IS in WITH border width := 1 ! **** add the rules to the new project database FOR (ii := 1 TO number instances OF rules WITH background color := 192,192,192 WITH background color := 192,192,192 WITH fill color := 192,192,192 WITH fill color := 192,192,192 WITH highlight color := 255,255,255 WITH shadow color := 128,128,128 navigate 2) BEGIN WITH highlight color := 255,255,255 WITH shadow color := 128,128,128 WITH location := 0,0,0,0 nce number OF rules navigate 2 := ii EXEC ISodbe | SQL INSERT INTO [AB-WITH location := 10.45,485,395 EXEC Isodho I SQL INSERT INTO [AB-general-rules] (building_element, affected_discipline, percent_change, result in_major_modification, cause_of_the_effect) values INSTANCE border 17 ISA border WITH style IS picture frame WITH perspective IS in WITH border width := 3 INSTANCE border 8 ISA border WITH style IS edit control (:AB_general_rules.building_element, WITH perspective IS in :AB general rules attribute of building element, :AB general rules affected discipline, WITH border width := 1 WITH background color := 192,192,192 WITH fill color := 192,192,192 WITH background color := 192,192,192 :AB general rules percent change,
:AB general rules result in major modification,
:AB general rules cause of the effect) WITH highlight color := 255,255,255 WITH shadow color := 128,128,128 WITH location := 20,55,475,195 WITH fill color := 192,192,192 WITH highlight color := 255,255,255 WITH shadow color := 128,128,128 WITH location := 5,50,250,395 INSTANCE border 18 ISA border . END INSTANCE border 9 ISA border WITH style IS picture frame WITH perspective IS in WITH border width := 3 WITH background color := 192,192,192 SQL WITH style IS edit control END WITH perspective IS in WITH border width := 1 WITH fill color := 192,192,192
WITH highlight color := 255,255,255
WITH shadow color := 128,128,128 WITH background color := 192,192,192 end of IF WITH fill color := 192,192,192
WITH fill color := 192,192,192
WITH highlight color := 255,255,255
WITH shadow color := 128,128,128
WITH location := 253,50,500,395 END WITH location := 20,205,475,345 end of method END INSTANCE border 19 ISA border **INSTANCE** border 10 ISA border WITH style IS group INSTANCE border 1 ISA border WITH perspective IS in WITH border width := 1 WITH style IS group WITH perspective IS in WITH style IS shadow WITH perspective IS in WITH border width := 4 WITH border width := 1 WITH background color := 192,192,192 WITH fill color := 192,192,192 WITH highlight color := 255,255,255 WITH shadow color := 128,128,128 WITH location := 5,40,490,400 WITH background color := 192,192,192 WITH background color := 192,192,192 WITH fill color := 192,192,192 WITH highlight color := 255,255,255 WITH shadow color := 128,128,128 WITH location := 5,45,175,415 WITH fill color := 0.0.128 WITH highlight color := 0,0,0 WITH shadow color := 0,0,0 WITH location := 5,5,165,100 INSTANCE border 20 ISA border INSTANCE border 11 ISA border WITH style IS group WITH perspective IS in INSTANCE border 2 ISA border WITH style IS group WITH style IS group WITH perspective IS in WITH border width := I WITH perspective IS in WITH border width := 1 WITH border width := 1 WITH background color := 192,192,192 WITH background color: = 192,192,192 WITH fill color := 192,192,192 WITH background color := 192,192,192 WITH fill color := 192,192,192 WITH highlight color := 255,255,255 WITH shadow color := 128,128,128 WITH location := 10,45,485,395 WITH fill color := 192,192,192 WITH highlight color:= 255,255,255 WITH shadow color:= 128,128,128 WITH highlight color := 255,255,255 WITH shadow color := 128,128,128 WITH location := 5,40,420,135 WITH location := 180,45,340,415 INSTANCE border 21 ISA border INSTANCE border 12 ISA border WITH style IS picture frame WITH style IS group WITH perspective IS in WITH border width := 1 INSTANCE border 3 ISA border WITH perspective IS in WITH style IS group WITH border width := 3 WITH perspective IS in WITH background color := 192,192,192 WITH fill color := 192,192,192
WITH highlight color := 255,255,255
WITH shadow color := 128,128,128 WITH background color := 192,192,192 WITH fill color := 192,192,192 WITH border width := 1 WITH background color := 192,192,192 WITH fill color:= 192,192,192
WITH highlight color:= 255,255,255
WITH shadow color:= 128,128,128 WITH highlight color := 255,255,255 WITH shadow color := 128,128,128 WITH location := 20.110.240.385 WITH location := 345,45,500,355 WITH location := 5,140,500,410 INSTANCE border 22 ISA border INSTANCE border 13 ISA border WITH style IS picture frame

WITH style IS shadow

INSTANCE border 4 ISA border

WITH perspective IS in

WITH heading justify IS heading center WITH heading wordwrap := TRUE WITH border width := 3 INSTANCE ISodbe 2 ISA ISodbe WITH data source := "general"
WITH connection prompting := FALSE
WITH auto commit := TRUE WITH background color := 192,192,192 WITH fill color := 192,192,192 WITH highlight color := 255,255,255 WITH shadow color := 128,128,128 INSTANCE column 3 ISA column WITH append := FALSE
WITH default error handling := TRUE WITH attachment := designing discipline OF milding_element_attribute WITH width := 60 WITH location := 255,110,475,320 WITH justify IS left INSTANCE border 23 ISA border INSTANCE ISodbe 3 ISA ISodbe WITH style IS picture frame WITH perspective IS in WITH border width := 3 WITH connection prompting := FALSE WITH auto commit := TRUE WITH wordwrap := TRUE WITH heading label := "Design WITH append := FALSE
WITH default error handling := TRUE Discipl' WITH heading justify IS heading center WITH heading wordwrap := TRUE WITH background color := 192,192,192 WITH fill color := 0,128,128
WITH fill color := 0,128,128
WITH shadow color := 255,255,255
WITH shadow color := 128,128,128
WITH location := 20,55,475,100 INSTANCE message box I ISA message box WITH title := "Warning Message"
WITH text := "Are you sure that you want to
DELETE the current building component" INSTANCE column 4 ISA column
WITH attachment := linked building element OF building_element_attribute **INSTANCE** border 24 ISA border WITH modal style IS sys modal WITH width - 160 WITH icon IS question WITH button IS yes no WITH style IS group WITH justify IS left WITH perspective IS in WITH border width := 1 WITH wordwrap := TRUE WITH default button IS one WITH heading label := "Linked WITH background color := 192,192,192 Building WITH fill color := 192,192,192 INSTANCE message box 2 ISA message box WITH title := "INPUT ERROR" Compone WITH highlight color := 255,255,255 WITH shadow color := 128,128,128 WITH heading justify IS heading center WITH heading wordwrsp := TRUE WITH modal style IS app modal WITH icon IS stop sign WITH location := 5,40,415,340 WITH button IS ok WITH default button IS one INSTANCE column 5 ISA column INSTANCE border 25 ISA border WITH attachment : WITH style IS group linked attribute of building element OF INSTANCE message box 3 ISA message box WITH title := "Warning Message" WITH text := "Are you sure that you want to building element attribute WITH width := 160 WITH justify IS left WITH perspective IS in WITH border width:= 1 WITH border width := 1
WITH background color := 192,192,192
WITH fill color := 192,192,192
WITH highlight color := 255,255,255
WITH shadow color := 128,128,128 WITH wordwrap := TRUE WITH heading label := "Linked DELETE the current component attribute WITH modal style IS sys modal Component WITH icon IS question WITH button IS yes no WITH location := 10,45,410,335 Attribute* WITH heading justify IS heading center WITH heading wordwrap := TRUE INSTANCE edit table column 1 ISA edit table column WITH attachment = attribute_of_building_element OF building_element_attribute WITH heading_label := WITH default button IS one INSTANCE nevigate 1 ISA navigate
WITH class attachment := building_ele INSTANCE Starting display ISA display WITH wait :- TRUE WITH wait := TRÜE
WITH delay changes := FALSE
WITH items [1] := UNDETERMINED
WITH items [2] := border |
WITH items [2] := border |
WITH items [3] := texthox |
WITH items [6] := UNDETERMINED
WITH items [6] := UNDETERMINED
WITH items [6] := UNDETERMINED
WITH items [7] := UNDETERMINED
WITH items [9] := pushbutton 34
WITH items [1] := pushbutton 35
WITH itel [1] := PUNDETERMINED
WITH items [1] := UNDETERMINED
WITH items [1] := UNDETERMINED "attribute_of_building_ele WITH width := 70 INSTANCE rules navigate 2 ISA navigate WITH class attachment := AB_general_rules WITH justify IS left WITH pen color [1] := UNDETERMINED WITH fill color [1] := UNDETERMINED INSTANCE navigate Building element ISA navigate
WITH class attachment := Building element INSTANCE edit table column 2 ISA edit table column WITH attachment := attribute_type OF building_element_attribute INSTANCE navigate menu_group_items ISA navigate WITH class attachment := Menus_group_items WITH heading label := "attribute_type"
WITH width := 70 INSTANCE pavigate building element attributes ISA WITH justify IS left navigate
WITH class attachment := building_element_attribute WITH pen color [1] := UNDETERMINED
WITH fill color [1] := UNDETERMINED WITH menus [1]:= UNDETERMINED INSTANCE navigate GENERAL building elements INSTANCE config ISA display WITH wait := FALSE ISA navigate
WITH class attachment := INSTANCE edit table column 3 ISA edit table col WITH attachment := designing_discipline OF WITH delay changes := FALSE WITH items [1] := textbox 2 building_element_attribute GENERAL Building element WITH items [2]:= border 2
WITH items [3]:= listbox 1
WITH items [4]:= valuebox 1
WITH items [5]:= textbox 3
WITH items [6]:= textbox 4 WITH heading label := "designing_discipline" WITH width := 70 INSTANCE navigate GENERAL building element WITH justify IS left attributes ISA navigate WITH pen color [1] := UNDETERMINED WITH fill color [1] := UNDETERMINED WITH class attachment :-GENERAL building element attribute WITH items [7] := border 3 INSTANCE edit table column 4 ISA edit table column WITH attachment := linked_building_element OF INSTANCE the application ISA application WITH items [8]:= listbox 2 WITH items [9]:= textbox 5 WITH unknowns fail := TRUE building_element_attribute WITH threshold := 50 WITH items [10]:= valuebox 2 WITH items [11] := border 4 WITH items [12] := textbox 6 WITH title display := Starting display WITH heading label := "linked_building_element" WITH width := 70 WITH ignore breakpoints := FALSE WITH items [13] = radiobution group 2
WITH items [14] := UNDETERMINED
WITH items [15] := textbox 7 WITH justify IS left WITH reasoning on := FALSE WITH pen color [1]: UNDETERMINED
WITH fill color [1]: UNDETERMINED WITH numeric precision := 8 WITH demon strategy IS fire first WITH items [16] := prompthox I
WITH items [17] := textbox 8
WITH items [18] := pushbutton 3 INSTANCE column 1 ISA column INSTANCE edit table column 5 ISA edit table column WITH attachment := attribute_of_building_element WITH attachment ? WITH items [19] = pushbutton 4
WITH items [20] := UNDETERMINED
WITH items [21] := listbox 3 linked_attribute_of_building_element OF OF building_element_attribute building_element_attribute WITH heading label := WITH width = 160 WITH justify IS left WITH wordwrap := TRUE
WITH heading label := "Attribute"
WITH heading justify IS heading center
WITH heading wordwrap := TRUE WITH items [22] := textbox 9
WITH items [23] := listbox 4
WITH items [24] := valuebox 3 "linked_attribute_of_building_element" WITH width = 70 WITH justify IS left WITH pen color [1]:= UNDETERMINED WITH fill color [1]:= UNDETERMINED WITH items [25] := valuebox 4 WITH items [26] := textbox 10 WITH items [27] := textbox 11 INSTANCE column 2 ISA column WITH items [28] := valuebox 5 WITH items [29] := valuebox 6 INSTANCE ISodbe 1 ISA ISodbe WITH attachment := attribute_type OF building_element_attribute WITH connection prompting := FALSE WITH auto commit := TRUE WITH width := 40 WITH items [30] := textbox 12 WITH append := FALSE
WITH default error handling := TRUE WITH items [31] := textbox 13 WITH items [32] := pushbutton 5 WITH items [33] := pushbutton 6 WITH justify IS left WITH wordwrap := TRUE WITH heading label := "Type"

```
WITH items [34] := textbox 14
WITH items [35] := radiobutton group 1
                                                                                                                                                                                                    WITH items [8 ] := textbox 37
WITH items [9 ] := pushbutton 27
                                                                                                                                                                                                                                                                                                                                                                                                 WITH wait := FALSE
       WITH items [36]:= reshbutton group i
WITH items [36]:= pushbutton 15
WITH items [37]:= pushbutton 15
WITH items [38]:= UNDETERMINED
WITH items [39]:= border 6 invisible
                                                                                                                                                                                                                                                                                                                                                                                                WITH dear changes = FALSE
WITH items [1] := border 15
WITH items [2] := border 16
WITH items [3] := border 17
WITH items [4] := border 18
WITH items [5] := textbox 53
WITH items [5] := textbox 53
WITH items [6] := textbox 53
                                                                                                                                                                                                     WITH items [10]:= listbox 10
                                                                                                                                                                                                     WITH items [11]:= textbox 38
WITH items [12]:= valuebox 13
                                                                                                                                                                                                     WITH items [13] := textbox 39
WITH items [14] := textbox 40
WITH items [15] := pushbutton 28
       WITH items [40]: "Fromption 2 invisible
WITH items [40]: "Fromption 2 invisible
WITH items [41]: "Fromption 17 invisible
WITH items [42]: "Fromption 10 invisible
WITH items [43]: "UNDETERMINED
WITH items [43]: "Fromption 11 invisible
WITH items [45]: "Fromption 11 invisible
WITH items [45]: "Fromption 11 invisible
WITH items [45]: "Fromption 17 invisible
WITH items [45]: "Fromption 17 invisible
WITH Items [45]: "With items
                                                                                                                                                                                                                                                                                                                                                                                                WITH items [6] = textbox 13
WITH items [6] := textbox 54
WITH items [7] := textbox 54
WITH items [8] := valuebox 16
WITH items [9] := textbox 55
WITH items [10] := UNDETERMINED
WITH items [11] := UNDETERMINED
                                                                                                                                                                                                     WITH items [16] := pushbutton 29
WITH items [17] := valuebox 14
WITH items [18] := pushbutton 30
                                                                                                                                                                                                   WITH items [19]: pushbutton 31
WITH items [20]:= pushbutton 32
WITH items [21]:= valuebox 15
WITH items [22]:= prompthox 8
WITH items [23]:= textbox 41
       WITH items [46]:= prompthox 3 invisible2
WITH items [47]:= tembox 18 invisible2
      WITH items [49] :- tembox 19 invisible2
WITH items [49] :- pushbutton 12 invisible2
WITH items [50] :- UNDETERMINED
WITH items [51] :- UNDETERMINED
                                                                                                                                                                                                                                                                                                                                                                                               WITH items [12] := listbox 14
WITH items [13] := textbox 57
WITH items [14] := pushbutton 47
WITH items [15] := pushbutton 48
WITH items [16] := UNDETERMINED
WITH fill color := 192,192,192
WITH older [1] := UNDETERMINED
                                                                                                                                                                                                    WITH tiems [24]:= textbox 42
WITH fill color:= 192,192,192
WITH menus [1]:= UNDETERMINED
      WITH items [52] := pushbutton 16 invisible2
WITH items [53] := pushbutton 26 invisible2
WITH items [54] := pushbutton 52
WITH items [55] := textbox 64
WITH fill color := 192,192,192
                                                                                                                                                                                                INSTANCE CURENT_project_start_display ISA
                                                                                                                                                                                                                                                                                                                                                                                                WITH menus [1]:= UNDETERMINED
                                                                                                                                                                                               display
WITH wait := FALSE
                                                                                                                                                                                                                                                                                                                                                                                            INSTANCE NEW from systems display ISA display WITH wait := FALSE
       WITH menus [1]:= UNDETERMINED
                                                                                                                                                                                                     WITH delay changes := FALSE
                                                                                                                                                                                                    WITH items [1]: = border 1
WITH items [2]: = UNDETERMINED
                                                                                                                                                                                                                                                                                                                                                                                                WITH delay changes := FALSE
WITH items [1] := UNDETERMINED
  INSTANCE attributes table display ISA display
                                                                                                                                                                                                                                                                                                                                                                                               WITH items [3] = UNDETERMINED
WITH items [3] = UNDETERMINED
WITH items [4] = UNDETERMINED
WITH items [5] = UNDETERMINED
WITH items [6] = UNDETERMINED
WITH items [6] = UNDETERMINED
WITH items [7] = UNDETERMINED
                                                                                                                                                                                                    WITH items [3 ] := UNDETERMINED WITH items [4 ] := UNDETERMINED WITH items [5 ] := UNDETERMINED
       WITH wait := FALSE
     WITH wait: FALSE
WITH delay changes: FALSE
WITH items [1] = border 5
WITH items [2] = table 1
WITH items [3] = textbox 15
WITH items [4] = textbox 16
WITH items [5] = UNDETERMINED
WITH items [6] := pushbutton 8
WITH fill color: = 192,192,192
WITH manufact [1] = INDETERMINED
                                                                                                                                                                                                   WITH items [6] = textbox 43
WITH items [7] = textbox 44
WITH items [7] = textbox 44
WITH items [8] = fusibution 33
WITH items [9] = fusibution 37
WITH items [10] := pushbution 37
WITH fill color = 192,192,192
WITH items [11] = INDESTER M
                                                                                                                                                                                                                                                                                                                                                                                               WITH items [8] = UNDETERMINED
WITH items [9] := textbox 58
WITH items [10] := border 19
WITH items [11] := border 20
WITH items [12] := border 21
                                                                                                                                                                                                     WITH menus [1]:= UNDETERMINED
       WITH memis [1] := UNDETERMINED
INSTANCE General rules display ISA display
WITH wait: = FALSE
WITH delay changes: = FALSE
WITH items [1]: = textbox 20
WITH items [2]: = border 8
WITH items [3]: = UNDETERMINED
WITH items [4]: = UNDETERMINED
WITH items [6]: = textbox 23
WITH items [6]: = textbox 24
WITH items [6]: = textbox 24
WITH items [6]: = textbox 24
WITH items [8]: = promptbox 4
WITH items [8]: = promptbox 4
WITH items [9]: = textbox 26
WITH items [11]: = textbox 26
WITH items [11]: = textbox 26
WITH items [12]: = promptbox 6
WITH items [13]: = listbox 5
WITH items [14]: = UNDETERMINED
WITH items [15]: = textbox 27
WITH items [15]: = textbox 27
                                                                                                                                                                                              INSTANCE NEW_project_start_display ISA display WITH wait:=FALSE
WITH delay changes:=FALSE
WITH items [1]:= border 1
WITH items [2]:= textbox 47
WITH items [3]:= pushbutton 33
WITH items [4]:= textbox 48
WITH items [5]:= promptbox 9
WITH items [6]:= pushbutton 38
WITH fill color:= 192,192,192
WITH menus [1]:= UNDETERMINED
                                                                                                                                                                                                                                                                                                                                                                                                WITH items [13] := border 22
                                                                                                                                                                                                                                                                                                                                                                                                WITH items [14]:= textbox 59
WITH items [15]:= textbox 60
                                                                                                                                                                                                                                                                                                                                                                                                WITH items [16]:= textbox 61
WITH items [17]:= valuebox 17
WITH items [18]:= textbox 62
                                                                                                                                                                                                                                                                                                                                                                                                WITH items [19] := valuebox 18
                                                                                                                                                                                                                                                                                                                                                                                                WITH items [20] := listbox 15
WITH items [21] := listbox 16
                                                                                                                                                                                                                                                                                                                                                                                              WITH items [21] := listbox 16
WITH items [22] := pushbutton 49
WITH items [23] := pushbutton 50
WITH items [23] := border 23
WITH items [24] := border 23
WITH items [25] := itembox 63
WITH items [26] := listbox 17
WITH items [27] := pushbutton 51
WITH fill color := 192,192,192
WITH mesus [1] := UNDETERMINED
                                                                                                                                                                                                     WITH menus [1]:= UNDETERMINED
                                                                                                                                                                                                 INSTANCE CURENT_project_selection_display ISA
                                                                                                                                                                                                display
WITH wait := FALSE
                                                                                                                                                                                                     WITH delay changes := FALSE
                                                                                                                                                                                                   WITH items [1]:= bushbutton 1
WITH items [2]:= pushbutton 1
WITH items [3]:= pushbutton 17
WITH items [4]:= pushbutton 26
WITH items [5]:= textbox 45
       WITH items [15] := textbox 27
WITH items [16] := textbox 28
                                                                                                                                                                                                                                                                                                                                                                                            INSTANCE add from general database display ISA
       WITH items [17] := textbox 29
WITH items [18] := valuebox 9
                                                                                                                                                                                                                                                                                                                                                                                           display
WITH wait := FALSE
       WITH items [19] := valuebox 10
                                                                                                                                                                                                   WITH items [6]:= pushbutton 36
WITH fill color:= 192,192,192
                                                                                                                                                                                                                                                                                                                                                                                               WITH delay changes := FALSE
WITH items [1] := UNDETERMINED
                                                                                                                                                                                                                                                                                                                                                                                              WITH items [1] = UNDETERMINED
WITH items [2] := UNDETERMINED
WITH items [3] := UNDETERMINED
WITH items [4] := UNDETERMINED
WITH items [4] := UNDETERMINED
WITH items [6] := UNDETERMINED
WITH items [7] := UNDETERMINED
WITH items [8] := UNDETERMINED
WITH items [9] := UNDETERMINED
WITH items [9] := UNDETERMINED
WITH items [10] := border 24
WITH items [11] := border 25
WITH items [12] := bistbox 18
       WITH items [20] := pushbutton 18
       WITH items [21] := pushbutton 19
WITH items [22] := pushbutton 20
                                                                                                                                                                                                     WITH menus [1] := UNDETERMINED
      WITH items [23] := pushbutton 21
WITH items [24] := listbox 6
WITH items [25] := textbox 30
                                                                                                                                                                                                INSTANCE NEW_project_selection_display ISA
                                                                                                                                                                                               display
WITH wait := FALSE
       WITH items [26] := valuebox 11
WITH items [27] := radiobutton group 3
WITH items [28] := textbox 31
                                                                                                                                                                                                     WITH delay changes := FALSE
                                                                                                                                                                                                   WITH items [1] := border I
WITH items [2] := textbox 49
WITH items [3] := pushbutton 39
WITH items [3] := pushbutton 40
WITH items [5] := pushbutton 41
WITH fill color := 192,192,192
       WITH items [29 ] := pushbutton 22
WITH items [30 ] := pushbutton 23
WITH items [31 ] := pushbutton 24
                                                                                                                                                                                                                                                                                                                                                                                                WITH items [12] := listbox 18
      WITH items [31] := pushbutton 24
WITH items [32] := textbox 32
WITH items [33] := textbox 33
WITH items [34] := promptbox 7
WITH items [36] := promptbox 69
WITH items [36] := promptbox 5
WITH items [37] := textbox 70
WITH items [38] := valuebox 8
WITH fill color := 192,192,192
                                                                                                                                                                                                                                                                                                                                                                                                WITH items [13] := valuebox 19
WITH items [14] := textbox 65
                                                                                                                                                                                                     WITH menus [1] := UNDETERMINED
                                                                                                                                                                                                                                                                                                                                                                                                WITH items [15]:= textbox 66
WITH items [16]:= listbox 19
                                                                                                                                                                                                INSTANCE NEW from another project display ISA
                                                                                                                                                                                              display
WITH wait := FALSE
                                                                                                                                                                                                                                                                                                                                                                                                 WITH items [17] := textbox 67
                                                                                                                                                                                                                                                                                                                                                                                                 WITH items [18] := textbox 68
                                                                                                                                                                                                                                                                                                                                                                                                WITH items [19] := pushbutton 53
WITH items [20] := pushbutton 54
                                                                                                                                                                                                    WITH delay changes := FALSE
WITH items [1] := border 13
                                                                                                                                                                                                    WITH items [2]:= UNDETERMINED
WITH items [2]:= tentbox 50
WITH items [4]:= tentbox 51
WITH items [5]:= listbox 12
                                                                                                                                                                                                                                                                                                                                                                                                WITH items [21] := pushbutton 55
WITH fill color := 192,192,192
       WITH menus [1]: UNDETERMINED
    INSTANCE Menus display ISA display
WITH wait := FALSE
                                                                                                                                                                                                                                                                                                                                                                                                 WITH menus [1] := UNDETERMINED
      WITH wait := FALSE
WITH delay changes := FALSE
WITH items [1] := border 12
WITH items [2] := border 11
WITH items [3] := border 10
WITH items [4] := textbox 34
WITH items [5] := listbox 8
WITH items [6] := textbox 36
WITH items [7] := listbox 9
                                                                                                                                                                                                    WITH items [5]: - IISBOOK 12
WITH items [6]: - pushbutton 42
WITH items [7]: - pushbutton 43
WITH items [8]: - pushbutton 44
WITH items [9]: - pushbutton 45
WITH fill color: - 192,192,192
                                                                                                                                                                                                                                                                                                                                                                                             INSTANCE file I ISA file
                                                                                                                                                                                                                                                                                                                                                                                                WITH filename := "project.ini"
WITH shared := FALSE
                                                                                                                                                                                                                                                                                                                                                                                                 WITH default error handling := TRUE
                                                                                                                                                                                                     WITH menus [1] := UNDETERMINED
                                                                                                                                                                                                                                                                                                                                                                                             INSTANCE file 2 ISA file
                                                                                                                                                                                                                                                                                                                                                                                                WITH filename := "project.bat" WITH shared := FALSE
                                                                                                                                                                                                 INSTANCE NEW from components display ISA
```

WITH default error handling := TRUE

INSTANCE file 3 ISA file WITH shared := FALSE WITH default error handling := TRUE

INSTANCE listbox 1 ISA listbox
WITH show current instance: "TRUE
WITH source: "building_element OF Building
element
WITH destination: "building_element OF
current_data
WITH location: "16,71,210,95

INSTANCE listbox 2 ISA fistbox
WITH show current instance: "TRUE
WITH source: "attribute_of_building_element OF
building_element_attribute
WITH destination: "attribute_of_building_element
OF current_data
WITH location: "16,171,235,240

INSTANCE listbox 3 ISA listbox
WITH show current instance: TRUE
WITH source: = building_element OF Linked
building element
WITH destination: = linked_building_element OF
current_data
WITH location: = 25,320,235,345

INSTANCE listbox 4 ISA listbox
WITH show current instance = TRUE
WITH source := stribute_of_building_element OF
Linked element attribute
WITH destination :=
linked_attribute_of_building_element OF current_data
WITH location := 25,365,235,390

INSTANCE listbox 5 ISA listbox
WITH show current instance: "TRUE
WITH source: "building_element OF Building
element
WITH destination: "building_element OF
Current_rule_data
WITH location: "265,75,489,99

INSTANCE listbox 6 ISA listbox
WITH show current instance: TRUE
WITH source: stribute of building element OF
building element stribute
WITH destination: stribute of building element
OF current rule data
WITH location: 265, 120, 489, 144

INSTANCE listbox 8 ISA listbox
WITH show current instance := TRUE
WITH source := label OF
Menus unclassified B_comp
WITH destination := current_menu_B_comp OF
Menus_manipulation_data
WITH location := 16.91,165.315

INSTANCE listbox 9 ISA listbox
WITH show current instance: TRUE
WITH source: label OF Menus_group
WITH destination: current_menu_group OF
Menus_manipulation_data
WITH location: 185,90,335,205

INSTANCE listbox 10 ISA listbox
WITH show current instance := TRUE
WITH source := label OF Menus, group items
WITH destination := current_menu_B_comp OF
Menus_manipulation_data
WITH location := 30,90,495,255

INSTANCE listbox 11 ISA listbox
WITH show current instance := TRUE
WITH source := project same OF Current_projects
WITH destination := project_uame
WITH location := 11,140,160,165

INSTANCE listbox 12 ISA listbox
WITH show current instance:= TRUE
WITH source:= project, name OF Current_projects
WITH destination:= another_project_name OF
NEW_to_make_a_new_project
WITH location:= 10,95,160,115

INSTANCE listbox 13 ISA listbox WITH show current instance:= TRUE WITH source := building_element OF Building element
WITH destination := building_element OF
NEW_to_make_a_new_project
WITH location := 35,95,460,180

INSTANCE listbox 15 ISA listbox WITH show current instance: TRUE WITH source: building_element OF Building element WITH location: = 265.200.465.215

INSTANCE listbox 16 ISA listbox
WITH show current instance: "TRUE
WITH source: "building_system OF
NEW_building_systems
WITH destination: "building_system OF
NEW_to_make_a_new_project
WITH location: "30,150,230,270

INSTANCE listbox 17 ISA listbox WITH show current instance: "TRUE WITH source: "discipline OF NEW_design_disciplines WITH destination: "design_discipline OF NEW_to_make_n_new_project WITH location: "2 65,65,460,87

INSTANCE listbox 18 ISA listbox WITH show current instance: "TRUE WITH source: "building_element OF GENERAL_Building_element WITH destination:" GENERAL_Building_component WITH location: "21,76,400,160

INSTANCE listbox 19 ISA listbox
WITH show current instance: "TRUE
WITH source: "attribute_of_building_element Of
GENERAL_building_element_attribute
WITH location: "110,205,400,220

INSTANCE menu c_Foundation ISA menu WITH label := "Foundation" WITH items [1] := UNDETERMINED

INSTANCE menu c_Floor ISA menu WITH label := "Floor" WITH items [1] := UNDETERMINED

INSTANCE menu c_Wall ISA menu WITH label := "Wall" WITH items [1] := UNDETERMINED

INSTANCE menu c_Roof ISA menu WITH label := "Roof" WITH items [1] := UNDETERMINED

INSTANCE menu c_Opening ISA menu WITH label := "Opening" WITH items [1] := UNDETERMINED

INSTANCE menu c_Protection ISA menu WITH label := "Protection" WITH items [1] := UNDETERMINED

INSTANCE menu c_Special ISA menu WITH label := "Special" WITH items [1] := UNDETERMINED

INSTANCE menu c_Finishing ISA menu WITH label := "Finishing" WITH items [1] := UNDETERMINED

INSTANCE menu c_Mechanical ISA menu WITH label := "Mechanical" WITH items [1] := UNDETERMINED

INSTANCE menu c_Electrical ISA menu WITH label := "Electrical" WITH items [1] := UNDETERMINED

INSTANCE menu c_Pier ISA menu WITH label := "Pier" WITH items [1] := UNDETERMINED INSTANCE menu c_Wood_Pier_Foundation ISA menu WITH label := "Wood_Pier_Foundation" WITH items [1]: = menu item c Circular Wood Pier Foundation

INSTANCE menu 13 ISA menu WITH label := "menu 13" WITH items [1] := menu item I

INSTANCE menu item 1 ISA menu item WITH label := "menu item 1"

INSTANCE menu item
c_Circular_Wood_Pier_Foundation ISA menu item
WITH label := "Circular_Wood_Pier_Foundation"

INSTANCE promptbox I ISA promptbox
WITH font: ""System"
WITH justify IS center
WITH frame: "TRUE
WITH show current: "TRUE
WITH strackment: "graph_name OF Building
element
WITH location: "100,100,210,125

INSTANCE promptbox 2 invisible ISA promptbox WITH pen color := 255,255,255 WITH fill color := 0,0,255 WITH fill color := 0,0,255 WITH font := "System" WITH justify IS center WITH frame := TRUE WITH show current := TRUE WITH stackment := new_component_name WITH location := 0,0,0,0

INSTANCE promptbox 3 invisible2 ISA promptbox WITH pen color := 255,255,255 WITH fill color := 0,0,255 WITH fint := "System" WITH frame := TRUE WITH show current := TRUE WITH stackment := new_ettribute_name WITH location := 0,0,0,0

INSTANCE promptbox 4 ISA promptbox
WITH pen color := 255,0,0
WITH fill color := 255,255,255
WITH fion: := "System"
WITH justify IS left
WITH frame := TRUE
WITH stroil := TRUE
WITH show current := TRUE
WITH show current := TRUE
WITH attachment := cause_of_the_effect OF
AB_general_rules
WITH location := 15,230,240,275

INSTANCE promptbox 5 ISA promptbox WITH pen color := 255,00 WITH fill color := 255,255,255 WITH font := "System" WITH justify IS center WITH frame := TRUE WITH show current := TRUE WITH attachment := percent_change OF AB_general_rules WITH location := 110,55,155,80

INSTANCE promptbox 6 ISA promptbox
WITH pen color := 255,0,0
WITH fill color := 255,255
WITH foat := "System"
WITH justify IS center
WITH frame := TRUE
WITH show current := TRUE
WITH attachment := result_in_major_modification
OF AB_general_rules
WITH location := 190,295,240,320

INSTANCE promptbox 7 ISA promptbox WITH pen color := 255,0,0 WITH fill color := 255,255,255 WITH font := "System" WITH, justify IS center WITH frame := TRUE WITH show current := TRUE WITH attachment := rule_ignored OF AB_general_rules WITH location := 190,345,240,370

INSTANCE promptbox 8 ISA promptbox WITH pen color := 128,0,0 WITH fill color := 255,255,255 WITH foot := "System" WITH justify IS center WITH frame := TRUE WITH show current := TRUE WITH attachment := new_name_of_menu_group OF ferus_manipulation_data WITH location := 185,300,335,325

INSTANCE promptbox 9 ISA promptbox WITH font := "System" WITH justify IS center WITH frame := TRUE WITH show current := TRUE WITH attachment := project_name WITH location := 10,135,160,165

INSTANCE pushbutton ! ISA pushbutton WITH label := "Building Components" WITH attribute strachment := Building Compo WITH location := 10.105,160,130

INSTANCE pushbutton 2 ISA pushbutton WITH label := "EXIT" WITH attribute attachment := exit OF application WITH location := 10,185,160,215

INSTANCE pushbutton 3 ISA pushbutton WITH label := "Delete" WITH location := 215,100,290,125

INSTANCE pushbutton 4 ISA pushbutton WITH label := "New Component" WITH location := 290,100,410,125

INSTANCE pushbutton 5 ISA pushbutton WITH label := "END" WITH attribute atlachment := nack_to_current_project_display WITH location := 425,5,500,35

INSTANCE pushbutton 6 ISA pushbutton WITH label := ** WTTH location := 425,90,500,135

INSTANCE pushbutton \$ ISA pushbutton WITH label := "Return" WITH location := 175,250,395,275

INSTANCE pushbutton 10 invisible ISA pushbutton WITH label := "Done" WITH attribute attachment := done of new component WITH location := 0,0,0,0

INSTANCE pushbutton 11 invisible ISA pushbutton WITH label := "Cancel" WITH location := 0.0.0.0

INSTANCE pushbutton 12 invisible2 ISA pushbutton WITH label := "Done" WITH attribute attackment := done of new attribute WITH location := 0,0,0,0

INSTANCE pushbutton 13 invisible2 ISA pushbutton WITH label := "Next Attribute" WITH attribute attachment := next new attribute WITH location := 0,0,0,0

INSTANCE pushbutton 14 ISA pushbutton WITH label := "Delete" WITH location := 180,250,235,270

INSTANCE pushbutton 15 ISA pushbutton WITH label := "New" WITH location := 180 270 235 290

INSTANCE pushbutton 16 invisible2 ISA pushbutton WITH label := "OK" WITH attribute attachment := ok of attribute name WITH location := 0,0,0,0

INSTANCE pushbutton 17 ISA pushbutton WITH label := "General Rules"
WITH attribute attachment := General Rules WITH location := 10,165,160,190

INSTANCE pushbutton 18 ISA pushbutton WITH label := "First"

WITH attribute attachment := action OF rules navigate 2 IS first WITH location := 400,240,490,265

INSTANCE pushbutton 19 ISA pushbutton WITH label := "Next" WITH attribute attachment := action OF pavigate IS WITH location := 400,265,490,290

INSTANCE pushbutton 20 ISA pushbutton WITH label := "Previous" WITH attribute attachment := action OF pavigate IS WITH location := 400,290,490,315

INSTANCE pushbutton 21 ISA pushbutton WITH attribute attachment := action OF navigate IS

WITH location := 400,315,490,340

INSTANCE pushbutton 22 ISA pushbutton WITH label := "END" WITH attribute attachment := back to current project display WITH location := 280,400,480,425

INSTANCE pushbutton 23 ISA pushbutton WITH label := "Update Rule" WITH attribute attachment := undate rule WITH location := 30,400,230,425

INSTANCE pushbutton 24 ISA pushbutton WITH label := "New Rule" WITH attribute attachment := new rule WITH location := 400,355,490,380

INSTANCE pushbutton 25 ISA pushbutton WITH location := 10,10,210,35

INSTANCE pushbutton 26 invisible2 ISA pushbutto WITH label := "Done" WITH attribute attachment := done of ID attribute WITH location := 0,0,0,0

INSTANCE pushbutton 26 ISA pushbutton WITH label := "Menus" WITH attribute attachment - Menus WITH location := 10,135,160,160

INSTANCE pushbutton 27 ISA pushbutton WITH label := "END" WITH attribute attackmen back_to_current_project_display WITH location := 345,360,500,415

INSTANCE pushbutton 28 ISA oushbutton WITH label := "Add to the Menu Group" WITH attribute attachment := add build comp OF Menus_manipulation_data
WITH location := 10,385,170,410

INSTANCE pushbutton 29 ISA pushbutton WITH label := "Del. Compon WITH attribute attachment := delete build comp OF Menus_manipulation_data WITH location := 350,325,495,350

INSTANCE oushbutton 30 ISA oushbutton WITH label := "Add A Menu Group" WITH attribute attachment := add menu group OF Menus_manipulation_data WITH location := 185,360,335,385

INSTANCE pushbutton 31 ISA pushbutton WITH label := "Delete Menu Group" WITH attribute attachment := delete menu group OF Menus manipulation data WITH location := 185,260,335,285

INSTANCE pushbutton 32 ISA pushbutton WITH label := "Rename Menu Group" WITH attribute attachment := rename menu group OF nus manipulation data WITH location := 185,385,335,410

INSTANCE pushbutton 33 ISA pushbutton WITH label := "EXIT" WITH attribute attachment := exit OF application WITH location := 10,195,160,220

INSTANCE pushbutton 34 ISA pushbutton WITH label := "New Project" WITH attribute attachment := New_project WITH display attachment := NEW_project_start_display WITH location := 10,105,160,135

INSTANCE pushbutton 35 ISA pushbutton WITH label := "Existing Project" WITH attribute attachment := Current_project WITH display attachment := CURENT_project_start_display WITH location := 10,145,160,175

INSTANCE pushbutton 36 ISA pushbutton WITH label := "EXIT" WITH attribute attache ent := exit OF application WITH location := 10,195,160,220

INSTANCE pushbutton 37 ISA pushbutton WITH label := "OK" WITH location := 10,170,160,195

INSTANCE pushbutton 38 ISA pushbutton WITH label := "OK" WITH location := 10,170,160,195

INSTANCE pushbutton 39 ISA pushbutton WITH label := "from Another Project" WITH display attachment := NEW from another project display WITH location := 10,105,160,140

INSTANCE pushbutton 40 ISA pushbutton WITH label := "from Systems WITH attribute attachment := from systems OF NEW_to_make_s_new_project WITH location := 10,145,160,180

INSTANCE pushbutton 41 ISA pushbutton WITH label := "from Components" WITH attribute attachment := from components OF NEW to make a new project WITH location := 10,185,160,220

INSTANCE pushbutton 42 ISA pushbutton WITH label := "Modify components" WITH location := 10,120,160,145

INSTANCE pushbutton 43 ISA pushbutton WITH label := "Modify Gen. Rules" WITH location := 10,145,160,170

INSTANCE pushbutton 44 ISA pushbutton WITH label := "Config. Menus" WITH location := 10,170,160,195

INSTANCE pushbutton 45 ISA pushbutton WITH label := "Back" WITH display attachment := NEW project selection display WITH location := 10,195,160,220

INSTANCE pushbutton 46 ISA pushbutton WITH label := "OK"
WITH attribute attachment := from systems OF NEW_to_make_a_new_project WITH location := 20,195,160,220

INSTANCE pushbutton 47 ISA pushbutton WITH label := "Add selected component to the new WITH attribute attachment := add component to the

new project OF NEW_to_make_s_new_project WITH location := 20,355,335,380

INSTANCE pushbutton 48 ISA pushbutton WITH label := "Finish" WITH attribute attachment := finish components of he new project OF NEW_to_make_a_new_project WITH location := 345,355,475,380

INSTANCE pushbutton 49 ISA pushbutton WITH label := "Add to the new project" WITH attribute attachment := add system to the new roject OF NEW_to_make_a_new_project WITH location := 255,330,475,355

INSTANCE pushbutton 50 ISA pushbutton WITH label := "Finish"

WITH attribute attachment: finish systems of the new project OF NEW_to_make_a_new_project WITH location: 255,360,475,385 WITH font := "System"
WITH text := "Current Type" INSTANCE textbox 2 ISA textbox WITH location := 355,225,445,250 WITH pen color := 255,255,255 WITH fill color := 128,0,0 INSTANCE pushbutton 51 ISA pushbutton WITH label: "Ignore highlighted compo INSTANCE textbox 13 ISA textbox WITH justify IS center WITH attribute attachment := ignore highlighted component OF NEW_to_make a new project WITH font := "Times New Roman"
WITH font style IS bold, italic CF FALSE, underline WITH location := 265,285,465,310 CF FALSE, strikeout CF FALSE WITH foot size := 12
WITH frame := TRUE
WITH text := "Configuration of Building Components INSTANCE pushbutton 52 ISA pushbutton WITH label := " " Discipline^e WITH location := 355,250,425,290 WITH attribute attachment := get from general and Attributes" INSTANCE textbox 14 ISA textbox WITH location := 5,5,420,35 WITH pen color := 0,0,0 WITH fill color := 192,192,192 WITH location := 425,40,500,85 INSTANCE textbox 3 ISA textbox INSTANCE pushbutton 53 ISA pushbutton WITH label := "Add the Component" WITH pen color := 0,0,255 WITH fill color := 192,192,192 WITH justify IS center WITH font := "System"
WITH text := "Review WITH justify IS left
WITH font := "System"
WITH text := "List of Building Components" WITH attribute attachment := add component from get from general datab
WITH location := 20,285,215,305 Attributes WITH location := 430,95,495,130 WITH location := 15,50,205,70 INSTANCE textbox 15 ISA textbox
WITH pen color := 0,0,255
WITH fill color := 192,192,192
WITH justify IS left
WITH fout style IS bold, italic CF FALSE, underline
CF FALSE, strikcout CF FALSE INSTANCE pushbutton 54 ISA pushbutton WITH label: = "Return" INSTANCE textbox 4 ISA textbox WITH attribute attachment := return from _get from WITH pen color := 0,0,255 WITH fill color := 192,192,192 WITH location := 215,285,400,305 WITH justify IS left WITH foot := "System"
WITH text := "Current Building Components" INSTANCE pushbutton 55 ISA pushbutton WITH font size := 12
WITH text := "Summary of Attributes of the Building WITH label := "Add the Component and its related WITH location := 215,50,410,70 General Rules nent := add component and WITH attribute attache INSTANCE textbox 5 ISA textbox NRSIANCE tention > 15A tention with the color: = 0,0,255
WITH fill color: = 192,192,192
WITH justify IS left
WITH font: = "System"
WITH text: = "Building Component
WITH location: = 15,150,220,170 WITH location := 10,10,355,35 WITH location := 20,305,400,325 INSTANCE textbox 16 ISA textbox
WITH pen color:= 128,0,0
WITH fill color:= 192,192,192
WITH justify IS left
WITH font:= "Times New Roman"
WITH font syle IS bold, italic, underline CF FALSE,
strikeout CF FALSE INSTANCE radiobutton group 1 ISA radiobutton ent Attributes WITH pen color := 0,0,0 WITH fill color := 192,192,192
WITH fill color := 192,192,192
WITH frame := TRUE
WITH group label := **
WITH show current := TRUE INSTANCE textbox 6 ISA textbox WITH pen color := 0,0,255 WITH fill color := 192,192,192 WITH foot size := 12 WITH text := ** WITH attachment := discipline OF input data WITH location := 235,145,355,295 WITH justify IS left
WITH font := "System"
WITH text := "Current Attribute" WITH location := 355,10,590,35 INSTANCE radiobutton group 2 ISA radiobutton WITH location := 15,240,130,260 INSTANCE textbox 17 invisible ISA textbox WITH pen color := 0,0,0 WITH fill color := 192,192,192 WITH pen color := 0,0,0 INSTANCE textbox 7 ISA textbox WITH fill color := 192,192,192 WITH frame := TRUE WITH justify IS left WITH font := "System WITH pen color := 0,0,255 WITH fill color := 192,192,192 WITH group label := "attribute type" WITH show current := TRUE WITH justify IS left WITH text := "Type the name of WITH font := "System"
WITH text := "Select Linked Component" building component"
WITH location := 0,0,0,0 WITH attachment := attribute type OF input data WITH location := 355,145,495,225 WITH location := 25,300,225,320 INSTANCE textbox 18 invisible2 ISA textbox INSTANCE radiobutton group 3 ISA radiobutton INSTANCE textbox \$ ISA textbox WITH pen color := 0,0,0 WITH fill color := 192,192,192 WITH pen color := 0,0,255 group
WITH pen color := 0.0.0 WITH fill color := 192,192,192 WITH justify IS left WITH justify IS left WITH font := "System" WITH fill color := 192,192,192 WITH font := "System"
WITH text := "Graph name"
WITH location := 15,105,100,125 WITH frame := TRUE WITH group label := " WITH text := "Type the attribute name of the building component and press OK* WITH location := 0.0.0.0 WITH show current := TRUE WITH attachment := discipline OF input data WITH location := 260,220,395,385 INSTANCE textbox 9 ISA textbox INSTANCE textbox 19 invisible2 ISA textbox WITH pen color := 0,0,255
WITH fill color := 192,192,192
WITH fill color := 192,192,192
WITH font := "System"
WITH text := "Select Linked Attribute"
WITH text := "Select Linked Attribute" WITH pen color := 0.0.0 INSTANCE table 1 ISA table
WITH attachment := building_element_attribute WITH fill color := 192,192,192 WITH justify IS left
WITH font := "System"
WITH text := "Fill the rest of the attribute data WITH columns [1]:= column 1 WITH columns [2] := column 2 WITH columns [3] := column 3 (discipline, type)"
WITH location := 0,0,0,0 WITH location := 25,345,185,365 WITH columns [4]:= column WITH columns [5] := column 5 INSTANCE textbox 10 ISA textbox WITH heading := TRUE INSTANCE textbox 20 ISA textbox WITH pen color := 255,255,255 WITH fill color := 128,0,0 WITH pen color := 0,0,255 WITH fill color := 192,192,192 WITH heading height := 50 WITH fill color := 0,255,255 WITH justify IS left WITH column lines := TRUE WITH font := "System"
WITH text := "Current Linked Component" WITH justify IS center
WITH font := "Times New Roman" WITH row lines := TRUE WITH frame := TRUE WITH location := 260,300,460,320 WITH font syle IS bold, italic CF FALSE, underline WITH show current instance := TRUE CF FALSE, strikeout CF FALSE WITH location := 10,45,590,240 INSTANCE textbox 11 ISA textbox WITH foot size := 18 WITH pen color := 0,0,255 WITH fill color := 192,192,192 WITH frame := TRUE WITH text := "GENERAL RULE INSTANCE textbox I ISA textbox WITH pen color := 255,255,255 WITH fill color := 0.0.128 WITH justify IS left CONFIGURATION* WITH font := "System"
WITH text := "Current Linked Attribute" WITH location := 5,5,500,45 WITH justify IS center WITH font := "Times New Roman"
WITH font style IS bold, italic, underline CF FALSE. WITH location := 260,345,420,365 INSTANCE textbox 23 ISA textbox WITH pen color := 0,0,0 strikeout CF FALSE INSTANCE textbox 12 ISA textbox WITH fill color := 192,192,192 WITH pen color := 0,0,255
WITH fill color := 192,192,192
WITH justify IS left WITH font size := 12
WITH text := "CONFIGURATION WITH justify IS left WITH font := "System"
WITH text := "then the discipline" MODULE*

WITH location := 10,30,155,80

WITH location := 15,170,140,190 WITH fill color := 192,192,192 WITH location := 15,25,145,75 WITH justify IS center
WITH font := "System"
WITH text := "Unclassified Building Components" INSTANCE textbox 24 ISA textbox INSTANCE textbox 46 ISA textbox WITH pen color := 0,0,0 WITH fill color := 192,192,192 WITH justify IS left WITH font := "System" WITH text := "textbox 46" WITH location := 15,55,160,90 WITH justify IS left WITH font := "System"
WITH text := "If such a change results in major INSTANCE textbox 37 ISA textbox WITH location := 210,65,360,140 WITH pen color := 0,0,0 WITH fill color := 192,192,192 redesign, please type YES" WITH location := 10,285,190,335 WITH justify IS center WITH foot := "System"
WITH text := "List of Menu Groups" INSTANCE textbox 26 ISA textbox WITH pen color := 0,0,0 WITH fill color := 192,192,192 WITH justify IS left WITH location := 200,55,320,90 INSTANCE textbox 38 ISA textbox WITH fone := "System"
WITH text := "Select a Building Component" WITH pen color := 0,0,0 WITH fill color := 192,192,192 WITH location := 270,60,465,80 WITH justify IS center WITH foot := "System"
WITH text := "Components of Current Menu Group"
WITH location := 350,50,490,85 INSTANCE textbox 27 ISA textbox WITH pen color := 0,0,0 WITH fill color := 192,192,192 WITH justify IS left
WITH font := "System"
WITH text := "Select a Component with location := 270,105,465,125 INSTANCE textbox 39 ISA textbox WITH pen color := 0,0,255 WITH fill color := 192,192,192 WITH justify IS center WITH font := "System" WITH text := "Selected Menu Group" INSTANCE textbox 28 ISA textb WITH pen color := 0,0,255 WITH fill color := 192,192,192 WITH location := 185,210,335,230 WITH justify IS left INSTANCE textbox 40 ISA text WITH font := "System"
WITH text := "# of Current Related Rules:" WITH pen color := 0,0,255 WITH fill color := 192,192,192 WITH location := 275,155,455,175 WITH justify IS center WITH font := "System"
WITH text := "Selected Building Component" INSTANCE textbox 29 ISA textbox WITH pen color := 0,0,255 WITH fill color := 192,192,192 WITH justify IS left WITH location := 15,320,165,355 INSTANCE temp invisible textbox 41 ISA textbox WITH font := "System"
WITH text := "Current Related Rules #:" WITH pen color := 0,0,255 WITH fill color := 192,192,192 WITH justify IS center
WITH foot := "System"
WITH text := "Write the new name" WITH location := 275,180,445,200 INSTANCE textbox 30 ISA textbox WITH pen color := 0,0,0 WITH fill color := 192,192,192 WITH location := 0,0,0,0 WITH justify IS left
WITH foot: = "System"
WITH text: = "of the following Building Component"
WITH text: = "of the following Building Component"
WITH location := 10,125,245,145 INSTANCE textbox 41 ISA textbox WITH pen color := 0,0,255 WITH fill color := 192,192,192 WITH justify IS center
WITH font: = "System"
WITH text: = "Type Added or Renamed Name above"
WITH location: = 185,125,335,360 INSTANCE textbox 31 ISA textbox WITH pen color := 0,0,255 WITH fill color := 192,192,192 WITH justify IS left INSTANCE textbox 42 ISA textbox WITH foot := "System"
WITH text := "Related General Rule" WITH pen color := 0,0,255 WITH fill color := 192,192,192 WITH location := 405,205,495,240 WITH justify IS center WITH font := "System"
WITH text := "Selected Building Component" INSTANCE textbox 32 ISA textbox WITH pen color := 0,0,0 WITH fill color := 192,192,192 WITH location := 355,260,490,295 WITH justify IS left INSTANCE textbox 43 ISA textbox WITH font := "System"
WITH text := "Affected Discipline" WITH pen color := 0,0,0 WITH fill color := 192,192,192 WITH location := 265,205,390,225 WITH justify IS center WITH font := "System" INSTANCE textbox 33 ISA textbox WITH text := "Double click to select a project then WITH pen color := 0,0,0 WITH fill color := 192,192,192 WITH location := 10,100,165,140 WITH justify IS left WITH font := "System"
WITH text := "If you need this rule to be ignored, INSTANCE textbox 44 ISA textbox WITH pen color := 255,255,255 WITH fill color := 0.0,128 olease type YES" WITH location := 10,340,180,375 WITH justify IS center WITH font := "Times New Roman" INSTANCE textbox 34 ISA textbox WITH font style IS bold, italic CF FALSE, underline WITH pen color := 255,255,255 WITH fill color := 128,0,0 CF FALSE, strikeout CF FALSE WITH font size := 16
WITH text := "Configure Existing Project" WITH justify IS center
WITH font := "Times New Roman"
WITH font style IS bold, italic CF FALSE, underline
CF FALSE, strikeout CF FALSE WITH location := 15,15,150,85 INSTANCE textbox 45 ISA textbox WITH pen color := 255,255,255 WITH fill color := 0,0,128 WITH font size := 16

WITH justify IS center
WITH font := "Times New Roman"
WITH font style IS bold, italic CF FALSE, underline

WITH font size := 12
WITH text := "What would you like to configure?"

CF FALSE, Strikeout CF FALSE

WITH frame := TRUE

WITH location := 5,5,500,40

WITH pen color := 0,0,0

INSTANCE textbox 36 ISA textbox

WITH text := "Configuration of Menu of Building omnoments"

INSTANCE textbox 47 ISA textbox WITH pen color := 255,255,255 WITH fill color := 0,0,128 WITH justify IS center WITH font := "Times New Roman"
WITH font style IS bold, italic CF FALSE, underline CF FALSE, strikeout CF FALSE WITH font size := 16
WITH text := "Assemble New Project" WITH location := 10,20,155,75 INSTANCE textbox 48 ISA textbox WITH pen color := 0,0,0 WITH fill color := 192,192,192 WITH justify IS center
WITH font: = "System"
WITH text: = "Type the project name then click OK"
WITH location:= 10,100,165,135 INSTANCE textbox 49 ISA textbox WITH pen color := 255,255,255 WITH fill color := 0,0,128 WITH justify IS center
WITH font := "Times New Roman"
WITH font style IS bold, italic CF FALSE, underline CF FALSE, strikeout CF FALSE WITH font size := 14 WITH text := "Select an Assembly Method" WITH location := 20,15,145,80 INSTANCE textbox 50 ISA textbox WITH pen color := 255,255,255 WITH fill color := 0,0,128 WITH justify IS cente WITH font := "Times New Roman"
WITH font style IS bold, italic CF FALSE, underline
CF FALSE, strikeout CF FALSE WITH foat size := 12 WITH text := "Assemble from another project" WITH location := 25,15,140,60 INSTANCE textbox 51 ISA textbox WITH pen color := 0,0,0 WITH fill color := 192,192,192 WITH justify IS left
WITH foot: = "System"
WITH text: = "Select project"
WITH location: = 40,75,140,95 INSTANCE textbox 52 ISA textbox WITH pen color := 255,255,255 WITH fill color := 0,0,128 WITH justify IS center
WITH font := "Times New Roman"
WITH font style IS bold, italic CF FALSE, underline
CF FALSE, strikeout CF FALSE WITH font size := 12 WITH text := "Select Participating Disciplines"
WITH location := 15.10.155.55 INSTANCE textbox 53 ISA textbox WITH pen color := 255,255,255 WITH fill color := 128,0,0 WITH justify IS center WITH font: "Times New Roman"
WITH font style IS bold, italic CF FALSE, underline
CF FALSE, strikeout CF FALSE WITH foot size := 14 WITH frame := TRUE WITH text := "Assemble the New Project from the WITH location := 5,5,490,35 INSTANCE textbox 54 ISA textbox WITH pen color := 0,0,0 WITH fill color := 192,192,192 WITH justify IS center
WITH font := "System"
WITH text := "Double click to select a building WITH location := 105.65.395.85 INSTANCE textbox 55 ISA textbox

WITH pen color := 0,0,0 WITH fill color := 192,192,192 WITH fill color := 192,192,192 WITH fill color := 192,192,192 WITH justify IS left WITH justify IS center WITH foat := "System
WITH frame := TRUE WITH justify IS left WITH fost := "System"
WITH text := "Attributes WITH font := "System"
WITH text := "Selected Componen
WITH location := 30,215,110,250 WITH clipped := TRUE
WITH attachment := designing_discipline OF of the Component"
WITH location := 20,215,100,270 surrent_data WITH location := 435,260,490,285 INSTANCE textbox 57 ISA textbox WITH pen color := 0,0,0 WITH fill color := 192,192,192 WITH justify IS left WITH form = State = 1 INSTANCE textbox 67 ISA textbox WITH pen color := 255,255,255 WITH fill color := 128,0,0 INSTANCE valuebox 6 ISA valuebox WITH justify IS center
WITH font: "Times New Roman"
WITH font style IS bold, italic CF FALSE, underline
CF FALSE, strikeout CF FALSE WITH pen color := 128,0,0 WITH fill color := 192,192,192 WITH font := "System"
WITH text := "Attributes WITH justify IS center WITH foat := "System" WITH frame := TRUE of the Component*
WITH location := 35,270,115,325 WITH font size := 14 WITH fract: "TRUE
WITH text: ""Components in the General Database"
WITH location: "5,5,415,35 WITH clipped := TRUE WITH attachment := attribute_type OF current_data WITH location := 465,225,490,250 INSTANCE textbox 58 ISA textbox WITH pen color := 255,255,255 WITH fill color := 128,0,0 WITH justify IS center WITH font := "Times New Roman" INSTANCE textbox 68 ISA textbox INSTANCE valuebox 7 ISA val WITH pen color := 0,0,0 WITH fill color := 192,192,192 WITH pen color := 0,0,255 WITH fill color := 192,192,192 WITH font style IS bold, italic CF FALSE, underline CF FALSE, strikeout CF FALSE WITH justify IS center WITH font := "System" WITH frame := TRUE WITH justify IS left WITH first: ""System"
WITH text: ""Double Click to Select a Component"
WITH location := 90,55,330,75 WITH font size := 14 WITH frame := TRUE WITH text := "Assemble the New Project from Building Systems" WITH clipped := TRUE
WITH attachment := building_element OF WITH location := 5,5,490,35 INSTANCE textbox 69 ISA textbox rrent_role_data WITH pen color := 0,0,0 WITH fill color := 192,192,192 WITH location := 15.145,240,165 **INSTANCE** textbox 59 ISA textbox WITH justify IS left
WITH font := "System"
WITH text := "If a change of WITH pen color := 0,0,0 WITH fill color := 192,192,192 INSTANCE valuebox \$ ISA va WITH pen color := 0,0,255 WITH fill color := 192,192,192 WITH justify IS center
WITH font:= "System"
WITH text:= "Double click to select a building % occurs o the following attribute"
WITH location := 15,65,245,100 WITH justify IS cente WITH frame := "System"
WITH frame := TRUE
WITH clipped := TRUE WITH location := 50,115,215,150 INSTANCE textbox 70 ISA textbox WITH pen color := 0,0,0 WITH fill color := 192,192,192 WITH attachment := attribute of building elem INSTANCE textbox 60 ISA textbox OF Current rule data WITH location := 15,100,240,120 WITH justify IS left
WITH font := "System"
WITH text := "need to be notified because it may WITH pen color := 0,0,0 WITH fill color := 192,192,192 WITH justify IS left
WITH font := "System"
WITH text := "Selected Building System" INSTANCE valuebox 9 ISA v WITH pen color := 0,0,255 WITH fill color := 192,192,192 WITH location := 15,193,245,230 WITH justify IS left WITH font := "System" WITH frame := TRUE WITH location := 280,115,450,135 INSTANCE valuebox 1 ISA valuebox WITH pen color := 128,0,0 WITH fill color := 192,192,192 INSTANCE textbox 61 ISA textbox WITH pen color := 0,0,0
WITH fill color := 192,192,192
WITH justify IS center WITH clipped := TRUE
WITH attachment := mumber instances OF rules WITH justify IS center WITH foot := "System" WITH font := "System"
WITH text := "Components in the building system"
WITH location := 275,165,460,202 WITH frame := TRUE WITH location := 455,155,490,175 WITH clipped := TRUE
WITH attachment := building element OF INSTANCE valuebox 10 ISA valuebox WITH pen color := 90,255
WITH fill color := 192,192,192
WITH justify 1S left.
WITH font := "System"
WITH frame := TRUE INSTANCE textbox 62 ISA textbox WITH pen color := 0,0,0 WITH fill color := 192,192,192 WITH location := 215.70.410.95 INSTANCE valuebox 2 ISA valuebox WITH justify IS center WITH pen color := 128,0,0 WITH fill color := 192,192,192 WITH font := "System" WITH clipped := TRUE WITH justify IS center WITH font := "System" WITH frame := TRUE WITH text := "Description of the highlighted building WITH attachment := instance number OF rules system" = 40,275,225,312 navigate 2 WITH location := 455,180,490,200 WITH dipped := TRUE INSTANCE valuebox 11 ISA valuebox **INSTANCE** textbox 63 ISA textbox WITH attachment := attribute_of_building_element WITH pen color := 255,255,255 WITH fill color := 0,128,128 WITH pen color := 0,0,255 WITH fill color := 192,192,192 OF corrent data WITH location := 15,260,170,285 WITH justify IS center
WITH font:= "System"
WITH text:= "Double click to select a design WITH justify IS center WITH font := "System" WITH frame := TRUE INSTANCE valuebox 3 ISA valuebox WITH pen color = 128,0.0
WITH fill color := 192,192,192
WITH justify IS center
WITH font := "System"
WITH frame := TRUE
WITH dipped := TRUE WITH clipped := TRUE
WITH attachment := affected_discipline OF WITH location := 30,60,255,95 AB_general_rules WITH location := 155,170,240,190 INSTANCE textbox 64 ISA textbox WITH pen color := 0,0,0 WITH fill color := 192,192,192 WITH attachment := linked_building_element OF INSTANCE valuebox 13 ISA WITH justify IS center WITH font := "System" WITH text := "From WITH pen color := 128,0,0 WITH fill color := 192,192,192 WITH location := 260,320,480,345 WITH justify IS center INSTANCE valuebox 4 ISA valuebox WITH font := "System" WITH frame := TRUE WITH location := 430,45,495,80 WITH pen color := 128,0,0 WITH fill color := 192,192,192 WITH dipped := TRUE WITH attachment := current_menu_B_comp OF Menus_manipulation_data INSTANCE textbox 65 ISA textbox WITH justify IS center WITH font := "System" WITH frame := TRUE WITH pen color := 0,0,0 WITH fill color := 192,192,192 WITH location := 10,355,170,380 WITH justify IS left WITH clipped := TRUE WITH font := "System"
WITH text := "Selected Commonent" WITH attachment := INSTANCE valuebox 14 ISA valuebox linked attribute_of_building_element OF current_data WITH location := 260,365,480,390 WITH pen color := 128,0,0 WITH fill color := 192,192,192 WITH location := 20,170,100,205 WITH justify IS center INSTANCE textbox 66 ISA textbox WITH foot := "System"
WITH frame := TRUE INSTANCE valuebox 5 ISA valuebox

WITH pen color := 128,0,0

WITH pen color := 0.0.0

WITH clipped := TRUE WITH stachment := current_menu_B_comp OF Menus_manipulation_data WITH location := 350,295,495,320

INSTANCE valuebox 15 ISA valuebox
WITH pen color := 128,0,0
WITH fill color := 192,192,192
WITH justify IS center
WITH font := "System"
WITH frame := TRUE
WITH disped := "TRUE
WITH attachment := current menu_group OF
Menus_manipulation_data
WITH location := 185,230,335,255

INSTANCE valuebox 16 ISA valuebox
WITH pen color: = 0,025
WITH fill color: = 192,192
WITH fill color: = 192,192
WITH fint: = TRUE
WITH finne: = TRUE
WITH clipped: = TRUE
WITH strackmest: = building_element OF
NEW_to_make_a_new_project.
WITH location: = 115,215,460,245

INSTANCE valuebox 17 ISA valuebox WITH pen color := 0,255 WITH fill color := 192,192,192 WITH justify IS center WITH foot := "System" WITH frame := TRUE WITH clipped := TRUE WITH attachment := building_system OF NEW_to_make_a_new_project WITH location := 265,135,465,160

INSTANCE valuebox 18 ISA valuebox WITH pen color: = 0,0,255 WITH fill color: = 192,192,192 WITH justify IS left: WITH foat: = "System" WITH frame: = TRUE WITH clipped: = "TRUE WITH strackment: = description OF NEW_building_systems WITH location: = 35,315,230,370

INSTANCE valuebox 19 ISA valuebox WITH pen color := 0,0,255 WITH pll color := 192,192,192 WITH justify IS center WITH font := "System" WITH frame := TRUE WITH clipped := TRUE WITH stackment := GENERAL Building_component WITH location := 110,170,400,200

INSTANCE main window ISA window WITH location := 207,160,381,413 WITH menus [1] := UNDETERMINED

WITH style IS moveable, sizzable, closeable WITH title := *** WITH visible := TRUE

INSTANCE configuration window ISA window WITH location: 61,11,578,464 WITH menus [1]: UNDETERMINED WITH style IS moveable, sizeable, closeable CF FALSE WITH title: = FALSE

INSTANCE attribute window ISA window WITH location := 22,62,627,371 WITH menus [1] := UNDETERMINED WITH style IS moveable, sizeable, closeable WITH title := ""
WITH visible := FALSE

INSTANCE get from general database window ISA window
WITH location := 192,114,617,490
WITH menus [1] := UNDETERMINED
WITH style IS moveable, sizeable, closeable
WITH title := ""
WITH visible := FALSE

! DEMON GROUP: affected_discipline OF AB_general_rules

DEMON 11
IF CONF(affected_discipline OF AB_general_rules)
>= 0
THEN FORGET discipline OF input data

! DEMON GROUP: button selected OF message box

DEMON 5
IF button selected OF message box 1 IS yes
THEN delete a component := TRUE

DEMON 10
IF button selected OF message box 3 IS yes
THEN delete of attribute: "TRUE

! DEMON GROUP: selected OF pushbutton

DEMON!
IF selected OF pushbutton 6
THEN text OF textbox 16:= building_element OF
current_data
AND output OF attribute window:= attributes table
display
AND visible OF attribute window:= TRUE

DEMON 2
IF selected OF pushbutton 8
THEN visible OF attribute window:= FALSE

DEMON 4
IF selected OF pushbutton 3
THEN put up OF message box 1 := TRUE

DEMON 6 IF selected OF pushbutton 4 THEN location OF border 6 invisible := 15,50,410,95 AND location OF textbox 17 invisible := 25,55,160,90 AND location OF promptbox 2 invisible :** AND location OF pushbutton 10 invisible: = 355,50,410,70 AND location OF pushbutton 11 invisible := 355,75,410.95 AND location OF pushbutton 12 invisible2 := 0,0,0,0 AND location OF promptbox 3 invisible2 := 0,0,0,0 AND location OF border 7 invisible2 := 0,0,0,0 AND location OF textbox 18 invisible2 := 0,0,0,0 AND location OF tembox 19 invisible2 := 0,0,0,0 AND location OF pushbutton 13 invisible2 := 0,0,0,0 AND FORGET current data AND FORGET building_element_attribute **DEMON 7** IF selected OF pushbutton 11 invisible THEN location OF border 6 invisible := 0,0,0,0
AND location OF textbox 17 invisible := 0,0,0,0

AND location OF prompthox 2 invisible: = 0,0,0,0
AND location OF pushbutton 10 invisible: = 0,0,0,0
AND location OF pushbutton 11 invisible: = 0,0,0,0
DEMON 8
If selected OF pushbutton 15
THEN location OF pushbutton 16 invisible2:=
180,195,220,220
AND location OF prompthox 3 invisible2:=
25,195,220,220
AND location OF border 7 invisible2:=
15,150,235,290
AND location OF textbox 18 invisible2:=
25,155,220,190
AND location OF textbox 19 invisible2:=
25,155,220,260
AND FORGET stribute_of_building_element OF current_data
AND FORGET stribute_type OF current_data
AND FORGET inked_building_element OF current_data
AND FORGET linked_building_element OF current_data
AND FORGET linked_attribute_of_building_element
AND FORGET linked_attribute_of_building_element
AND FORGET linked_attribute_of_building_element

OF current_data

DEMON 9
IF selected OF pushbutton 14

THEN put up OF message box 3 := TRUE ! DEMON GROUP: double clicked OF table

DEMON 3
IF double clicked OF table 1
THEN visible OF antibute window := FALSE
AND antibute_of_building_element OF current_data
:= attribute_of_building_element OF
building_element_attribute

END

C.2. DATA MANIPULATION MODULE

```
WITH cause of the effect STRING
WITH percent_change NUMERIC
WITH rule_ignored STRING
 $VERSION35
                                                                                                 ste_of_building_element STRING
                                                                                 WHEN CHANGED
 SLOCATIONS ARE PIXELS
                                                                                   BEGIN
 CLASS AB general rules
WITH AB general rules ID NUMERIC
WITH building element STRING
WITH attribute of building element STI
WITH affected discipline STRING
                                                                              IF ( CONF(attribute_of_building_element OF current_data) >= 0) THEN
                                                                                                                                                            CLASS edit table INHERITS add on, display item
                                                                                                                                                             WITH attachment CLASS REFERENCE
WITH columns INSTANCE REFERENCE
                                                                                      BEGIN
                                                                                        attribute_type OF current_data := attribute_type
                                                                                                                                                               COLLECTION
                                                                              OF building_element_attribute
designing_discipline OF current_data :=
                                                                                                                                                             WITH heading SIMPLE
INIT TRUE
  WITH percent change NUMERIC
  WITH result in major modification STRING
WITH cause of the effect STRING
WITH rule ignored STRING
                                                                                       ing_discipline OF building_element_attribute
linked_building_element OF current_data :=
                                                                                                                                                             WITH heading height NUMERIC
WITH row height NUMERIC
WITH fill color COLOR
                                                                              linked_building_element OF
building_element_attribute
                                                                                                                                                              WITH row separator SIMPLE INIT TRUE
                                                                              linked_attribute_of_building_element OF
current_data := linked_attribute_of_building_element
CLASS border INHERITS add on, display item
  WITH style COMPOUND
                                                                                                                                                              WITH column separator SIMPLE
    picture frame,
                                                                              OF building_element_attribute
                                                                                       FORGET attribute type OF input data
FORGET discipline OF input data
                                                                                                                                                              INIT TRUE
    edit control.
                                                                                                                                                              WITH max rows NUMERIC
    group,
                                                                                                                                                              WITH total rows NUMERIC
                                                                                      END
                                                                                                                                                             WITH current cell NUMERIC ARRAY SIZE 2
  INIT picture frame
WITH perspective COMPOUND
                                                                                   END
                                                                                WITH attribute_type STRING
WITH designing_discipline STRING
WITH linked building_element STRING
                                                                                                                                                              INIT[1]1
INIT[2]1
                                                                                                                                                             WITH current cell data STRING
WITH highlight text color COLOR
                                                                                  WHEN CHANGED
    INIT in
  WITH border width NUMERIC
                                                                                   BEGIN
                                                                                                                                                             WITH highlight fill color COLOR
WITH display only SIMPLE
WITH selected SIMPLE
                                                                                    IF ( CONF(linked_building_element OF
    INIT I
                                                                              current_data) >= 0) THEN
  WITH background color COLOR.
    INIT 192, 192, 192
                                                                                      BEGIN
                                                                                        EXEC | Sodbe | SOL SELECT * from [AB-
                                                                                                                                                              WITH double clicked SIMPLE
  WITH 611 color COLOR
                                                                              attribute-designer] where building_element = :current_data_linked_building_element
END SQL INTO Linked element
   INIT 192,192,192
  AVAL 192,192
WITH highlight color COLOR
INIT 255,255,255
WITH shadow color COLOR
INIT 128,128,128
                                                                                                                                                           CLASS edit table column
                                                                                                                                                             LIASS edit table column
WITH stackment ATTRIBUTE REFERENCE
WITH heading shel STRING
WITH heading style INSTANCE REFERENCE
WITH column style INSTANCE REFERENCE
WITH width NUMERIC
                                                                                        FIND Linked building eleme
CLASS Building element
WITH building_element STRING
WITH graph_name STRING
WHEN CHANGED
                                                                             WHERE building element OF Linked
building element = linked_building_element OF
                                                                                                                                                             WITH max characters NUMERIC
                                                                                                                                                             WITH formst STRING
WITH justify COMPOUND
                                                                              current_data
                                                                                         FIND END
                                                                                                                                                              ieft,
center,
     BEGIN
graph_name OF current_data := graph_name OF
Building element
                                                                                 linked_building_element OF
nilding_element_attribute := linked_building_element
                                                                                                                                                              right
INIT left
    END
                                                                              OF curr
                                                                                         nt data
                                                                                                                                                             WITH per color COLOR
COLLECTION
                                                                                     FND
CLASS building_element_attribute
                                                                                                                                                              WITH fill color COLOR
  WITH anribute of building element STRING
                                                                                   END
  WITH attribute_type STRING
WITH designing_discipline STRING
WITH designing_discipline STRING
WITH linked_building_element STRING
WITH linked_attribute_of_building_element STRING
                                                                                                                                                               COLLECTION
                                                                                WITH linked_attribute_of_building_element STRING
                                                                                 WHEN CHANGED
                                                                                                                                                           CLASS GENERAL Building element
                                                                                   REGIN
                                                                                                                                                             WITH building element STRING
WITH graph_name STRING
                                                                                    IF ( CONF(linked_attribute_of_building_element
                                                                              OF current_data) >= 0) THEN
BEGIN
CLASS current_data
WITH building_element STRING
WHEN CHANGED
                                                                                        linked_attribute_of_building_element OF
                                                                                                                                                           CLASS GENERAL_building_element_attribute
                                                                                                                                                             WITH attribute of building element STRING WITH attribute type STRING
                                                                                uilding_element_attribute := linked_building_element
     REGIN
                                                                              OF current data
                                                                                                                                                             WITH designing discipline STRING
WITH linked building element STRING
WITH linked_attribute_of_building_element STRING
       EXEC ISodbe | SQL SELECT * from [AB-
                                                                                   END
 attribute-designer] where building_element
                                                                                WITH graph_name STRING
:current_data_building_element
END SQL INTO
                                                                                  .ASS Current_projects
                                                                                                                                                           CLASS input data
WITH discipline COMPOUND
building_element_attribute
                                                                                WITH project_name STRING
       FOR (i := 1 TO number instances OF navigate
                                                                                CLASS Current_rule_data
WITH building_element STRING
WHEN CHANGED
                                                                                                                                                               Structure,
                                                                                                                                                               HVAC,
         BEGIN
          instance number OF navigate 1 := i
                                                                                                                                                               Interior.
          IF ( CONF(linked_building_element OF
                                                                                   REGIN
                                                                                    EXEC |Sodbe | SQL SELECT * from [AB-
building_element_attribute) >= 0) THEN
                                                                              attribute-designer] where building_element
:Current_rule_data_building_element
END SQL INTO
                                                                                                                                                               WHEN CHANGED
            BEGIN
             EXEC ISodbe 1 SQL SELECT
                                                                                                                                                                 BEGIN
linked attribute of building element from [AB-
attribute-designer] where building element =
current_data.building_element AND
                                                                                                                                                                  IF ( CONF(discipline OF input data) >= 0) THEN
                                                                              building_element_attribute
                                                                                   FND
END SQL INTO domain (tt)
linked_attribute_of_building_element OF
building_element_attribute := tt
END
                                                                                WITH attribute_of_building_element STRING
 linked_building_element =
                                                                                 WHEN CHANGED
                                                                                                                                                                     IF discipline OF input data IS Architect THEN
                                                                                                                                                                      BEGIN
                                                                                   REGIN
                                                                                    EXEC |Sodbe | SQL SELECT * from [AB-
                                                                                                                                                                        designing_discipline OF
                                                                                                                                                           building_element_attribute := "arch"
designing_discipline OF current_data :=
                                                                               general-rules) where building_element
                                                                              :Current_rule_data.building_element AND attribute_of_building_element =
                                                                                                                                                            "arch"
         END
                                                                                Current_rule_data.attribute_of_building_element
                                                                                                                                                                        affected_discipline OF AB_general_rules :=
       FORGET attribute type OF input data
FORGET discipline OF input data
                                                                                                                                                            "arch"
                                                                                                     END SQL INTO AB_general_rules
                                                                                                                                                                      END
                                                                                 WITH affected_discipline STRING
                                                                                WITH result_in_major_modification STRING
                                                                                                                                                                      IF discipline OF input data IS Structure THEN
     END
```

Ahmed Mokhtar . Ph.D Thesis208

BEGIN	1 WEEK COLOROLDED	I FLSE
designing discipline OF	WITH action COMPOUND connect.	BEGIN
building_element_attribute := "stru"	disconnect	text OF message box 2 := "A selected Menu
	WITH records NUMERIC	Group or Building Component is missing"
designing_discipline OF current_data :== "stru"		put up OF message box 2 := TRUE
	WITH transaction COMPOUND	END
affected_discipline OF AB_general_rules :=	commit,	, end
"stru"	rollback	l !
END	WITH append SIMPLE	1000 1001
[INIT FALSE	! *** end of method
IF discipline OF input data IS HVAC THEN	WITH status STRING	END
BEGIN	WITH show error SIMPLE	WITH delete build comp SIMPLE
designing_discipline OF	WITH default error handling SIMPLE	WHEN CHANGED
building_element_attribute := "kvac"	INIT TRUE	BEGIN
designing_discipline OF current_data :=	WITH error message STRING	
"hvac"	WITH native error NUMERIC	IF CONF(current_menu_B_comp OF
affected_discipline OF AB_general_rules :=	WITH trace file STRING	Menus_manipulation_data) > 0 AND CONF(
"hvac"		current_merat_group OF Menus_manipulation_data) >
END	CLASS Linked building element	0 THEN
1	WITH building_element STRING	BEGIN
IF discipline OF input data IS Illuminat THEN		1
BEGIN	CLASS Linked element attribute	EXEC ISodbe 1 SQL DELETE from [AB-
designing_discipline OF	WITH attribute_of_building_element STRING	menu_item] WHERE label =
building_element_attribute := "illu"		:Menus_manipulation_data.current_menu_B_comp
designing_discipline OF current_data :=	CLASS Menus_classified_B_comp	END SQL
"illu"	WITH label STRING	! !
affected_discipline OF AB_general_rules :=		EXEC ISodbc 1 SQL SELECT label from [AB-
"illu"	CLASS Menus_group	menu_item] WHERE main_menu_label =
END	· WITH label STRING	:Menus manipulation_data.current_menu_group
1	·	END SQL INTO
IF discipline OF input data IS Interior THEN	CLASS Menus_group_items	Menus group items
BEGIN	WITH label STRING	
designing discipline OF		MAKE Menus unclassified B comp
building element attribute := "inte"	CLASS Menus_manipulation_data	WITH label := current_menu_B_comp OF
	WITH current_menu_group STRING	Menus manipulation_data
designing_discipline OF current_data := "inte"	WHEN CHANGED	wicturs_membratricu_nere
	BEGIN	FORGET current_menu_B_comp OF
affected_discipline OF AB_general_rules :=		Menus_manipulation_data
"inte"	T 00) T(wears_membranou_aara
END	IF CONF(current_menu_group OF	1000 1 -675
!	Menus_manipulation_data) > 0 THEN	! *** end of IF
IF discipline OF input data IS Envelope THEN	BEGIN	END
BEGIN	EXEC ISodbe ! SQL SELECT label from [AB-	ELSE
designing_discipline OF	menu_item] WHERE main_menu_label =	BEGIN
building_element_attribute := "enve"	:Menus_manipulation_data.current_menu_group	text OF message box 2 := "A selected Menu
designing_discipline OF current_data :=	END SQL INTO	Group or Building Component is missing
"enve"	Menus_group_items	put up OF message box 2 := TRUE
affected_discipline OF AB_general_rules :=		END
"enve"	FORGET current menu_B_comp OF	!
END	Menus manipulation data	! *** end of method
		END
END		END WITH add menu group SIMPLE
END !	Menus_manipulation_data	END
END ! END	Menus_manipulation_data	END WITH add menu group SIMPLE
END ! END ! END	Menus_manipulation_data END	END WITH add menu group SIMPLE WHEN CHANGED
END ! END !	Menus_manipulation_data END	END WITH add menu group SIMPLE WHEN CHANGED BEGIN IF CONF(current_menu_B_comp OF
END ! END ! END WITH attribute type COMPOUND	Menus_manipulation_data END ! ! *** end of method END	END WITH add mean group SIMPLE WHEN CHANGED BEGIN
END ! END ! END WITH attribute type COMPOUND Text data,	Menus_manipulation_data END ** end of method END WITH current_menu_B_comp STRING	END WITH add menu group SIMPLE WHEN CHANGED BEGIN IF CONF(current_menu_B_comp OF
END END END WITH attribute type COMPOUND Text data, Numeric data	Menus_manipulation_data END I *** end of method	END WITH add menu group SIMPLE WHEN CHANGED BEGIN IF CONF(current_menu_B_comp OF Menus_manipulation_data) > 0 AND CONF(
END END END WITH attribute type COMPOUND Text data, Numeric data WHEN CHANGED	Menus_manipulation_data END *** end of method	END WITH add menu group SIMPLE WHEN CHANGED BEGIN IF CONF(current_menu_B_comp OF Menus_manipulation_data) > 0 AND CONF(new_name_of_menu_group OF
END END END WITH attribute type COMPOUND Text data, Numeric data WHEN CHANGED BEGIN	Menus_manipulation_data END ! ! *** end of method END WITH current_menu_B_comp STRING WITH add build comp SIMPLE WHEN CHANGED BEGIN !	END WITH add mean group SIMPLE WHEN CHANGED BEGIN IF CONF(current_mean_B_comp OF Means_manipulation_data) > 0 AND CONF(new_name_of_mean_group OF Means_manipulation_data) > 0 THEN
END END END Text data, Numeric data WHEN CHANGED BEGIN IF (CONF(attribute type OF input data) >= 0)	Menus_manipulation_data END ! ! *** end of method END WITH current_menu_B_comp STRING WITH add build comp SIMPLE WHEN CHANGED BEGIN IF CONF(current_menu_B_comp OF	END WITH add menu group SIMPLE WHEN CHANGED BEGIN IF CONF(current_menu_B_comp OF Menus_manipulation_data) > 0 AND CONF(new_name_of_menu_group OF Menus_manipulation_data) > 0 THEN BEGIN !
END END END WITH attribute type COMPOUND Text data, Numeric data WHEN CHANGED BEGIN	Menus_manipulation_data END ! *** end of method END WITH current_menu_B_comp STRING WITH add build comp SIMPLE WHEN CHANGED BEGIN IF CONF(current_menu_B_comp OF Menus_manipulation_data) > 0 AND CONF(END WITH add menu group SIMPLE WHEN CHANGED BEGIN IF CONF(current_menu_B_comp OF Menus_manipulation_data) > 0 AND CONF(new_name_of_menu_group OF Menus_manipulation_data) > 0 THEN BEGIN
END END END WITH attribute type COMPOUND Text data, Numeric data WHEN CHANGED BEGIN F (CONF(attribute type OF input data) >= 0) THEN BEGIN	Menus_manipulation_data END *** end of method	END WITH add menu group SIMPLE WHEN CHANGED BEGIN IF CONF(current_menu_B_comp OF Menus_manipulation_data) > 0 AND CONF(new_name_of_menu_group OF Menus_manipulation_data) > 0 THEN BEGIN EXEC ISodbc SQL INSERT into [AB-
END END END WITH attribute type COMPOUND Text data, Numeric data WHEN CHANGED BEGIN IF (CONF(attribute type OF input data) >= 0) THEN BEGIN !	Menus_manipulation_data END ! *** end of method END WITH current_menu_B_comp STRING WITH add build comp SIMPLE WHEN CHANGED BEGIN IF CONF(current_menu_B_comp OF Menus_manipulation_data) > 0 AND CONF(END WITH add menu group SIMPLE WHEN CHANGED BEGIN IF CONF(current_menu_B_comp OF Menus_manipulation_data) > 0 AND CONF(new_name_of_menu_group OF Menus_manipulation_data) > 0 THEN BEGIN EXEC ISodbc 1 SQL INSERT into [AB-menu_item] (label, main_menu_label) values (
END END END WITH attribute type COMPOUND Text data, Numeric data WHEN CHANGED BEGIN F (CONF(attribute type OF input data) >= 0) THEN BEGIN	Menus_manipulation_data END ** end of method	END WITH add menu group SIMPLE WHEN CHANGED BEGIN IF CONF(current_menu_B_comp OF Menus_manipulation_data) > 0 AND CONF(new_name_of_menu_group OF Menus_manipulation_data) > 0 THEN BEGIN EXEC ISodbe I SQL INSERT into [AB-menu_tem] (label, main_menu_label) values (Menus_manipulation_data_current_menu_B_comp,
END END WITH attribute type COMPOUND Text data, Numeric data WHEN CHANGED BEGIN IF (CONF(attribute type OF input data) >= 0) THEN BEGIN IF attribute type OF input data IS Text data	Menus_manipulation_data END *** end of method	END WITH add menu group SIMPLE WHEN CHANGED BEGIN IF CONF(current_menu_B_comp OF Menus_manipulation_data) > 0 AND CONF(new_name_of_menu_group OF Menus_manipulation_data) > 0 THEN BEGIN EXEC ISodbe I SQL INSERT into [AB- menu_item] (label, main_menu_label) values (Menus_manipulation_data.current_menu_B_comp, :Menus_manipulation_data.new_name_of_menu_group
END ! END ! END WITH attribute type COMPOUND Text data, Numeric data WHEN CHANGED BEGIN ! IF (CONF(attribute type OF input data) >= 0) THEN BEGIN ! IF attribute type OF input data IS Text data THEN BEGIN	Menus_manipulation_data END *** end of method	END WITH add menu group SIMPLE WHEN CHANGED BEGIN IF CONF(current_menu_B_comp OF Menus_manipulation_data) > 0 AND CONF(new_name_of_menu_group OF Menus_manipulation_data) > 0 THEN BEGIN EXEC ISodbe I SQL INSERT into [AB-menu_tem] (label, main_menu_label) values (Menus_manipulation_data_current_menu_B_comp,
END ! END ! END WITH attribute type COMPOUND Text data, Numeric data WHEN CHANGED BEGIN ! IF (CONF(attribute type OF input data) >= 0) THEN BEGIN ! IF attribute type OF input data IS Text data THEN BEGIN attribute_type OF current_data := "t"	Menus_manipulation_data END *** end of method	END WITH add menu group SIMPLE WHEN CHANGED BEGIN IF CONF(current_menu_B_comp OF Menus_manipulation_data) > 0 AND CONF(new_name_of_menu_group OF Menus_manipulation_data) > 0 THEN BEGIN EXEC ISodbc 1 SQL INSERT into [AB-menu_item] (label, main_menu_label) values (Menus_manipulation_data_current_menu_B_comp, Menus_manipulation_data_new_name_of_menu_group) END SQL
END END END WITH attribute type COMPOUND Text data, Numeric data WHEN CHANGED BEGIN IF (CONF(attribute type OF input data) >= 0) THEN BEGIN IF attribute type OF input data IS Text data THEN BEGIN attribute_type OF current_data := "t" attribute_type OF building_element_attribute	Menus_manipulation_data END *** end of method	END WITH add menu group SIMPLE WHEN CHANGED BEGIN IF CONF(current_menu_B_comp OF Menus_manipulation_data) > 0 AND CONF(new_name_of_menu_group OF Menus_manipulation_data) > 0 THEN BEGIN EXEC ISodbe I SQL INSERT into [AB- menu_item] (label, main_menu_label) values (Menus_manipulation_data.current_menu_B_comp, Menus_manipulation_data.new_name_of_menu_group) END SQL
END ! END ! END ! END WITH attribute type COMPOUND Text data, Numeric data WHEN CHANGED BEGIN ! IF (CONF(attribute type OF input data) >= 0) THEN BEGIN ! IF attribute type OF input data IS Text data THEN BEGIN attribute_type OF current_data := "t" attribute_type OF building_element_attribute := "t"	Menus_manipulation_data END *** end of method	END WITH add menu group SIMPLE WHEN CHANGED BEGIN IF CONF(current_menu_B_comp OF Menus_manipulation_data) > 0 AND CONF(new_name_of_menu_group OF Menus_manipulation_data) > 0 THEN BEGIN EXEC ISodbe I SQL INSERT into [AB- menu_item] (label, main_menu_label) values ('Menus_manipulation_data.current_menu_B_comp, Menus_manipulation_data.new_name_of_menu_group) END SQL EXEC ISodbe I SQL SELECT label from [AB-
END END END WITH attribute type COMPOUND Text data, Numeric data WHEN CHANGED BEGIN IF (CONF(attribute type OF input data) >= 0) THEN BEGIN IF attribute type OF input data IS Text data THEN BEGIN attribute_type OF current_data := "t" attribute_type OF building_element_attribute = "t" END	Menus_manipulation_data END *** end of method	END WITH add menu group SIMPLE WHEN CHANGED BEGIN IF CONF(current_menu_B_comp OF Menus_manipulation_data) > 0 AND CONF(new_name_of_menu_group OF Menus_manipulation_data) > 0 THEN BEGIN EXEC ISodbc I SQL INSERT into [AB-menu_item] (label, main_menu_label) values (: Menus_manipulation_data.current_menu_B_comp, :Menus_manipulation_data.new_name_of_menu_group) END SQL EXEC ISodbc I SQL SELECT label from [AB-menu_item] WHERE main_menu_label =
END END END WITH attribute type COMPOUND Text data, Numeric data WHEN CHANGED BEGIN IF (CONF(attribute type OF input data) >= 0) THEN BEGIN IF attribute type OF input data IS Text data THEN BEGIN BEGIN ### BEGIN #### BEGIN #### BEGIN ###################################	Menus_manipulation_data END *** end of method	END WITH add menu group SIMPLE WHEN CHANGED BEGIN IF CONF(current_menu_B_comp OF Menus_manipulation_data) > 0 AND CONF(new_name_of_menu_group OF Menus_manipulation_data) > 0 THEN BEGIN EXEC ISodbe I SQL INSERT into [AB- menu_item] (label, main_menu_label) values (Menus_manipulation_data.current_menu_B_comp, Menus_manipulation_data.new_name_of_menu_group) END SQL EXEC ISodbe I SQL SELECT label from (AB- menu_item) WHERE main_menu_label = Menus_manipulation_data.new_name_of_menu_group
END ! END ! END ! END WITH attribute type COMPOUND Text data, Numeric data WHEN CHANGED BEGIN ! IF (CONF(attribute type OF input data) >= 0) THEN BEGIN ! IF attribute type OF input data IS Text data THEN BEGIN attribute_type OF current_data := "t" attribute_type OF building_element_attribute := "t" END ! IF attribute type OF input data IS Numeric data	Menus_manipulation_data END *** end of method	END WITH add menu group SIMPLE WHEN CHANGED BEGIN IF CONF(current_menu_B_comp OF Menus_manipulation_data) > 0 AND CONF(new_name_of_menu_group OF Menus_manipulation_data) > 0 THEN BEGIN EXEC ISodbe! SQL INSERT into [AB-menu_item] (label, main_menu_label) values ('Menus_manipulation_data.current_menu_B_comp, 'Menus_manipulation_data.new_name_of_menu_group) END SQL EXEC ISodbe! SQL SELECT label from [AB-menu_item] WHERE main_menu_label = END SQL INTO
END END WITH attribute type COMPOUND Text data, Numeric data WHEN CHANGED BEGIN IF (CONF(attribute type OF input data) >= 0) THEN BEGIN IF attribute type OF input data IS Text data THEN BEGIN attribute_type OF current_data := "t" attribute_type OF building_element_attribute := "t" END IF attribute type OF input data IS Numeric data THEN	Menus_manipulation_data END *** end of method	END WITH add menu group SIMPLE WHEN CHANGED BEGIN IF CONF(current_menu_B_comp OF Menus_manipulation_data) > 0 AND CONF(new_name_of_menu_group OF Menus_manipulation_data) > 0 THEN BEGIN EXEC ISodbc I SQL INSERT into [AB- menu_item] (label, main_menu_label) values ('Menus_manipulation_data.current_menu_B_comp, Menus_manipulation_data.new_name_of_menu_group) END SQL EXEC ISodbc I SQL SELECT label from [AB- menu_item] WHERE main_menu_label = Menus_manipulation_data.new_name_of_menu_group END SQL INTO Menus_group_items
END END END WITH attribute type COMPOUND Text data, Numeric data WHEN CHANGED BEGIN IF (CONF(attribute type OF input data) >= 0) THEN BEGIN IF attribute type OF input data IS Text data THEN BEGIN attribute_type OF current_data := "t" attribute_type OF building_element_attribute := "t" END IF attribute type OF input data IS Numeric data THEN BEGIN BEGIN BEGIN IF attribute type OF input data IS Numeric data	Menus_manipulation_data END *** end of method	END WITH add menu group SIMPLE WHEN CHANGED BEGIN IF CONF(current_menu_B_comp OF Menus_manipulation_data) > 0 AND CONF(new_name_of_menu_group OF Menus_manipulation_data) > 0 THEN BEGIN EXEC ISodbe I SQL INSERT into [AB- menu_item] (label, main_menu_label) values (Menus_manipulation_data.current_menu_B_comp, Menus_manipulation_data.new_name_of_menu_group) END SQL EXEC ISodbe I SQL SELECT label from [AB- menu_item] WHERE main_menu_label = Menus_manipulation_data.new_name_of_menu_group END SQL INTO Menus_group_items !
END END END WITH attribute type COMPOUND Text data, Numeric data WHEN CHANGED BEGIN IF (CONF(attribute type OF input data) >= 0) THEN BEGIN IF attribute type OF input data IS Text data THEN BEGIN attribute_type OF current_data := "t" END IF attribute type OF building_element_attribute := "t" END IF attribute type OF input data IS Numeric data THEN BEGIN BEGIN BEGIN STRIBUTE TYPE OF INPUT data IS Numeric data	Menus_manipulation_data END *** end of method	END WITH add menu group SIMPLE WHEN CHANGED BEGIN IF CONF(current_menu_B_comp OF Menus_manipulation_data) > 0 AND CONF(new_name_of_menu_group OF Menus_manipulation_data) > 0 THEN BEGIN EXEC ISodbe! SQL INSERT into [AB-menu_item] (label, main_menu_label) values ('Menus_manipulation_data.current_menu_B_comp, 'Menus_manipulation_data.new_name_of_menu_group) END SQL EXEC ISodbe! SQL SELECT label from [AB-menu_item] WHERE main_menu_label = 'Menus_manipulation_data.new_name_of_menu_group END SQL INTO Menus_group_items FIND Menus_unclassified_B_comp
END END END WITH attribute type COMPOUND Text data, Numeric data WHEN CHANGED BEGIN IF (CONF(attribute type OF input data) >= 0) THEN BEGIN IF attribute type OF input data IS Text data THEN BEGIN attribute_type OF current_data := "t" END IF attribute type OF building_element_attribute "" END IF attribute_type OF input data IS Numeric data THEN BEGIN STRIBUTE_type OF input data IS Numeric data THEN BEGIN attribute_type OF input data IS Numeric data	Menus_manipulation_data END *** end of method	END WITH add menu group SIMPLE WHEN CHANGED BEGIN IF CONF(current_menu_B_comp OF Menus_manipulation_data) > 0 AND CONF(new_name_of_menu_group OF Menus_manipulation_data) > 0 THEN BEGIN EXEC ISodbc 1 SQL INSERT into [AB- menu_item] (label, main_menu_label) values ('Menus_manipulation_data.current_menu_B_comp, Menus_manipulation_data.new_name_of_menu_group) END SQL EXEC ISodbc 1 SQL SELECT label from [AB- menu_item] WHERE main_menu_label = Menus_manipulation_data.new_name_of_menu_group END SQL INTO Menus_group_items FIND Menus_unclassified_B_comp WHERE label OF
END END END WITH attribute type COMPOUND Text data, Numeric data WHEN CHANGED BEGIN IF (CONF(attribute type OF input data) >= 0) THEN BEGIN IF strribute type OF input data IS Text data THEN BEGIN attribute_type OF current_data := "t" strribute_type OF building_element_attribute := "t" END IF attribute type OF input data IS Numeric data THEN BEGIN strribute_type OF input data IS Numeric data THEN BEGIN strribute_type OF current_data := "n" attribute_type OF building_element_attribute := "t"	Menus_manipulation_data END *** end of method	END WITH add menu group SIMPLE WHEN CHANGED BEGIN IF CONF(current_menu_B_comp OF Menus_manipulation_data) > 0 AND CONF(new_name_of_menu_group OF Menus_manipulation_data) > 0 THEN BEGIN EXEC ISodbe I SQL INSERT into [AB- menu_item] (label, main_menu_label) values (Menus_manipulation_data.current_menu_B_comp, Menus_manipulation_data.new_name_of_menu_group) END SQL EXEC ISodbe I SQL SELECT label from [AB- menu_item] WHERE main_menu_label = Menus_manipulation_data.new_name_of_menu_group END SQL INTO Menus_manipulation_data.new_name_of_menu_group END SQL INTO Menus_group_items FIND Menus_unclassified_B_comp WHERE label OF Menus_unclassified_B_comp = current_menu_B_comp Menus_unclassified_B_comp = current_menu_B_comp
END END END WITH attribute type COMPOUND Text data, Numeric data WHEN CHANGED BEGIN IF (CONF(attribute type OF input data) >= 0) THEN BEGIN IF attribute type OF input data IS Text data THEN BEGIN ### ### ### END ### BEGIN IF attribute_type OF ourtest_data := "t" ### ### ### END ### BEGIN ### ### ### ### ### ### ###	Menus_manipulation_data END *** end of method	END WITH add menu group SIMPLE WHEN CHANGED BEGIN IF CONF(current_menu_B_comp OF Menus_manipulation_data) > 0 AND CONF(new_name_of_menu_group OF Menus_manipulation_data) > 0 THEN BEGIN EXEC ISodbe ! SQL INSERT into [AB- menu_item] (label, main_menu_label) values (Menus_manipulation_data.current_menu_B_comp, Menus_manipulation_data.new_name_of_menu_group) END SQL EXEC ISodbe ! SQL SELECT label from [AB- menu_item] WHERE main_menu_label = Menus_manipulation_data.new_name_of_menu_group END SQL INTO Menus_manipulation_data.new_name_of_menu_group END SQL INTO Menus_group_items FIND Menus_unclassified_B_comp WHERE label OF Menus_unclassified_B_comp OF Menus_manipulation_data
END END WITH attribute type COMPOUND Text data, Numeric data WHEN CHANGED BEGIN IF (CONF(attribute type OF input data) >= 0) THEN BEGIN IF attribute type OF input data IS Text data THEN BEGIN attribute_type OF current_data := "t" attribute_type OF building_element_attribute := "t" END IF attribute type OF input data IS Numeric data THEN BEGIN STITUEN BEGIN STITUEN BEGIN STITUEN BEGIN STITUEN END IF attribute_type OF input data IS Numeric data THEN BEGIN STITUEN BEGIN STITUEN STITUE	Menus_manipulation_data END I *** end of method END WITH current_menu_B_comp STRING WITH add build comp SIMPLE WHEN CHANGED BEGIN IF CONF(current_menu_B_comp OF Menus_manipulation_data) > 0 AND CONF(current_menu_group OF Menus_manipulation_data) > 0 THEN BEGIN EXEC ISodbe 1 SQL INSERT into [AB-menu_item] (label, main_menu_label) values (Menus_manipulation_data.current_menu_group) END SQL EXEC ISodbe 1 SQL SELECT label from [AB-menu_item] WHERE main_menu_label * :Menus_manipulation_data.current_menu_group END SQL EXEC ISOdbe 1 SQL SELECT label from [AB-menu_item] WHERE main_menu_label * :Menus_manipulation_data.current_menu_group END SQL INTO Menus_group_items !	END WITH add menu group SIMPLE WHEN CHANGED BEGIN IF CONF(current_menu_B_comp OF Menus_manipulation_data) > 0 AND CONF(new_name_of_menu_group OF Menus_manipulation_data) > 0 THEN BEGIN EXEC ISodbc SQL INSERT into [AB- menu_item] (label, main_menu_label) values (Menus_manipulation_data.current_menu_B_comp, Menus_manipulation_data.new_name_of_menu_group) END SQL EXEC ISodbc SQL SELECT label from [AB- menu_item] WHERE main_menu_label = Menus_manipulation_data.new_name_of_menu_group END SQL INTO Menus_group_items FIND Menus_unclassified_B_comp WHERE label OF Menus_unclassified_B_comp WHERE tabel OF Menus_unclassified_B_comp WHERE found WHEN FOUND
END END END WITH attribute type COMPOUND Text data, Numeric data WHEN CHANGED BEGIN IF (CONF(attribute type OF input data) >= 0) THEN BEGIN IF attribute type OF input data IS Text data THEN BEGIN attribute_type OF current_data := "t" stribute_type OF building_element_attribute != "t" END ! IF attribute type OF input data IS Numeric data THEN BEGIN attribute_type OF input data IS Numeric data THEN BEGIN attribute_type OF input data IS Numeric data THEN BEGIN attribute_type OF building_element_attribute := "a" attribute_type OF building_element_attribute := "a" END	Menus_manipulation_data END I *** end of method END WITH current_menu_B_comp STRING WITH add build comp SIMPLE WHEN CHANGED BEGIN IF CONF(current_menu_B_comp OF Menus_manipulation_data) > 0 AND CONF(current_menu_group OF Menus_manipulation_data) > 0 THEN BEGIN EXEC ISodbe 1 SQL INSERT into [AB-menu_item] [label, main_menu_label) values (:Menus_manipulation_data.current_menu_group) END SQL EXEC ISodbe 1 SQL SELECT label from [AB-menu_item] WHERE main_menu_label ** :Menus_manipulation_data.current_menu_group END SQL INTO Menus_manipulation_data.current_menu_group END SQL INTO Menus_group_items FIND Menus_unclassified_B_comp WHERE label OF	END WITH add menu group SIMPLE WHEN CHANGED BEGIN IF CONF(current_menu_B_comp OF Menus_manipulation_data) > 0 AND CONF(new_name_of_menu_group OF Menus_manipulation_data) > 0 THEN BEGIN EXEC ISodbc I SQL INSERT into [AB- menu_item] (label, main_menu_label) values (Menus_manipulation_data.current_menu_B_comp, Menus_manipulation_data.current_menu_B_roup END SQL EXEC ISodbc I SQL SELECT label from [AB- menu_item] WHERE main_menu_label = Menus_manipulation_data.new_name_of_menu_group END SQL INTO Menus_group_items FIND Menus_unclassified_B_comp WHERE label OF Menus_unclassified_B_comp OF Menus_unclassified_B_comp OF Menus_unclassified_B_comp OF Menus_unclassified_B_comp FORGET_CURRENT
END END WITH attribute type COMPOUND Text data, Numeric data WHEN CHANGED BEGIN IF (CONF(attribute type OF input data) >= 0) THEN BEGIN IF attribute type OF input data IS Text data THEN BEGIN Stribute_type OF current_data := "t" attribute_type OF building_element_attribute != "t" END IF attribute type OF input data IS Numeric data THEN BEGIN stribute_type OF current_data := "n" attribute_type OF current_data := "n" attribute_type OF building_element_attribute END END END	Menus_manipulation_data END *** end of method	END WITH add menu group SIMPLE WHEN CHANGED BEGIN IF CONF(current_menu_B_comp OF Menus_manipulation_data) > 0 AND CONF(new_name_of_menu_group OF Menus_manipulation_data) > 0 THEN BEGIN EXEC ISodbe I SQL INSERT into [AB- menu_item] (label, main_menu_label) values (Menus_manipulation_data.new_name_of_menu_group) END SQL EXEC ISodbe I SQL SELECT label from [AB- menu_item] WHERE main_menu_label = 'Menus_manipulation_data.new_name_of_menu_group END SQL INTO Menus_manipulation_data.new_name_of_menu_group END SQL INTO Menus_group_items FIND Menus_unclassified_B_comp WHERE label OF Menus_unclassified_B_comp = current_menu_B_comp OF Menus_unclassified_B_comp FORGET_CURRENT Menus_unclassified_B_comp
END END END WITH attribute type COMPOUND Text data, Numeric data WHEN CHANGED BEGIN IF (CONF(attribute type OF input data) >= 0) THEN BEGIN IF attribute type OF input data IS Text data THEN BEGIN attribute_type OF current_data := "t" stribute_type OF building_element_attribute != "t" END ! IF attribute type OF input data IS Numeric data THEN BEGIN attribute_type OF input data IS Numeric data THEN BEGIN attribute_type OF input data IS Numeric data THEN BEGIN attribute_type OF building_element_attribute := "a" attribute_type OF building_element_attribute := "a" END	Menus_manipulation_data END *** end of method	END WITH add menu group SIMPLE WHEN CHANGED BEGIN IF CONF(current_menu_B_comp OF Menus_manipulation_data) > 0 AND CONF(new_name_of_menu_group OF Menus_manipulation_data) > 0 THEN BEGIN EXEC ISodbe! SQL INSERT into [AB- menu_item] (label, main_menu_label) values (Menus_manipulation_data.current_menu_B_comp, Menus_manipulation_data.new_name_of_menu_group) END SQL EXEC ISodbe! SQL SELECT label from [AB- menu_item] WHERE main_menu_label = Menus_manipulation_data.new_name_of_menu_group END SQL INTO Menus_group_items FIND Menus_unclassified_B_comp WHERE label OF Menus_unclassified_B_comp OF Menus_manipulation_data WHEN FOUND FORGET_CURRENT Menus_unclassified_B_comp FIND END FIND END
END END END WITH attribute type COMPOUND Text data, Numeric data WHEN CHANGED BEGIN IF (CONF(attribute type OF input data) >= 0) THEN BEGIN IF attribute type OF input data IS Text data THEN BEGIN attribute_type OF current_data := "t" stribute_type OF building_element_attribute := "t" END IF attribute type OF input data IS Numeric data THEN BEGIN attribute_type OF input data IS Numeric data THEN BEGIN attribute_type OF input data IS Numeric data THEN BEGIN attribute_type OF current_data := "n" attribute_type OF building_element_attribute := "a" END END	Menus_manipulation_data END I *** end of method END WITH current_menu_B_comp STRING WITH add build comp SIMPLE WHEN CHANGED BEGIN IF CONF(current_menu_B_comp OF Menus_manipulation_data) > 0 AND CONF(current_menu_group OF Menus_manipulation_data) > 0 THEN BEGIN EXEC ISodbe 1 SQL INSERT into [AB-menu_item] (label, main_menu_label) values (Menus_manipulation_data_current_menu_group) END SQL EXEC ISodbe 1 SQL SELECT label from [AB-menu_item] WHERE main_menu_label ** :Menus_manipulation_data_current_menu_group END SQL EXEC ISodbe 1 SQL SELECT label from [AB-menu_item] WHERE main_menu_label ** :Menus_manipulation_data_current_menu_group END SQL INTO Menus_group_items FIND Menus_unclassified_B_comp WHERE label OF Menus_unclassified_B_comp = current_menu_B_comp OF Menus_manipulation_data WHEN FOUND	END WITH add menu group SIMPLE WHEN CHANGED BEGIN IF CONF(current_menu_B_comp OF Menus_manipulation_data) > 0 AND CONF(new_name_of_menu_group OF Menus_manipulation_data) > 0 THEN BEGIN EXEC ISodbc I SQL INSERT into [AB- menu_item] (label, main_menu_label) values (Menus_manipulation_data.current_menu_B_comp, :Menus_manipulation_data.new_name_of_menu_group) END SQL EXEC ISodbc I SQL SELECT label from [AB- menu_item] WHERE main_menu_label = :Menus_manipulation_data.new_name_of_menu_group END SQL INTO Menus_group_items FIND Menus_unclassified_B_comp WHERE label OF Menus_unclassified_B_comp = current_menu_B_comp OF Menus_unclassified_B_comp FORGET_CURRENT Menus_unclassified_B_comp FIND END !
END END END WITH attribute type COMPOUND Text data, Numeric data WHEN CHANGED BEGIN IF (CONF(attribute type OF input data) >= 0) THEN BEGIN IF attribute type OF input data IS Text data THEN BEGIN Stribute_type OF current_data := "t" attribute_type OF building_element_attribute != "t" END IF attribute type OF input data IS Numeric data THEN BEGIN stribute_type OF current_data := "n" attribute_type OF current_data := "n" attribute_type OF building_element_attribute != "t" END END CLASS ISodbe INHERITS add on	Menus_manipulation_data END *** end of method	END WITH add menu group SIMPLE WHEN CHANGED BEGIN IF CONF(current_menu_B_comp OF Menus_manipulation_data) > 0 AND CONF(new_name_of_menu_group OF Menus_manipulation_data) > 0 THEN BEGIN EXEC ISodbe I SQL INSERT into [AB- menu_item] (label, main_menu_label) values (Menus_manipulation_data.current_menu_B_comp, Menus_manipulation_data.new_name_of_menu_group END SQL EXEC ISodbe I SQL SELECT label from [AB- menu_item] WHERE main_menu_label = Menus_manipulation_data.new_name_of_menu_group END SQL INTO Menus_group_items FIND Menus_unclassified_B_comp WHERE label OF Wenus_unclassified_B_comp = current_menu_B_comp OF Menus_unclassified_B_comp = current_menu_B_comp FORGET_CURRENT Menus_unclassified_B_comp FIND END MAKE Menus_group
END END END WITH attribute type COMPOUND Text data, Numeric data WHEN CHANGED BEGIN IF (CONF(attribute type OF input data) >= 0) THEN BEGIN IF attribute type OF input data IS Text data THEN BEGIN attribute_type OF current_data := "t" END IF attribute type OF building_element_attribute """ END IF attribute_type OF input data IS Numeric data THEN BEGIN attribute_type OF input data IS Numeric data THEN BEGIN attribute_type OF current_data := "n" attribute_type OF current_data := "n" attribute_type OF current_data := "n" attribute_type OF building_element_attribute """ END CLASS ISodbe INHERITS add on WITH data source STRING	Menus_manipulation_data END I *** end of method END WITH current_menu_B_comp STRING WITH add build comp SIMPLE WHEN CHANGED BEGIN IF CONF(current_menu_B_comp OF Menus_manipulation_data) > 0 AND CONF(current_menu_group OF Menus_manipulation_data) > 0 THEN BEGIN EXEC ISodbe I SQL INSERT into [AB-menu_item] (label, main_menu_label) values (Menus_manipulation_data.current_menu_B_comp, :Menus_manipulation_data.current_menu_group) END SQL EXEC ISodbe I SQL SELECT label from [AB-menu_item] WHERE main_menu_label * :Menus_manipulation_data.current_menu_group END SQL INTO Menus_group_items FIND Menus_unclassified_B_comp WHERE label OF Menus_manipulation_data WHEN FOUND FORGET CURRENT Menus_unclassified_B_comp	END WITH add menu group SIMPLE WHEN CHANGED BEGIN IF CONF(current_menu_B_comp OF Menus_manipulation_data) > 0 AND CONF(new_name_of_menu_group OF Menus_manipulation_data) > 0 THEN BEGIN EXEC ISodbe! SQL INSERT into [AB- menu_item] (label, main_menu_label) values (Menus_manipulation_data.current_menu_B_comp, Menus_manipulation_data.new_name_of_menu_group END SQL EXEC ISodbe! SQL SELECT label from [AB- menu_item] WHERE main_menu_label = Menus_manipulation_data.new_name_of_menu_group END SQL INTO Menus_group_items FIND Menus_unclassified_B_comp WHERE label OF Menus_unclassified_B_comp WHEN FOUND FORGET CURRENT Menus_tastified_B_comp FIND END MAKE Menus_group WITH label:= new_name_of_menu_group WITH label:= new_name_of_menu_group
END END END WITH attribute type COMPOUND Text data, Numeric data WHEN CHANGED BEGIN IF (CONF(attribute type OF input data) >= 0) THEN BEGIN IF attribute type OF input data IS Text data THEN BEGIN attribute_type OF current_data := "t" attribute_type OF building_element_attribute := "t" END IF attribute type OF input data IS Numeric data THEN BEGIN attribute_type OF input data IS Numeric data THEN BEGIN attribute_type OF current_data := "n" attribute_type OF building_element_attribute := "n" END END CLASS ISodbe INHERITS add on WITH data source STRING WITH userid STRING WITH userid STRING	Menus_manipulation_data END *** end of method END WITH current_menu_B_comp STRING WITH add build comp SIMPLE WHEN CHANGED BEGIN IF CONF(current_menu_B_comp OF Menus_manipulation_data) > 0 AND CONF(current_menu_group OF Menus_manipulation_data) > 0 THEN BEGIN EXEC ISodbe SQL INSERT into [AB-menu_item] (label, main_menu_label) values (Menus_manipulation_data_current_menu_B_comp, :Menus_manipulation_data_current_menu_group) END SQL EXEC ISodbe SQL SELECT label from [AB-menu_item] WHERE main_menu_label = :Menus_manipulation_data_current_menu_group END SQL INTO Menus_manipulation_data_current_menu_group END SQL INTO Menus_group_items FIND Menus_unclassified_B_comp WHERE label OF Menus_unclassified_B_comp = current_menu_B_comp OF Menus_manipulation_data WHEN FOUND FORGET CURRENT Menus_unclassified_B_comp FORGET CURRENT Menus_unclassified_B_comp FORGET CURRENT Menus_unclassified_B_comp FORGET CURRENT Menus_unclassified_B_comp FIND END	END WITH add menu group SIMPLE WHEN CHANGED BEGIN IF CONF(current_menu_B_comp OF Menus_manipulation_data) > 0 AND CONF(new_name_of_menu_group OF Menus_manipulation_data) > 0 THEN BEGIN EXEC ISodbe I SQL INSERT into [AB- menu_item] (label, main_menu_label) values (Menus_manipulation_data.current_menu_B_comp, Menus_manipulation_data.new_name_of_menu_group END SQL EXEC ISodbe I SQL SELECT label from [AB- menu_item] WHERE main_menu_label = Menus_manipulation_data.new_name_of_menu_group END SQL INTO Menus_group_items FIND Menus_unclassified_B_comp WHERE label OF Wenus_unclassified_B_comp = current_menu_B_comp OF Menus_unclassified_B_comp = current_menu_B_comp FORGET_CURRENT Menus_unclassified_B_comp FIND END MAKE Menus_group
END END END WITH attribute type COMPOUND Text data, Numeric data WHEN CHANGED BEGIN IF (CONF(attribute type OF input data) >= 0) THEN BEGIN IF attribute type OF input data IS Text data THEN BEGIN END IF attribute type OF ourtent_data := "t" attribute_type OF building_element_attribute "= "t" END IF attribute type OF current_data := "n" attribute_type OF current_data := "n" attribute_type OF current_data := "n" attribute_type OF building_element_attribute "= "n" END END CLASS ISodbe INHERITS add on WITH data source STRING WITH userid STRING WITH password STRING WITH password STRING WITH password STRING WITH password STRING	Menus_manipulation_data END I *** end of method END WITH durrent_menu_B_comp STRING WITH add build comp SIMPLE WHEN CHANGED BEGIN IF CONF(current_menu_B_comp OF Menus_manipulation_data) > 0 AND CONF(current_menu_group OF Menus_manipulation_data) > 0 THEN BEGIN EXEC ISodbe I SQL INSERT into [AB-menu_item] [label, main_menu_label] values (!Menus_manipulation_data.current_menu_group) END SQL EXEC ISodbe I SQL SELECT label from [AB-menu_item] WHERE main_menu_label = :Menus_manipulation_data.current_menu_group END SQL INTO Menus_manipulation_data.current_menu_group END SQL INTO Menus_manipulation_data.current_menu_group END SQL INTO Menus_manipulation_data.current_menu_group END SQL INTO Menus_unclassified_B_comp = current_menu_B_comp OF Menus_unclassified_B_comp = current_menu_B_comp OF Menus_unclassified_B_comp = current_menu_B_comp OF Menus_unclassified_B_comp = current_menu_B_comp FORGET_CURRENT Menus_unclassified_B_comp FIND_END	END WITH add menu group SIMPLE WHEN CHANGED BEGIN IF CONF(current_menu_B_comp OF Menus_manipulation_data) > 0 AND CONF(new_name_of_menu_group OF Menus_manipulation_data) > 0 THEN BEGIN EXEC ISodbc I SQL INSERT into [AB- menu_item] (label, main_menu_label) values (Menus_manipulation_data.current_menu_B_comp, Menus_manipulation_data.new_name_of_menu_group END SQL EXEC ISodbc I SQL SELECT label from [AB- menu_item] WHERE main_menu_label = Menus_manipulation_data.new_name_of_menu_group END SQL INTO Menus_group_items FIND Menus_unclassified_B_comp WHERE label OF WHERE label OF OF Menus_manipulation_data WHEN FOUND FORGET CURRENT Menus_unclassified_B_comp FIND END MAKE Menus_group WITH label := new_name_of_menu_group OF Menus_manipulation_data MAKE Menus_group WITH label := new_name_of_menu_group OF Menus_manipulation_data
END END END WITH attribute type COMPOUND Text data, Numeric data WHEN CHANGED BEGIN IF (CONF(attribute type OF input data) >= 0) THEN BEGIN IF attribute type OF input data IS Text data THEN BEGIN attribute_type OF current_data := "t" attribute_type OF building_element_attribute := "t" END IF attribute type OF input data IS Numeric data THEN BEGIN attribute_type OF input data IS Numeric data THEN BEGIN attribute_type OF current_data := "n" attribute_type OF building_element_attribute := "t" END END CLASS ISodbe INHERITS add on WITH data source STRING WITH userid STRING WITH pessword STRING WITH connection string STRING	Menus_manipulation_data END I *** end of method END WITH current_menu_B_comp STRING WITH add build comp SIMPLE WHEN CHANGED BEGIN IF CONF(currens_menu_B_comp OF Menus_manipulation_data) > 0 AND CONF(current_menu_group OF Menus_manipulation_data) > 0 THEN BEGIN EXEC ISodbe 1 SQL INSERT into [AB-menu_item] (label, main_menu_label) values (Menus_manipulation_data.current_menu_B_comp, :Menus_manipulation_data.current_menu_group) END SQL EXEC ISodbe 1 SQL SELECT label from [AB-menu_item] WHERE main_menu_label * :Menus_manipulation_data.current_menu_group END SQL INTO Menus_group_items FIND Menus_unclassified_B_comp WHERE label OF Menus_manipulation_data WHEN FOUND FORGET CURRENT Menus_unclassified_B_comp FIND END FORGET CURRENT Menus_unclassified_B_comp FIND END FORGET current_menu_B_comp OF	END WITH add menu group SIMPLE WHEN CHANGED BEGIN IF CONF(current_menu_B_comp OF Menus_manipulation_data) > 0 AND CONF(new_name_of_menu_group OF Menus_manipulation_data) > 0 THEN BEGIN EXEC ISodbe! SQL INSERT into [AB- menu_item] (label, main_menu_label) values ('Menus_manipulation_data.current_menu_B_comp, 'Menus_manipulation_data.new_name_of_menu_group END SQL EXEC ISodbe! SQL SELECT label from [AB- menu_item] WHERE main_menu_label = 'Menus_manipulation_data.new_name_of_menu_group END SQL INTO Menus_group_items FIND Menus_unclassified_B_comp WHERE label OF Menus_unclassified_B_comp = current_menu_B_comp OF Menus_manipulation_data WHEN FOUND FORGET CURRENT Menus_nanipulation_data * MAKE Menus_group WITH label:= new_name_of_menu_group OF Menus_manipulation_data * MAKE Menus_group WITH label:= new_name_of_menu_group OF Menus_manipulation_data * FORGET current_menu_B_comp OF
END END END WITH attribute type COMPOUND Text data, Numeric data WHEN CHANGED BEGIN IF (CONF(attribute type OF input data) >= 0) THEN BEGIN IF attribute type OF input data IS Text data THEN BEGIN attribute_type OF current_data := "t" attribute_type OF building_element_attribute := "t" END IF attribute type OF input data IS Numeric data THEN BEGIN attribute_type OF current_data := "n" attribute_type OF building_element_attribute := "t" END END CLASS ISodot INHERITS add on WITH data source STRING WITH userid STRING WITH password STRING WITH connection string STRING WITH connection prompting SIMPLE	Menus_manipulation_data END I *** end of method END WITH durrent_menu_B_comp STRING WITH add build comp SIMPLE WHEN CHANGED BEGIN IF CONF(current_menu_B_comp OF Menus_manipulation_data) > 0 AND CONF(current_menu_group OF Menus_manipulation_data) > 0 THEN BEGIN EXEC ISodbe I SQL INSERT into [AB-menu_item] [label, main_menu_label] values (!Menus_manipulation_data.current_menu_group) END SQL EXEC ISodbe I SQL SELECT label from [AB-menu_item] WHERE main_menu_label = :Menus_manipulation_data.current_menu_group END SQL INTO Menus_manipulation_data.current_menu_group END SQL INTO Menus_manipulation_data.current_menu_group END SQL INTO Menus_manipulation_data.current_menu_group END SQL INTO Menus_unclassified_B_comp = current_menu_B_comp OF Menus_unclassified_B_comp = current_menu_B_comp OF Menus_unclassified_B_comp = current_menu_B_comp OF Menus_unclassified_B_comp = current_menu_B_comp FORGET_CURRENT Menus_unclassified_B_comp FIND_END	END WITH add menu group SIMPLE WHEN CHANGED BEGIN IF CONF(current_menu_B_comp OF Menus_manipulation_data) > 0 AND CONF(new_name_of_menu_group OF Menus_manipulation_data) > 0 THEN BEGIN EXEC ISodbc I SQL INSERT into [AB- menu_item] (label, main_menu_label) values (Menus_manipulation_data.current_menu_B_comp, Menus_manipulation_data.new_name_of_menu_group) END SQL EXEC ISodbc I SQL SELECT label from [AB- menu_item] WHERE main_menu_label = Menus_manipulation_data.new_name_of_menu_group END SQL INTO Menus_group_items FIND Menus_unclassified_B_comp WHERE label OF Menus_unclassified_B_comp OF Menus_unclassified_B_comp FORGET_CURRENT Menus_unclassified_B_comp FIND END MAKE Menus_group WITH label := new_name_of_menu_group OF Menus_manipulation_data FORGET_current_menu_B_comp OF Menus_manipulation_data
END END END WITH attribute type COMPOUND Text data, Numeric data WHEN CHANGED BEGIN IF (CONF(attribute type OF input data) >= 0) THEN BEGIN IF attribute type OF input data IS Text data THEN BEGIN attribute_type OF current_data := "t" attribute_type OF building_element_attribute := "t" END IF attribute type OF current_data := "n" attribute_type OF current_data := "n" attribute_type OF current_data := "n" attribute_type OF building_element_attribute := "t" END END END CLASS ISodbe INHERITS add on WITH data source STRING WITH connection string STRING WITH connection string STRING WITH connection prompting SIMPLE INIT FALSE	Menus_manipulation_data END I *** end of method END WITH durrent_menu_B_comp STRING WITH add build comp SIMPLE WHEN CHANGED BEGIN IF CONF(current_menu_B_comp OF Menus_manipulation_data) > 0 AND CONF(current_menu_group OF Menus_manipulation_data) > 0 THEN BEGIN EXEC ISodbe 1 SQL INSERT into [AB- menu_item] [label, main_menu_label) values (:Menus_manipulation_data_current_menu_group) END SQL EXEC ISodbe 1 SQL SELECT label from [AB- menu_item] WHERE main_menu_label = :Menus_manipulation_data_current_menu_group END SQL INTO Menus_group_items FIND Menus_unclassified_B_comp WHERE label OF Menus_unclassified_B_comp = current_menu_B_comp OF Menus_unclassified_B_comp = current_menu_B_comp OF Menus_unclassified_B_comp = current_menu_B_comp OF Menus_unclassified_B_comp FIND END FORGET current_menu_B_comp OF Menus_unclassified_B_comp FIND END FORGET current_menu_B_comp OF Menus_manipulation_data	END WITH add menu group SIMPLE WHEN CHANGED BEGIN IF CONF(current_menu_B_comp OF Menus_manipulation_data) > 0 AND CONF(new_name_of_menu_group OF Menus_manipulation_data) > 0 THEN BEGIN EXEC ISodbc I SQL INSERT into [AB- menu_item] (label, main_menu_label) values (Menus_manipulation_data.current_menu_B_comp, Menus_manipulation_data.new_name_of_menu_group) END SQL EXEC ISodbc I SQL SELECT label from [AB- menu_item] WHERE main_menu_label == Menus_manipulation_data.new_name_of_menu_group END SQL INTO Menus_group_items FIND Menus_unclassified_B_comp WHERE label OF Menus_unclassified_B_comp = current_menu_B_comp OF Menus_manipulation_data WHEN FOUND FORGET CURRENT Menus_unclassified_B_comp FIND END MAKE Menus_group WITH label := new_name_of_menu_group OF Menus_manipulation_data FORGET current_menu_B_comp OF Menus_manipulation_data FORGET current_menu_B_comp OF Menus_manipulation_data FORGET current_menu_B_comp OF
END END END WITH attribute type COMPOUND Text data, Numeric data WHEN CHANGED BEGIN IF (CONF(attribute type OF input data) >= 0) THEN BEGIN IF attribute type OF input data IS Text data THEN BEGIN attribute_type OF current_data := "t" attribute_type OF building_element_attribute := "t" END IF attribute type OF input data IS Numeric data THEN BEGIN attribute_type OF current_data := "n" attribute_type OF building_element_attribute := "t" END END CLASS ISodot INHERITS add on WITH data source STRING WITH userid STRING WITH password STRING WITH connection string STRING WITH connection prompting SIMPLE	Menus_manipulation_data END I *** end of method END WITH current_menu_B_comp STRING WITH add build comp SIMPLE WHEN CHANGED BEGIN IF CONF(currens_menu_B_comp OF Menus_manipulation_data) > 0 AND CONF(current_menu_group OF Menus_manipulation_data) > 0 THEN BEGIN EXEC ISodbe 1 SQL INSERT into [AB-menu_item] (label, main_menu_label) values (Menus_manipulation_data.current_menu_B_comp, :Menus_manipulation_data.current_menu_group) END SQL EXEC ISodbe 1 SQL SELECT label from [AB-menu_item] WHERE main_menu_label * :Menus_manipulation_data.current_menu_group END SQL INTO Menus_group_items FIND Menus_unclassified_B_comp WHERE label OF Menus_manipulation_data WHEN FOUND FORGET CURRENT Menus_unclassified_B_comp FIND END FORGET CURRENT Menus_unclassified_B_comp FIND END FORGET current_menu_B_comp OF	END WITH add menu group SIMPLE WHEN CHANGED BEGIN IF CONF(current_menu_B_comp OF Menus_manipulation_data) > 0 AND CONF(new_name_of_menu_group OF Menus_manipulation_data) > 0 THEN BEGIN EXEC ISodbc I SQL INSERT into [AB- menu_item] (label, main_menu_label) values (Menus_manipulation_data.current_menu_B_comp, Menus_manipulation_data.new_name_of_menu_group) END SQL EXEC ISodbc I SQL SELECT label from [AB- menu_item] WHERE main_menu_label = Menus_manipulation_data.new_name_of_menu_group END SQL INTO Menus_group_items FIND Menus_unclassified_B_comp WHERE label OF Menus_unclassified_B_comp OF Menus_unclassified_B_comp FORGET_CURRENT Menus_unclassified_B_comp FIND END MAKE Menus_group WITH label := new_name_of_menu_group OF Menus_manipulation_data FORGET_current_menu_B_comp OF Menus_manipulation_data

Menus_manipulation_data	END SQL	WITH building_element STRING
! ! *** end of UF	END !	CLASS NEW to make a new project
END	FIND Menus group	WITH from another project SIMPLE
ELSE	WHERE label OF Menus group =	WITH from systems SIMPLE
BEGIN	current_menu_group OF Menus_manipulation_data	WHEN CHANGED
text OF message box 2 := "Type a New Menu	WHEN FOUND	BEGIN EXEC ISodho 2 SQL select distinct (discipline)
Group and select a Building Component* put up OF message box 2 := TRUE	label OF Menus_group := new_name_of_menu_group OF	from [building_systems]
END	Menus manipulation data	END SQL INTO NEW design_disciplines
1	FIND END	1
•	!	action OF ISodbc 1 IS connect := TRUE
*** end of method	current_menu_group OF	visible OF main window := FALSE
END WITH delete menu group SIMPLE	Menus_manipulation_data := new_name_of_menu_group OF	visible OF configuration window := TRUE
WHEN CHANGED	Menus manipulation data	output OF configuration window := NEW from
BEGIN	FORGET new_name_of_menu_group OF	systems display
!	Menus_manipulation_data	END
IF CONF(current_ments_group OF Ments manipulation data) > 0 THEN	I *** end of IF END	WITH from components SIMPLE WHEN CHANGED
BEGIN	FLSE	BEGIN
!	BEGIN	EXEC Isodbe 2 SQL select distinct
EXEC 15odbc 1 SQL DELETE from [AB-	text OF message box 2 := "A selected Menu	(building_element) from [General-AB-attribute-
menu_item] WHERE main_menu_label =	Group or a New name for the Menu Group is missing"	designer)
:Menus_manipulation_data_current_menu_group	put up OF message box 2 := TRUE END	END SQL INTO Building element
END SQL	, END	action OF ISodbe 1 IS connect := TRUE
FORGET current_menu_B_comp OF	! *** end of method	!
Menus_manipulation_data	END	visible OF main window := FALSE
FORGET CUITERS menu group OF	WITH new_name_of_menu_group STRING	visible OF configuration window := TRUE
Menus_manipulation_data	CT ASS Verms unadersided B come	output OF configuration window := NEW from components display
FORGET Menus_group_items	CLASS Menus_unclassified_B_comp WITH label STRING	END
EXEC Sodbe SQL SELECT distinct	***************************************	WITH another_project_name STRING
(main_menu_label) from [AB-menu_item]	CLASS message box INHERITS add on	WHEN CHANGED
END SQL INTO Menus_group	WITH title STRING	BEGIN
(label)	WITH test STRING	! ! **** this part will do the following
EXEC Sodbe SQL SELECT label from [AB-	WITH variable text STRING WITH modal style COMPOUND	! **** i) make connection to the other project database
menu_item]	app modal	1 **** 2) read the components from the table AB-
END SQL INTO	sys modal,	attribute-designer form the database of the
Menus_classified_B_comp	not modal	! **** other project
POR G = 170	INIT app modal WITH icon COMPOUND	1 **** 3) create tables in the new project database 1 **** 4) insert the tables data into the table AB-
FOR (i := 1 TO number instances OF navigate Building element)	exclamation	attribute-designer of the new table
BEGIN	information,	[**** 5) read the general rules and menus from the old
instance number OF navigate Building	question,	project database and put them
element := i	stop sign	! **** in the database of the new project database
FIND Menus_classified_B_comp	INIT exclamation	! connect to the new project database
WHERE label OF Menus_classified_B_comp = building_element OF	WITH button COMPOUND abort retry ignore,	action OF ISodbe I IS connect := TRUE
Building element	ok,	1
WHEN NONE FOUND	ok cancel,	! make connection with the database of the
MAKE Menus_unclassified_B_comp	retry cancel,	other project
WITH label :≈ building_element OF	yes no, yes no cancel	data source OF 15odbc 3 := another_project_name OF NEW to make a new project
Building element FIND END	INIT ok	action OF ISodbe 3 IS connect := TRUE
END	WITH default button COMPOUND	1
1	one,	! read the components from the other project
! *** end of IF	two,	EXEC ISodbe 3 SQL SELECT distinct
END ELSE	three INIT one	(building_element), graph_name from [AB-attribute- designer]
BEGIN	WITH button selected COMPOUND	END SQL INTO Building element
text OF message box 2 := "A selected Menu	abort,	FOR (i := 1 TO number instances OF navigate
Group is missing"	cancel,	Building element)
put up OF message box 2 := TRUE	ignore,	BEGIN instance number OF navigate Building element
END	no, ok,	:= i
! *** end of method	retry,	! *** create the database table with its id attribut
END	yes	general statement of sql := CONCAT(
WITH rename menu group SIMPLE	WITH put up SIMPLE	"CREATE TABLE (", building_element OF Building
WHEN CHANGED BEGIN	CLASS navigate INHERITS add on	element, "] ([", building_element OF Building element " id] TEXT(50));")
BEGIN !	WITH class attachment CLASS REFERENCE	EXEC ISodbe I SQL WITH general statement
IF CONF(new_name_of_menu_group OF	WITH action COMPOUND	of sql END SQL
Menus_manipulation_data) > 0 AND CONF(lirst,	END
current_menu_group OF Menus_manipulation_data) >	previous,	1 read the attributes of the components from
0 THEN BEGIN	next,	the other project
BEGIN!	WITH instance number NUMERIC	FOR (i := 1 TO number instances OF navigate
FOR (i := 1 TO number instances OF navigate	WITH number instances NUMERIC	Building element)
menu_group_items)	or According to the	BEGIN
' BEGIN	CLASS NEW_building_systems	instance number OF navigate Building element
instance number OF navigate	WITH building_system STRING WITH description STRING	! **** the next step is to overcome the name
menu_group_items := i EXEC !Sodbc ! SQL update [AB-	WITH discipline STRING	"Building element" which can not be linked to Access
menu_item] set		tt := building_element OF Building element
main_menu_tabet =	CLASS NEW_design_disciplines	EXEC ISodbe 3 SQL SELECT * from [AB-
:Menus_manipulation_data.new_name_of_menu_group	WITH discipline STRING	attribute-designer] where building_element = :domain.tt
WHERE label =	CLASS MENU other project components	COTAL TO CATA

```
building_element_attribute
FOR (ii := 1 TO number instances OF
                                                                                                                                                  END
                                                                      building element attribute attribute of building element, building element attribute attribute type,
navigate building element attributes)
          BEGIN
                                                                                      ent attribute designing discipline)
            instance number OF navigate building
                                                                                                                                          t ***** If the name was not used before
element attributes := ii
           *** ignore the attribute that has the _id of the
                                                                                                                        END
                                                                                                                                                IF i = 2 THEN
                                                                      SOL
                                                                                                                                                  REGIN
           IF SEARCHSTR("_id",
                                                                                  END
       ste_of_building_elem
                                                                               END
                                                                                                                                           ! **** create the database table with its id attribute
              ment attribute) = 0 THEN
                                                                             END
                                                                                                                                           general statement of sql := CONCAT(
*CREATE TABLE [*, building_element OF
building_ele
            REGIN
                                                                                                                                          NEW_to_make_a_new_project, "] ([", building_element OF NEW_to_make_a_new_project,
         ! **** put the attributes in the datab
 general statement of sql := CONCAT(
'alter table (", building element OF Building element)
                                                                                 - get the general rules from the old database
                                                                           save them in the new one (ignore the id attribute)
EXEC ISodbe 3 SQL SELECT * from [AB-
                                                                                                                                            _id] TEXT(50);")
                                                                                                                                                  EXEC ISodbe I SQL WITH general statement
"] add [", attribute_of_building_elen
                                                                                                                                          of sal END SOL
building element_attribute, "] ")

IF attribute type OF
                                                                      general-rules]
                                                                                         END SQL INTO AB general_rules
building_element_attribute = "t" THEN
                                                                                                                                          ! **** add the rest of attributes to the created table FOR (ii := 1 TO number instances OF
               general statement of sql := CONCAT(
                                                                           FOR (i := 1 TO number instances OF rules
general statement of sql, "TEXT (50);")
                                                                      navigate 2)
                                                                                                                                          navigate building element attributes)
              IF attribute_type OF
                                                                            BEGIN
                                                                                                                                                    BEGIN
building_element_attribute = "n" THEN
general statement of sql := CONCAT(
                                                                              instance number OF rules navigate 2 :=
                                                                                                                                                     instance number OF navigate building
                                                                              EXEC Isodbe I SQL INSERT INTO [AB-
                                                                                                                                          element attributes := ii
             tement of sql, "Number;")
EXEC ISodbc I SQL WITH general
                                                                           ral-rules] (building_element,
                                                                                                                                                   ! **** ignore the attribute that has the _id of the
                                                                      attribute of building element, affected discipline,
                                                                      percent_change, result_in_major_modification,
cause_of_the_effect) values
statement of sql END SQL
                                                                                                                                                     IF SEARCHSTR("_id",
                                                                                                                                           attribute_of_building_eleme
                                                                      ('AB general rules building element,
'AB general rules stribute of building element,
'AB general rules affected discipline,
                                                                                                                                          building_element_attribute) = 0 THEN
                                                                                                                                                      BEGIN
          END
                                                                                                                                                   ! **** put the attributes in the database
       END
                                                                                                                                                        general statement of sql := CONCAT(
                                                                      :AB_general_rules.percent_change,
:AB_general_rules.result_in_major_modification,
          insert the tables data into the table AB-
                                                                                                                                           alter table [", building element OF Building element,
     buts-designer of the new table
FOR (i = 1 TO number instances OF navigate
                                                                                                                                          "] add [", attribute_of_building_element OF
building_element_attribute, "] ")
                                                                      :AB_general_rules.cause_of_the_effect)
END SQL
                                                                                                                                                       IF attribute_type OF
                                                                            END
                                                                                                                                          building_element_attribute = "t" THEN
       BEGIN
                                                                                                                                                         general statement of sql := CONCAT(
        instance number OF navigate Building element
                                                                                                                                                         ment of sql, "TEXT (50);")
     1 *** the next step is to overcome the name
                                                                                to activate the push bottons
                                                                                                                                          IF attribute_type OF
building_element_attribute = "n" THEN
"Building element" which can not be linked to Access
It := building element OF Building element
                                                                           attribute attachment OF pushbutton 42 := Building
                                                                                                                                                         general statement of sql := CONCAT(
                                                                                                                                             neral statement of sql, "Number;")

EXEC ISodbc 1 SQL WITH general
                                                                           attribute attachment OF pushbutton 43 := Menus attribute attachment OF pushbutton 44 := General
        EXEC iSodbe 3 SQL SELECT * from [AB-
                                                                                                                                                  ent of sql END SQL
attribute-designer] where building element =
                                                                                                                                                      END
                                                                         ••• end of method
                                                                                                                                                    END
                    END SOL INTO
                                                                         END
building element attribute
                                                                       WITH building element STRING WHEN CHANGED
                                                                                                                                             *** add the attributes to the table (AB-attribute-
                                                                                                                                          designer) in the new project
        FOR (ii := 1 TO number instances OF
navigate building element attributes)
BEGIN
                                                                                                                                                  FOR (ii := 1 TO number instances OF
                                                                           EXEC ISodbe 2 SQL SELECT * from [General-
                                                                                                                                          navigate building element attributes)
                                                                                                                                                    BEGIN
                                                                      AB-attribute-designer] where building elem
           instance number OF navigate building
                                                                      :NEW_to_make_s_new_project.build
                                                                                                                                                     instance number OF navigate building
     ent attributes := ii
                                                                                                                ing_element
IF CONF(linked_building_element OF building_element_attribute) = 100 THEN
                                                                                                                                          element attributes := ii

IF CONF(linked_building_element OF
                                                                                         END SOL INTO
                                                                                     ent_attribute
                                                                                                                                          building_element_attribute) = 100 THEN
BEGIN
                                                                         ENE
                                                                       WITH add component to the new project SIMPLE
                                                                        WHEN CHANGED
           ***** the following step is necessary as data
can not be got directly from access to this attribute
                                                                         BEGIN
                                                                                                                                                     ***** the following step is necessary as data
                                                                                                                                          can not be got directly from access to this attribute
EXEC ISodbe 2 SQL SELECT
             EXEC ISodo: 3 SQL SELECT
                                                                           i := 0
(linked_attribute_of_building_element) from [AB-
                                                                           CONF(u) := -2
                                                                                                                                          (linked_attribute_of_building_element) from
attribute-designer)
              where building_element = :domain.tt and
                                                                        *** check that there is an inserted name and that this
                                                                                                                                          [GENERAL-AB-attribute-designer]
                                                                                                                                                       where building elen
attribute_of_building_element =
                                                                      name does not exist
:building_element_attribute.attribute_of_building_elem
                                                                           IF ( CONF(building_element OF
                                                                                                                                          :NEW_to_make_a_new_project.building_element and
                                                                     NEW_to_make_a_new_project) >= 0) THEN
BEGIN
                                                                                                                                          attribute_of_building_element =
                                                                                                                                          :building_element_attribute.attribute_of_building_elem
                    END SOL INTO domain (ttt)
              linked_attribute_of_building_element OF
                                                                                                                                                              END SOL INTO domain (tttt)
building_element_attribute := titt
                                                                              EXEC ISodbc 1 SQL select distinct
                                                                     building_element from [AB-attribute-designer]
                                                                                                                                                        linked_attribute_of_building_element OF
                                                                                                                                          building_element_attribute := tttt
              EXEC ISodbe 1 SQL INSERT INTO (AB-
                                                                                    where building_element =
attribute-designer] (building element,
attribute of building element, attribute_type,
designing discipline, linked_building_element
                                                                     :NEW_to_make_a_new_project.building_ele
END SQL INTO domain (tt)
                                                                                                                                                        EXEC ISodbe 1 SQL INSERT INTO [AB-
                                                                                                                                          attribute-designer] (building element,
attribute_of_building_element, attribute_type,
designing_discipline, linked_building_element,
linked attribute of building element) values
(:domain.tt,
                                                                            END
building element_attribute.attribute_of_building_elem
                                                                                                                                          linked_attribute_of_building_element) values
ent, :building_element_attribute.attribute_type, :building_element_attribute.designing_disciplin
                                                                           IF CONF(n) = 100 THEN
                                                                                                                                           :NEW_to_make_a_new_project.building_ele
ing element attribute.linked building element buildin
                                                                                                                                          :building_element_auribute_attribute_of_building_elem
g_element_attribute.linked_attribute_of_building_elem
                                                                           ELSE
                                                                                                                                          ent, building element attribute.attribute_type,
                                                                                                                                           building element_attribute.designing_discipline.:build
ent )
                                                                                                                                          ing element attribute linked building element building element attribute linked attribute of building elem
   ÉND SQL
            FND
            ELSE
            BEGIN
                                                                           IF i = 1 THEN
              EXEC ISodbe 1 SOL INSERT INTO (AB-
                                                                             BEGIN
              signer] (building_elem
                                                                                                                                                                                             END
                                                                              text OF message box 2 := "There is already a
attribute of building element, attribute type, designing discipline) values
                                                                      building component with that name"
                                                                                                                                          SOL
                                                                              put up OF message box 2 := TRUE
```

```
:NEW_to_make_a_new_project.building_system
END SQL INTO Building element
                                                                                                                                               attribute_of_building_element =
            ELSE
              BEGIN
                                                                                                                                               :building_element_attribute.attribute_of_building_elem
               EXEC ISOMO I SOL INSERT INTO [AB-
attribute-designer] (building element,
attribute of building element, attribute_type,
designing_discipline) values
                                                                             END
                                                                                                                                                                    END SQL INTO domain (ms)
                                                                          WITH ignore highlighted component SIMPLE WHEN CHANGED
                                                                                                                                               linked attribute of building element
OF building element attribute := tttt
                                                                              FORGET CURRENT Building element
                                                                                                                                                                EXEC Isodbe 1 SOL INSERT INTO
                            r_project.building
 NEW to make a me
                                                                                                                                               [AB-attribute-designer] (building element, attribute of building element, attribute type, designing discipline, linked_building element,
building element attribute attribute of building element, building element attribute attribute type,
                                                                             END
                                                                          WITH add system to the new project SIMPLE
 building element attribute designing discipline)
                                                                           WHEN CHANGED
 END SOL
                                                                             BEGIN
                                                                                                                                               linked_attribute_of_building_element) values
             END
                                                                                                                                                              (:domain.tt,
           END
                                                                        ! **** check that there is an inserted same and that this
                                                                                                                                               :building_element_attribute.attribute_of_building_elem
                                                                                                                                               ent, building element attribute attribute type,
       END
                                                                        name does not exist
                                                                                                                                              building element attribute.designing discipline, building element attribute liaked building element, building element attribute.linked_attribute_of_building_elem
                                                                              IF ( CONF(building_system OF
 end of method
                                                                        NEW to make a new project) >= 0) THEN BEGIN
    END
  WITH finish components of the new project SIMPLE WHEN CHANGED
                                                                                                                                              ent )
END SOL
                                                                                                                                                              END
     BEGIN
                                                                                                                                                             FLSE
                                                                                 FOR (i := 1 TO number instances OF navigate
                                                                        Building elem
                                                                                                                                                              BEGIN
 **** this method aims to add the general rules of the
                                                                                  BEGIN
                                                                                                                                                                EXEC ISodbe I SQL INSERT INTO
 chosen components to the new project database
                                                                                    instance number OF navigate Building
                                                                                                                                               [AB-attribute-designer] (building_eleme
                                                                                                                                               attribute_of_building_element, attribute_type,
 **** get the names of the components from the new
                                                                                                                                               designing discipline) values
 project database
      EXEC ISodbe I SQL select distinct
                                                                          **** get the attributes of the building element
                                                                                                                                                              (:domain.tt,
                                                                                                                                               building element attribute attribute of building element, building element attribute attribute type, building element attribute designing discipline)
(building element) from [AB-attribute-designer]
END SQL INTO Building element
                                                                                   tt := building_element OF Building elem
EXEC (Sodbc 2 SQL select * from
                                                                       [GENERAL-AB-attribute-designer] where building element = :domain.tt
                                                                                                                                              END SQL
      FOR (i := 1 TO number instances OF navigate
                                                                                  END SQL INTO building_element_attribute
                                                                                                                                                              END
Building element)
                                                                                                                                                           END
        BEGIN
                                                                        I *** build the tables and the attributes in the new
                                                                                                                                              ! **** end of FOR i
         instance number OF navigate Building element
                                                                       project data! ***** create the database table with its id attribute
                                                                                                                                                        END
                                                                                                                                               end of IF
         tt := building_element OF Building
EXEC 15odbc 2 SQL select * from
                             ment OF Building elen
                                                                        general statement of sql := CONCAT(
"CREATE TABLE [", building element OF Building element, "] [[", building element OF Building element
                                                                                                                                                     END
(GENERAL-AB-general-rules) where
building_element = :domain.tt
END SQL INTO AB_general_rules
                                                                                                                                              ! **** end of method
                                                                        id] TEXT(50));")

EXEC ISodbc I SQL WITH general
                                                                                                                                                  END
                                                                                                                                                WITH finish systems of the new project SIMPLE
                                                                                                                                                  WHEN CHANGED
                                                                                ent of sql END SQL
  **** add the rules to the new project database
                                                                                                                                                   BEGIN
                                                                                                                                              [ **** this method aims to add the general rules of the components of the chosen system to the new project
         FOR (ii := 1 TO number instances OF rules
                                                                           **** add the rest of attributes to the created table
                                                                       FOR (ii = 1 TO number instances OF navigate building element attributes)
navigate 2)
          BEGIN
                                                                                     BEGIN
                     e number OF rules navigate 2 :
                                                                                                                                               ! **** get the names of the components from the new
            EXEC ISodbe 1 SQL INSERT INTO [AB-
                                                                                      instance number OF navigate building
   meral-rules] (building_element
                                                                        element attributes := ii
attribute of building element, affected discipline,
percent change, result in major modification,
                                                                                ! **** ignore the attribute that has the _id of the
                                                                                                                                                    EXEC ISodbc 1 SQL select distinct
                                                                        rable
                                                                                                                                              (building_element) from [AB-attribute-designer]
END SQL INTO Building element
                                                                                      IF SEARCHSTR("_id",
 cause_of_the_effect) values
(:AB_general_rules.building_elemens,
:AB_general_rules.attribute_of_building_elemens,
:AB_general_rules.affected_discipline,
                                                                        attribute_of_building_element OF
                                                                        building_element_attribute) = 0 THEN
                                                                                                                                                    FOR (i := 1 TO number instances OF navigate
                                                                                       BEGIN
                                                                                                                                               Building element)
                                                                                                                                                     BEGIN
                                                                                 ! **** put the attributes in the databa
 :AB general_rules.percent_change,
                                                                         general statement of sql := CONCAT(
'alter table [*, building_element OF Building_element
:AB general rules result in major modification,
:AB general rules cause of the effect)
END SQL
                                                                                                                                                       instance number OF navigate Building element
                                                                        "] add [", attribute_of_building_element OF
                                                                                                                                                        tt := building_element OF Building element
                                                                       building element attribute "] ")

IF attribute type OF
building element attribute = "1" THEN
           END
                                                                                                                                                       EXEC ISodbc 2 SQL select * from
                                                                                                                                               [GENERAL-AB-general-rules] where
                                                                                                                                               building_element = :domain.tt
END SQL INTO AB_general_rules
                                                                        general statement of sql := CONCAT(
general statement of sql, "TEXT (50);")
 sees move to the
                                                                                                                                                **** add the rules to the new project database
 CURRENT_project_selection_display to allow more
                                                                                         IF attribute_type OF
                                                                        building_element_attribute = "n" THEN
 configuration
                                                                                          general statement of sql := CONCAT(
       visible OF configuration window := FALSE
                                                                                                                                                        FOR (ii := 1 TO number instances OF rules
                                                                                         nent of sql, "Number,")
EXEC (Sodbe 1 SQL WITH general
 visible OF main window:= TRUE
output OF main window:=
CURENT_project_selection_display
                                                                                                                                               navigate 2)
                                                                                                                                                         BEGIN
                                                                                                                                                          instance number OF rules navigate 2 := ii
                                                                        statement of sql END SQL
                                                                                        ĖND
                                                                                                                                                          EXEC ISodbe I SQL INSERT INTO [AB-
                                                                                                                                               general-rules] (building_element,
                                                                                     END
 ! **** end of method
     END
                                                                         **** add the attributes to the table [AB-attribute-
                                                                                                                                               attribute of building element, affected discipline,
  WITH design_discipline STRING
                                                                        designer] in the n\
                                                                                                                                               percent_change, result_in_major_modification, cause_of_the_effect) values
    WHEN CHANGED
                                                                        ew project
                                                                                    FOR (ii := 1 TO number instances OF
                                                                                                                                                               (:AB_general_rules.building_element,
     BEGIN
                                                                        navigate building element attributes)
BEGIN
                                                                                                                                               :AB general rules attribute of building element, :AB general rules affected discipline,
      EXEC 15odbc 2 SOL select * from
 building_systems where discipline =
                                                                                       instance number OF navigate building
                                                                                                                                                AB general rules percent change,
                                                                                                                                               :AB general rules result in major modification,
:AB general rules cause of the effect)
       V_to_make_s_new_project.design_discipline
END SQL INTO NEW_building_systems
                                                                         element attributes := ii
 :NEW
                                                                                     - IF CONF(linked building element OF
                                                                        building_element_attribute) = 100 THEN
                                                                                                                                               END SQL
                                                                                                                                                         תעם,
     END
                                                                                        BEĞIN
  WITH building_system STRING
WHEN CHANGED
                                                                                    ***** the following step is necessary as data
                                                                                                                                                      END
                                                                        can not be got directly from access to this attribute
     BEGIN
                                                                                        EXEC ISodbe 2 SQL SELECT
                                                                                                                                                       move to the
                                                                         (linked_attribute_of_building_element) from
                                                                                                                                               CURRENT_project_selection_display to allow more
       EXEC ISodbc 2 SQL SELECT * from
                                                                        [GENERAL-AB-attribute-designer]
where building_element = :domain.tt and
      mponents_of_building_system] where
                                                                                                                                               configuratio
                                                                                                                                                     visible OF configuration window := FALSE
 building system =
```

visible OF main window := TRUE	1	THEN
output OF main window :=	MAKE Building element	BEGIN
CURENT_project_selection_display	WITH building_element := new_component_name	EXEC 15odbc 1 SQL UPDATE [AB-attribute- designer] set graph_name = :current_data.graph_name
**** end of method	1	WHERE building element =
END	MAKE building_element_attribute	current_data.building_element AND
	WITH attribute_of_building_element :=	attribute of building element =
ATTRIBUTE Building Components SIMPLE WHEN CHANGED	new_attribute_name WITH attribute_type := "t"	:current_data.attribute_of_building_element END SQL
BEGIN	Williamions_type := 1	END
action OF ISodbc 1 IS connect := TRUE	building_element OF current_data :=	1
1	new_component_name	!
EXEC ISodbe 1 SQL SELECT distinct	attribute_of_building_element OF	FORGET new_attribute_name
(building_element), graph_name from [AB-attributo- designer]	current_data := new_attribute_name attribute_type OF current_data := "t"	FORGET attribute type OF input data FORGET discipline OF input data
END SQL INTO Building element	!	location OF pushbutton 12 invisible2 := 0,0,0,0
1	END	location OF pushbutton 13 invisible2 := 0,0,0,0
EXEC ISodbe 1 SQL SELECT distinct	END	location OF border 7 invisible2 := 0,0,0,0
(building_element) from [AB-attribute-designer] END SQL INTO Linked building element	! END	location OF textbox 18 invisible2 := 0,0,0,0 location OF textbox 19 invisible2 := 0,0,0,0
!	ATTRIBUTE new_component_name STRING	location OF pushbutton 16 invisible2 := 0,0,0,0
visible OF main window := FALSE	ATTRIBUTE new_attribute_name_iTRING	location OF promptbox 3 invisible2 := 0,0,0,0
output OF configuration window: = config	ATTRIBUTE done of new attribute SIMPLE	1
visible OF configuration window := TRUE	WHEN CHANGED	END
! END	BEGIN	ELSE BEGIN
ATTRIBUTE & STRING	i:=0	text OF message box 2 := "Some essential data
ATTRIBUTE I NUMERIC	1	are missing. Check attribute name, type, and designing
ATTRIBUTE general statement of sql STRING	IF (CONF(new_attribute_name) >= 0 AND	discipline."
ATTRIBUTE done of new component SIMPLE	CONF(attribute_type OF current_data) >= 0 AND	put up OF message box 2 := TRUE
WHEN CHANGED BEGIN	CONF(designing_discipline OF current_data) >= 0) THEN	END END
i := 0	BEGIN	ATTRIBUTE next new attribute SIMPLE
1	1	WHEN CHANGED
! **** check that there is an inserted name and that this	attribute_of_building_element OF current_data	BEGIN
name does not exist IF (CONF(new_component_name) >= 0) THEN	= new_attribute_name attribute_type OF building_element_attribute :=	IF (CONF(new_attribute_name) >= 0 AND CONF(attribute_type OF current_data) >= 0 AND
BEGIN	attribute_type OF current_data	CONF(designing discipline OF current_data) >= 0)
FIND Building element	designing discipline OF	THEN
WHERE building_element OF Building	building_element_attribute := designing_discipline OF	BEGIN
element = new_component_name	CUITER data	
WHEN FOUND i := 1	IF CONF(linked_building_element OF current data) >= 0 THEN	attribute_type OF building_element_attribute := attribute_type OF current_data
WHEN NONE FOUND	BEGIN	designing discipline OF
i:=2	linked_building_element OF	building element_attribute := designing_discipline OF
FIND END	building_element_attribute := linked_building_element	current_data
! IF i = 1 THEN	OF current_data linked_attribute_of_building_element OF	IF CONF(linked_building_element OF current data) >= 0 THEN
BEGIN	building element attribute :=	BEGIN
text OF message box 2 := "There is already a	linked_attribute_of_building_element OF current_data	linked_building_element OF
building component with that name"	END	building_element_attribute := linked_building_element
put up OF message box 2 := TRUE FORGET_new_component_name	! **** put in the database	OF current_data linked_sttribute_of_building_element OF
END -	general statement of sql := CONCAT(*alter	building element_stribute :=
1	table [", building_element OF current_data, "] add [",	linked_attribute_of_building_element OF current_data
! If the name was not used befor	attribute_of_building_element OF current_data, "] ")	END
IF i = 2 THEN BEGIN	IF attribute_type OF current_data = "t" THEN general statement of sql := CONCAT(general	! ron corr
! **** hide the temporary put elements and put the	statement of sql, "TEXT (50);")	FORGET new_attribute_name FORGET attribute_of_building_element OF
temporary elements for ID data	IF attribute_type OF current_data = "n" THEN	current data
location OF border 6 invisible := 0,0,0,0	general statement of sql := CONCAT(general	FORGET attribute_type OF current_data
location OF textbox 17 invisible := 0,00,0	statement of sql, "Number,")	FORGET designing_discipline OF current_data
location OF promptbox 2 invisible := 0,0,0,0 location OF pushbutton 10 invisible := 0,0,0,0	EXEC ISodbe 1 SQL WITH general statement of sql END SQL	FORGET linked_building_element OF
location OF pushbutton 11 invisible:= 0,0,0,0	1 -4 m m 4m	Current_data FORGET linked attribute of building element
location OF pushbutton 26 invisible2 :=	EXEC ISodbe 1 SQL INSERT INTO (AB-	OF current_data
25,265,220,285	attribute-designer] (building_element,	FORGET attribute type OF input data
location OF promptbox 3 invisible2 := 25,195,220,220	attribute_of_building_element, attribute_type, designing_discipline) values	FORGET discipline OF input data location OF pushbutton 16 invisible2 :=
location OF border 7 invisible2 :=	(:current data building element,	180,195,220,220
15,150,235,290	current_data_attribute_of_building_element.	location OF promptbox 3 invisible2 :=
location OF textbox 19 invisible2 :=	:current_data.attribute_type,	25,195,220,220
25,225,220,260	:current_data.designing_discipline) END SQL	location OF textbox 18 invisible2 :=
! **** create the database table with its id attribute	END 3QL	25,155,220,190
general statement of sql := CONCAT(IF i = 1 THEN	END
"CREATE TABLE [", new_component_name, "] ([",		
	BEGIN	ELSE
new_component_name, "_id] TEXT(50));") EYEC (Sodbe 1 SO) WITH general statement	EXEC Sodbc SQL UPDATE [AB-attribute-	BEGIN
EXEC ISodbc 1 SQL WITH general statement	EXEC ISodbe 1 SQL UPDATE [AB-attribute- designer] set	BEGIN text OF message box 2 := "Some essential data
	EXEC Sodbc SQL UPDATE [AB-attribute-	BEGIN
EXEC Isodbc 1 SQL WITH general statement of aql END SQL !!	EXEC I5odbe I SQL UPDATE [AB-attribute- designer] set linked_building_element = :current_data_linked_building_element, linked_attribute_of_building_element =	BEGIN text OF message box 2 := "Some essential data are missing. Check attribute name, type, and designing
EXEC Isodbc 1 SQL WITH general statement of sql END SQL ! ! **** add the information to [AB-attribute-designer] table	EXEC ISodbe I SQL UPDATE [AB-attribute-designer] set linked_building_element = :current_data_linked_building_element, linked_stribute_of_building_element = :current_data_linked_attribute_of_building_element	BEGIN text OF message box 2 := "Some essential data are missing. Check attribute name, type, and designing discipline." put up OF message box 2 := TRUE END
EXEC ISodbc 1 SQL WITH general statement of sql END SQL ! ! *** add the information to [AB-attribute-designer] table new_attribute_name := CONCAT(EXEC ISodbe 1 SQL UPDATE [AB-attribute-designer] set linked building element = :current_data.linked_building_element = :current_data.linked_building_element = :current_data.linked_attribute_of_building_element WHERE building_element = :	BEGIN text OF message box 2 := "Some essential data are missing. Check attribute name, type, and designing discipline." put up OF message box 2 := TRUE END END
EXEC Isodbc 1 SQL WITH general statement of sql END SQL ! ! **** add the information to [AB-attribute-designer] table new_attribute_name := CONCAT(new_component_name, "_id")	EXEC ISodbe I SQL UPDATE [AB-attribute-designer] set linked_building_element = :current_data_linked_building_element, linked_attribute_of_building_element = :current_data_linked_attribute_of_building_element WHERE building_element = :current_data_building_element AND	BEGIN text OF message box 2 := "Some essential data are missing. Check attribute name, type, and designing discipline." put up OF message box 2 := TRUE END END ATTRIBUTE ok of attribute name SIMPLE
EXEC Isodbe 1 SQL WITH general statement of sql END SQL ! "" add the information to [AB-attribute-designer] table new_attribute_name := CONCAT(new_component_name, "_id") EXEC Isodbe 1 SQL INSERT INTO [AB- attribute-designer] (building_element,	EXEC ISodbe 1 SQL UPDATE [AB-attribute-designer] set linked building element = :current_data.linked_building_element = :current_data.linked_building_element = :current_data.linked_attribute_of_building_element WHERE building_element = :	BEGIN text OF message box 2 := "Some essential data are missing. Check attribute name, type, and designing discipline." put up OF message box 2 := TRUE END END
EXEC Isodbc 1 SQL WITH general statement of sql END SQL ! ! **** add the information to [AB-attribute-designer] table new_attribute_name := CONCAT(new_component_name, "_id") EXEC Isodbc 1 SQL INSERT INTO [AB- attribute-designer] (building_element, antribute_of_building_element) values	EXEC ISodbe 1 SQL UPDATE [AB-attribute-designer] set linked_building_element == :current_data_linked_building_element == :current_data_linked_attribute_of_building_element == :current_data_linked_attribute_of_building_element == :current_data_building_element AND attribute_of_building_element == :current_data_attribute_of_building_element == :current_data_attribute_of_building_element END SQL	BEGIN text OF message box 2 := "Some essential data are missing. Check attribute name, type, and designing discipline." put up OF message box 2 := TRUE END END ATTRIBUTE ok of attribute name SIMPLE WHEN CHANGED BEGIN IF (CONF(new_attribute_name) >= 0) THEN
EXEC Isodbe 1 SQL WITH general statement of sql END SQL ! "" add the information to [AB-attribute-designer] table new_attribute_name := CONCAT(new_component_name, "_id") EXEC Isodbe 1 SQL INSERT INTO [AB- attribute-designer] (building_element,	EXEC ISodbe 1 SQL UPDATE [AB-attribute-designer] set linked building element = :current data.linked building element = icurrent data.linked attribute of building element = :current data.linked attribute of building element WHERE building element = :current data.building element AND attribute of building element = :current data.attribute_of_building_element	BEGIN text OF message box 2 := "Some essential data are missing. Check attribute name, type, and designing discipline." put up OF message box 2 := TRUE END END ATTRIBUTE ok of attribute name SIMPLE WHEN CHANGED BEGIN

WHERE attribute of building element OF	:AB_general_rules.perce
building element_stribute = new attribute name	:AB_general_rules.resul
WHEN FOUND	:AB_general_rules.rule
i:=1	:AB_general_rules.caus
WHEN NONE FOUND	END SOL
i:=2	END
FIND END	t
1	END
IF i = 2 THEN	ELSE
BEGIN	BEGIN
MAKE building_element_attribute	text OF message b
WITH attribute_of_building_element :=	art missing.
new_attribute_name	You may need to puch h
location OF pushbutton 12 invisible2 :=	put up OF messag
25,265,225,285	END
location OF pushbutton 16 invisible2 := 0,0,0,0	END ATTRIBUTE new rule S
END	WHEN CHANGED
END	BEGIN
1	
! **** end of method	IF (CONF(building Current_rule_data) >= 0
END .	attribute of building el
ATTRIBUTE General Rules SIMPLE	>= 0) THEN
WHEN CHANGED	BEGIN
BEGIN	FORGET discipli
Ţ.	MAKE AB_gener
action OF ISodbc 1 IS connect := TRUE	WITH building
!	OF Current_rule_data
EXEC ISodoc 1 SQL SELECT distinct	WITH attribute_
(building_element) from [AB-attribute-designer]	attribute_of_building_ele
END SQL INTO Building element	END
1	END
visible OF main window := FALSE	ATTRIBUTE delete a co
output OF configuration window := General rules	WHEN CHANGED
display	BEGIN
visible OF configuration window := TRUE	IF CONF(building_o
! ***** end of method	BEGIN
END	general statement
ATTRIBUTE update rule SIMPLE	table ", building_elemen
WHEN CHANGED	EXEC ISodbe 1 SC
BEGIN	sql END SQL
IF (CONF(building_element OF	EXEC ISodbe 1 Se
AB general rules) >= 0 AND CONF(designer) where building
attribute_of_building_element OF AB_general_rules)	:current_data.building_e
>= 0 AND CONF(affected_discipline OF	EXEC (Sodbe 1 So
AB_general_rules) >= 0) THEN	(building_element), grap
BEGIN	designer]
IF CONF(AB_general_rules_ID OF	END SQL IN
AB_general_rules) >= 0 THEN	EXEC ISodbe 1 SC
BEGIN	(building_element) from
EXEC ISodbc 1 SQL update [AB-general-	END SQL IN
rules) set	FORGET current_
affected_discipline =	FORGET building
:AB_general_rules.affected_discipline, percent_change = :AB_general_rules.percent_change,	ELSE
result_in_major_modification =	BEGIN
:AB_general_rules.result_in_major_modification,	text OF message b
rule_ignored = :AB_general_rules.rule_ignored.	Component is currently of
cause of the effect =	put up OF message
:AB_general_rules.cause_of_the_effect	END
where AB_general_rules_ID =	ŧ
:AB_general_rules.AB_general_rules_ID	! **** end of method
:AB_general_rules.AB_general_rules_ID END SQL	end of method END
:AB_general_rules.AB_general_rules_ID END SQL END	i **** end of method END ATTRIBUTE done of II
:AB_general_rules.AB_general_rules_ID END SQL END END ELSE	! **** end of method END ATTRIBUTE done of ID WHEN CHANGED
:AB_general_rules.AB_general_rules_ID END SQL END ELSE BEGIN	I **** end of method END ATTRIBUTE done of ID WHEN CHANGED BEGIN
:AB_general_rules.AB_general_rules_ID END SQL END ELSE BEGIN IF CONF(percent_change OF	i **** end of method END ATTRIBUTE done of ID WHEN CHANGED BEGIN
:AB_general_rules.AB_general_rules_ID END SQL END ELSE BEGIN IF CONF(percent_change OF AB_general_rules) < 0 THEN	i **** end of method END ATTRIBUTE done of ID WHEN CHANGED BEGIN ! i := 0
:AB_general_rules.AB_general_rules_ID END SQL END ELSE BEGIN IF CONF(percent_change OF AB_general_rules) < 0 THEN percent_change OF AB_general_rules := 0	end of method END ATTRIBUTE done of II WHEN CHANGED BEGIN ! i := 0
:AB_general_rules.AB_general_rules_ID END SQL END ELSE BEGIN IF CONF(percent_change OF AB_general_rules) < 0 THEN percent_change OF AB_general_rules := 0 IF CONF(result_in_major_modification OF	end of method END ATTRIBUTE done of IL WHEN CHANGED BEGIN i := 0 If (CONF(new_attri
:AB_general_rules.AB_general_rules_ID END SQL END SQL END ELSE BEGIN IF CONF(percent_change OF AB_general_rules) < 0 THEN percent_change OF AB_general_rules := 0 IF CONF(result_in_major_modification OF AB_general_rules) < 0 THEN	! **** end of method END ATTRIBUTE done of II WHEN CHANGED BEGIN ! i := 0 ! IF (CONF(new_attri CONF(attribute_type 0)
:AB_general_rules.AB_general_rules_ID END SQL END ELSE BEGIN IF CONF(percent_change OF AB_general_rules) < 0 THEN percent_change OF AB_general_rules := 0 IF CONF(result, in_major_modification OF AB_general_rules) < 0 THEN result, in_major_modification OF	end of method END ATTRIBUTE done of IL WHEN CHANGED BEGIN i := 0 If (CONF(new_attri
:AB_general_rules.AB_general_rules_ID END SQL END ELSE BEGIN IF CONF(percent_change OF AB_general_rules) < 0 THEN percent_change OF AB_general_rules := 0 IF CONF(result_in_major_modification OF AB_general_rules) < 0 THEN result_in_major_modification OF AB_general_rules: == "No"	! *** end of method END ATTRIBUTE done of II WHEN CHANGED BEGIN ! i := 0 ! IF (CONF(new_attri CONF(attribute_type 0) CONF(designing_discip
:AB_general_rules.AB_general_rules_ID END SQL END ELSE BEGIN IF CONF(percent_change OF AB_general_rules) < 0 THEN percent_change OF AB_general_rules := 0 IF CONF(result, in_major_modification OF AB_general_rules) < 0 THEN result, in_major_modification OF	end of method END ATTRIBUTE done of II WHEN CHANGED BEGIN i := 0 [CONF(attribute_type Oi CONF(designing_discip THEN
:AB_general_rules.AB_general_rules_ID END SQL END SQL END ELSE BEGIN IF CONF(percent_change OF AB_general_rules) < 0 THEN percent_change OF AB_general_rules := 0 IF CONF(result in_major_modification OF AB_general_rules) < 0 THEN result in_major_modification OF AB_general_rules := "No" IF CONF(rule_ignored OF AB_general_rules) < 0 THEN rule_ignored OF AB_general_rules := "No"	end of method END ATTRIBUTE done of II WHEN CHANGED BEGIN i = 0 f (CONF(new_attri CONF(attribute_type Of THEN BEGIN attribute_type OF
:AB_general_rules.AB_general_rules_ID END SQL END SQL END FLSE BEGIN IF CONF(percent_change OF AB_general_rules) < 0 THEN percent_change OF AB_general_rules := 0 IF CONF(result, in_major_modification OF AB_general_rules) < 0 THEN result, in_major_modification OF AB_general_rules := "No" IF CONF(rule_ignored OF AB_general_rules) < 0 THEN rule_ignored OF AB_general_rules := "No" IF CONF(cause_of_the_effect OF	end of method END ATTRIBUTE done of IE WHEN CHANGED BEGIN i = 0 IF (CONF(new_attri CONF(attribute_type Of BEGIN BEGIN attribute_type OF attribute_type OF attribute_type OF
:AB_general_rules.AB_general_rules_ID END SQL END SQL END ELSE BEGIN IF CONF(percent_change OF AB_general_rules) < 0 THEN percent_change OF AB_general_rules := 0 IF CONF(result_in_major_modification OF AB_general_rules) < 0 THEN result_in_major_modification OF AB_general_rules := "No" IF CONF(rule_ignored OF AB_general_rules) < 0 THEN rule_ignored OF AB_general_rules := "No" IF CONF(cause_of_the_effect OF AB_general_rules) < 0 THEN AB_general_rules) < 0 THEN	end of method END ATTRIBUTE done of II WHEN CHANGED BEGIN i = 0 IF (CONF(new_attri CONF(attribute_type O) CONF(designing_discip THEN BEGIN attribute_type OF curren designing_discipli
:AB_general_rules.AB_general_rules_ID END SQL END SQL END ELSE BEGIN IF CONF(percent_change OF AB_general_rules) < 0 THEN percent_change OF AB_general_rules := 0 IF CONF(result_in_major_modification OF AB_general_rules) < 0 THEN result_in_major_modification OF AB_general_rules := "No" IF CONF(rule_ignored OF AB_general_rules) < 0 THEN rule_ignored OF AB_general_rules := "No" IF CONF(cause_of_the_effect OF AB_general_rules) < 0 THEN cause_of_the_effect OF AB_general_rules :=	end of method END ATTRIBUTE done of II WHEN CHANGED BEGIN i = 0 IF (CONF(new_attri CONF(attribute_type Of CONF(designing_discip THEN BEGIN attribute_type OF curren designing_discipli building_element_attribu
:AB_general_rules.AB_general_rules_ID END SQL END SQL END ELSE BEGIN IF CONF(percent_change OF AB_general_rules) < 0 THEN percent_change OF AB_general_rules := 0 IF CONF(result_in_major_modification OF AB_general_rules) < 0 THEN result_in_major_modification OF AB_general_rules := "No" IF CONF(rule_ignored OF AB_general_rules) < 0 THEN rule_ignored OF AB_general_rules := "No" IF CONF(cause_of_the_effect OF AB_general_rules) < 0 THEN cause_of_the_effect OF AB_general_rules := "No"	emo end of method END ATTRIBUTE done of IE WHEN CHANGED BEGIN i = 0 IF (CONF(new_attri CONF(attribute_type Of CONF(designing_discip THEN BEGIN attribute_type OF attribute_type OF attribute_type OF curren designing_discipli building_element_attribute_tyreta
:AB_general_rules.AB_general_rules_ID END SQL END SQL END ELSE BEGIN IF CONF(percent_change OF AB_general_rules) < 0 THEN percent_change OF AB_general_rules := 0 IF CONF(result_in_major_modification OF AB_general_rules) < 0 THEN result_in_major_modification OF AB_general_rules := "No" IF CONF(rule_ignored OF AB_general_rules) < 0 THEN rule_ignored OF AB_general_rules := "No" IF CONF(rule_ignored OF AB_general_rules) < 0 THEN cause_of_the_effect OF AB_general_rules := "No" EXEC ISodbc I SQL insert into [AB-general-rules]	end of method END ATTRIBUTE done of II WHEN CHANGED BEGIN i = 0 IF (CONF(new_attri CONF(attribute_type O) CONF(designing_discip THEN BEGIN attribute_type OF curren designing_discipli building_element_attribute_current_data IF CONF(linked_
:AB_general_rules.AB_general_rules_ID END SQL END ELSE BEGIN IF CONF(percent_change OF AB_general_rules) < 0 THEN percent_change OF AB_general_rules := 0 IF CONF(result in_major_modification OF AB_general_rules) < 0 THEN result_in_major_modification OF AB_general_rules := "No" IF CONF(rule_ignored OF AB_general_rules) < 0 THEN rule_ignored OF AB_general_rules := "No" IF CONF(cause_of_the_effect OF AB_general_rules) < 0 THEN cause_of_the_effect OF AB_general_rules := "EXEC ISodbc 1 SQL insert into [AB-general_rules]	end of method END ATTRIBUTE done of IE WHEN CHANGED BEGIN i = 0 IF (CONF(new_attri CONF(attribute_type OF CONF(designing_discip THEN BEGIN attribute_type OF curren designing_discipli building_element_attribucurrent_data IF CONF(linked_current_data) >= 0 THEI
:AB_general_rules.AB_general_rules_ID END SQL END ELSE BEGIN IF CONF(percent_change OF AB_general_rules) < 0 THEN percent_change OF AB_general_rules := 0 IF CONF(result in_major_modification OF AB_general_rules) < 0 THEN result in_major_modification OF AB_general_rules := "No" IF CONF(rule_ignored OF AB_general_rules) < 0 THEN rule_ignored OF AB_general_rules := "No" IF CONF(cause_of_the_effect OF AB_general_rules) < 0 THEN cause_of_the_effect OF AB_general_rules := EXEC ISodbc I SQL insert into [AB-general-rules] (building_element, attribute_of_building_element,	emo end of method END ATTRIBUTE done of IE WHEN CHANGED BEGIN i = 0 IF (CONF(new_attri CONF(attribute_type Of CONF(designing_discip THEN BEGIN attribute_type OF curren designing_discipli building_element_attribute_transpecturent_data IF CONF(linked_current_data) >= 0 THEI BEGIN
:AB_general_rules.AB_general_rules_ID END SQL END SQL END FLSE BEGIN IF CONF(percent_change OF AB_general_rules) < 0 THEN percent_change OF AB_general_rules := 0 IF CONF(result_in_major_modification OF AB_general_rules) < 0 THEN result_in_major_modification OF AB_general_rules := "No" IF CONF(rule_ignored OF AB_general_rules) < 0 THEN rule_ignored OF AB_general_rules := "No" IF CONF(rule_ignored OF AB_general_rules) < 0 THEN cause_of_the_effect OF AB_general_rules := "EXEC ISodbe I SQL insert into [AB-general-rules] (building_element_attribute_of_building_element_affected_discipline, percent_change,	end of method END ATTRIBUTE done of II WHEN CHANGED BEGIN i = 0 IF (CONF(new_attri CONF(attribute_type O) CONF(designing_discip THEN BEGIN attribute_type OF curren designing_discipli building_element_attribute_type of current_data IF CONF(linked_current_data) BEGIN inked_building
:AB_general_rules.AB_general_rules_ID END SQL END SQL END ELSE BEGIN IF CONF(percent_change OF AB_general_rules) < 0 THEN percent_change OF AB_general_rules := 0 IF CONF(result_in_major_modification OF AB_general_rules) < 0 THEN result_in_major_modification OF AB_general_rules := "No" IF CONF(rule_ignored OF AB_general_rules) < 0 THEN rule_ignored OF AB_general_rules := "No" IF CONF(cause_of_the_effect OF AB_general_rules) < 0 THEN cause_of_the_effect OF AB_general_rules := EXEC ISodbc I SQL insert into [AB-general-rules] (building_element_attribute_of_building_element, affected_discipline, percent_change, result_in_major_modification,	end of method END ATTRIBUTE done of IE WHEN CHANGED BEGIN i = 0 IF (CONF(new_attri CONF(attribute_type Of CONF(designing_discip THEN BEGIN attribute_type OF curren designing_discipli building_element_attribu current_data IF CONF(linked current_data) >= 0 THEI BEGIN linked_building_element_attribu building_element_attribu
:AB_general_rules.AB_general_rules_ID END SQL END ELSE BEGIN IF CONF(percent_change OF AB_general_rules) < 0 THEN percent_change OF AB_general_rules := 0 IF CONF(result_in_major_modification OF AB_general_rules) < 0 THEN result_in_major_modification OF AB_general_rules := "No" IF CONF(rule_ignored OF AB_general_rules) < 0 THEN rule_ignored OF AB_general_rules := "No" IF CONF(cause_of_the_effect OF AB_general_rules) < 0 THEN cause_of_the_effect OF AB_general_rules := EXEC ISodbc I SQL insert into [AB-general-rules] (building_element_attribute_of_building_element, affected_discipline_percent_change, result_in_major_modification, rule_ignored_cause_of_the_effect)	emed of method END ATTRIBUTE done of IE WHEN CHANGED BEGIN i = 0 IF (CONF(new_attri CONF(attribute_type Of CONF(designing_discip THEN BEGIN attribute_type OF curren designing_discipli building_element_attribute_trent_data IF CONF(linked_current_data) >= 0 THEI BEGIN linked_building building_element_attribute_trent_data OF current_data OF current_data OF current_data
:AB_general_rules.AB_general_rules_ID END SQL END SQL END FLSE BEGIN IF CONF(percent_change OF AB_general_rules) < 0 THEN percent_change OF AB_general_rules := 0 IF CONF(result_in_major_modification OF AB_general_rules) < 0 THEN result_in_major_modification OF AB_general_rules := "No" IF CONF(rule_ignored OF AB_general_rules) < 0 THEN rule_ignored OF AB_general_rules := "No" IF CONF(cause_of_the_effect OF AB_general_rules) < 0 THEN cause_of_the_effect OF AB_general_rules := EXEC ISodbc I SQL insert into [AB-general_rules] (building_element_attribute_of_building_element, affected_discipline, percent_change, result_in_major_modification, rule_ignored_cause_of_the_effect) values	end of method END ATTRIBUTE done of IE WHEN CHANGED BEGIN i = 0 IF (CONF(new_attri CONF(attribute_type O) CONF(designing_discip THEN BEGIN attribute_type OF curren designing_discipli building_element_attribute_trent_data IF CONF(linked_current_data) = 0 THEI BEGIN linked_building building_element_attribute Corrent_data Linked_building_building_element_attribute linked_attribute linked_attribute
:AB_general_rules.AB_general_rules_ID END SQL END ELSE BEGIN IF CONF(percent_change OF AB_general_rules) < 0 THEN percent_change OF AB_general_rules := 0 IF CONF(result_in_major_modification OF AB_general_rules) < 0 THEN result_in_major_modification OF AB_general_rules := "No" IF CONF(rule_ignored OF AB_general_rules) < 0 THEN rule_ignored OF AB_general_rules := "No" IF CONF(cause_of_the_effect OF AB_general_rules) < 0 THEN cause_of_the_effect OF AB_general_rules := EXEC ISodbc I SQL insert into [AB-general-rules] (building_element_attribute_of_building_element, affected_discipline_percent_change, result_in_major_modification, rule_ignored_cause_of_the_effect)	emed of method END ATTRIBUTE done of IE WHEN CHANGED BEGIN i = 0 IF (CONF(new_attri CONF(attribute_type Of CONF(designing_discip THEN BEGIN attribute_type OF curren designing_discipli building_element_attribute_trent_data IF CONF(linked_current_data) >= 0 THEI BEGIN linked_building building_element_attribute_trent_data OF current_data OF current_data OF current_data

```
END
                                  t_in_major_modification,
                                                                                  ! **** put in the database
                                 ignored.
                                 e_of_the_effect)
                                                                                          EXEC ISodbc 1 SQL UPDATE [AB-attribute-
                                                                                  designer] set
                                                                                   attribute_type = :current_data.attribute_type,
                                                                                    designing_discipline =
                                                                                  current_data_designing_discipline
WHERE building_element =
:current_data_building_element_AND
                                 ox 2 := "Some important data
                                                                                  attribute_of_building_element = :current_data_attribute_of_building_element
                                  iew Rule botton"
                                                                                           END SQL
                                 e box 2 := TRUE
                                                                                          IF i = I THEN
                                                                                           BEGIN
                                 SIMPLE
                                                                                             EXEC ISodbc | SQL UPDATE [AB-attribute-
                                                                                  designer) set
                                                                                  linked_building_element =
.current_data.linked_building_element,
                                 element OF
AND CONF(
                                                                                   linked attribute of building element
                                  ement OF Current_rule_data)
                                                                                  current_data.linked_attribute_of_building_element
                                                                                  WHERE building element =: current data building element
                                                                                  :current_data.building_element_AND
attribute_of_building_element =
:current_data.attribute_of_building_element
END SOV
                                 ne OF input data
                                 al_rules
element := building_elem
                                                                                           END SQL
                                 of_building_element :=
                                 ement OF Current_rule_data
                                                                                          IF CONF(graph_name OF current_data) >= 0
                                                                                  THEN
                                                                                           BEGIN
                                 emponent SIMPLE
                                                                                             EXEC Bodbe I SQL UPDATE [AB-attribute-
                                                                                  designer set graph, name = :current_data_graph_name
WHERE building_element =
:current_data.building_element AND
attribute_of_building_element =
:current_data.stribute_of_building_element
                                 element OF current_data) >= 0
                                 of aql := CONCAT( "drop
                                                                                           END SQL
                                 t OF current_data)
QL WITH general sta
                                                                                           END
                                                                                          FORGET new_attribute_na
                                                                                          FORGET attribute type OF input data
FORGET discipline OF input data
                                 QL delete from (AB-attribute-
                                                                                          location OF pushbutton 12 invisible2 := 0,0,0,0 location OF pushbutton 13 invisible2 := 0,0,0,0
                                  ement END SQL
                                 OL SELECT distinct
                                                                                          location OF prompthox 3 invisible2 := 0,0,0,0 location OF border 7 invisible2 := 0,0,0,0
                                 h name from (AB-attribute-
                                                                                          location OF textbox 18 invisible2 := 0,0,0,0 location OF textbox 19 invisible2 := 0,0,0,0 location OF textbox 19 invisible2 := 0,0,0,0
                                 ITO Building element
QL SELECT distinct
                                 [AB-attribute-designer]
TO Linked building element
                                                                                          location OF pushbutton 16 invisible2 := 0,0,0,0
                                                                                          location OF pushbutton 26 invisible2 := 0,0,0,0
                                 data
                                  _element_attribute
                                                                                        END
                                                                                      FI SE
                                                                                        BEGIN
                                                                                          text OF message box 2 := "Some essential data
                                 ox 2 := "No Building
                                                                                  are missing. Check attribute name, type, and designing
                                                                                  discipline.
                                                                                       put up OF message box 2 := TRUE END
                                  box 2 := TRUE
                                                                                     END
                                                                                  ATTRIBUTE delete of attribute SIMPLE WHEN CHANGED
                                 attribute SIMPLE
                                                                                     BEGIN
                                                                                  IF CONF(aitribute_of_building_element OF current_data) >= 0 THEN
BEGIN
                                                                                   *** delete from its table and from table [AB-attribute-
                                 bute_name) >= 0 AND
                                                                                  designer]
                                                                                 oenger]
general statement of sql := CONCAT( "alter
table [", building_element OF current_data, "] drop [",
attribute_of_building_element OF current_data, "];")
EXEC [Sodbe 1 SQL WITH general statement of
                                 F current_data) >= 0 AND bline OF current_data) >= 0)
                                                                                  sql END SQL
                                 building_element_attribute :=
                                                                                          EXEC ISodbc 1 SQL delete from [AB-attribute-
                                                                                  designer] where building_element = :current_data_building_element AND attribute_of_building_element =
                                 t_data
ne OF
                                 nte := designing_discipline OF
                                                                                  current_data attribute_of_building_element_END SQL
                                 building_element OF
                                                                                        * delete the related data from Level 5
                                                                                          FIND building_element_attribute
WHERE attribute_of_building_element OF
                                  element OF
                                                                                  building element attribute = attribute of building element OF current data WHEN FOUND
                                 ne := linked_building_element
                                  _of_building_element OF
                                                                                             FORGET CURRENT
linked_attribute_of_building_element OF current_data
                                                                                 building_element_attribute
FIND END
           i := 1
```

FORGET attribute_of_building_element OF	output OF main window:=	(*.kib *)
current_data FORGET attribute_type OF current_data	CURENT_project_selection_display END	action OF file 2 IS close := TRUE
FORGET designing discipline OF current_data	ATTRIBUTE	make a copy of the empty database
FORGET linked building element OF	check_the_existance_of_new_project_name SIMPLE	ACTIVATE "IPU, EXTERN, project bat"
current_data	WHEN CHANGED	1
FORGET linked_attribute_of_building_element	BEGIN	! connect the display to the window
OF current_data END	FIND Current_projects	output OF main window:
ELSE	WHERE project_name OF Current_projects = project_name	NEW_project_selection_display
BEGIN	WHEN FOUND	*** end of method
text OF message box 2 := "No Building	text OF message box 2 := "This name already	END
Component Attribute is currently chosen	exists for a current project.	ATTRIBUTE path_of_odbc STRING
put up OF message box 2 := TRUE END	Type another name please. * put up OF message box 2 := TRUE	ATTRIBUTE current_path STRING
!	WHEN NONE FOUND	ATTRIBUTE III NUMERIC ATTRIBUTE IIII STRING
! **** end of method	setup a new project := TRUE	ATTRIBUTE get from general database SIMPLE
END	FIND END	WHEN CHANGED
ATTRIBUTE Menus SIMPLE	! end of method	BEGIN
WHEN CHANGED BEGIN	END ATTRIBUTE setup a new project SIMPLE	
1	WHEN CHANGED	EXEC 15odbc 2 SQL select distinct
action OF ISodbc 1 IS connect := TRUE	BEGIN	(building_element) from [General-AB-attribute-
1	1	designer)
EXEC ISodbe 1 SQL SELECT distinct	1 ***** this method does the following:	END SQL INTO
(building_element) from [AB-attribute-designer] END SQL INTO Building element	! ***** 1) Read the paths of the ODBC.ini file and the path of the directorey	GENERAL_Building_element
EAD 205 HATO bending element	! ***** that is used for this program and contains an	! 1 **** remove the building elements that are already in
EXEC 15odbe 1 SQL SELECT distinct	"empty" database	use
(main_menu_label) from [AB-menu_item]	! **** 2) Modify the ODBC.INI file so it links to the	FOR (i := 1 TO number instances OF navigate
END SQL INTO Menus_group (label)	new database	Building element)
EXEC ISodbe 1 SQL SELECT label from [AB-	! ***** 3) Add the file name to the General database so	BEGIN
menu_item]	as to be considered existing project ! ***** 4) Prepare a batch file to exceute the next step	instance number OF navigate Building element
END SQL INTO	! **** 5) Make a new database by copy the empty	FIND GENERAL Building element
Metsus_classified_B_comp	database to a file with	WHERE building element OF
1 707 5 170 1 1 1 07 1	! ***** the new name	GENERAL Building element = building element OF
FOR (i = 1 TO number instances OF navigate Building element)	 	Building element
BEGIN	! read the path of the files from the file	WHEN FOUND FORGET CURRENT
instance number OF navigate Building element	project.ini -	GENERAL Building element
æi	action OF file 1 IS open old := TRUE	FIND END
FIND Menus_classified_B_comp	WHILE (NOT cof OF file 1)	END
WHERE label OF Menus_classified_B_comp = building_element OF Building element	BEGIN	in the OR are for a second database with the second
WHEN NONE FOUND	read line OF file 1 := TRUE IF SUBSTR(current line OF file 1, 1, 16) =	visible OF get from general database window :== TRUE
MAKE Menus_unclassified_B_comp	"path of ODBC.ini" THEN	output OF get from general database window := add
WITH label := building_element OF	path_of_odbc := SUBSTR(current line OF file	from general database display
Building element	1, 36, 11)	
FIND END END	IF SUBSTR(current line OF file 1, 1, 29) =	! **** end of method
!	"path of the developed program" THEN current_path := SUBSTR(current line OF file	END
visible OF main window := FALSE	I, 36, 16)	ATTRIBUTE GENERAL_Building_component STRING
output OF configuration window := Menus display	END	WHEN CHANGED
visible OF configuration window := TRUE	action OF file I IS close: TRUE	BEGIN
! ! *** end of method	I If the ODDC int	EXEC iSodbc 2 SQL SELECT * from [General-
END .	1 modify the ODBC.ini filename OF file 3 := CONCAT(path_of_odbc,	AB-attribute-designer] where building_element = :domain.GENERAL Building component
ATTRIBUTE project_name STRING	"ODBC.INI")	END SQL INTO
WHEN CHANGED	action OF file 3 IS open old := TRUE	GENERAL_building_element_attribute
BEGIN	WHILE (NOT eof OF file 3)	END
data source OF ISodbc 1 := project_name title OF configuration window := CONCAT(BEGIN read line OF file 3 := TRUE	ATTRIBUTE add component from _get from general
"Configuring Project : ", project_name)	END END	datab SIMPLE WHEN CHANGED
attribute attachment OF pushbutton 38 :=	write line OF file 3 := CONCAT("[",	BEGIN
check_the_existance_of_new_project_name	project_name, "]")	i := 0
display attachment OF pushbutton 37 := CURENT_project_selection_display	write line OF file 3 := CONCAT("Driver",	FIND Building element
END	puth_of_odbc, "SYSTEM\ODBCJT16.DLL")	WHERE LOWCASE(building_element OF
ATTRIBUTE New project SIMPLE	write line OE 61a 1 := CONCAT(*DDO=*	
VI IVIDO I E New Diolect STATETE	write line OF file 3 := CONCAT("DBQ=", project name. ".MDB")	Building element) = LOWCASE(
WHEN CHANGED	write line OF file 3 := CONCAT("DBQ=", project_name, ".MDB") write line OF file 3 := CONCAT("DefaultDir=",	
WHEN CHANGED BEGIN	project_name, ".MDB") write line OF file 3 := CONCAT("DefaultDir=", current_path)	Building element) = LOWCASE(GENERAL_Building_component) WHEN FOUND i := 1
WHEN CHANGED BEGIN action OF ISodbc 2 IS connect := TRUE	project_name, ".MDB") write line OF file 3 := CONCAT("DefaultDir=", current_path) write line OF file 3 := "FIL=Microsoft Access"	Building element) = LOWCASE(GENERAL_Building_component) WHEN FOUND i := 1 WHEN NONE FOUND
WHEN CHANGED BEGIN action OF ISodbe 2 IS connect := TRUE EXEC ISodbe 2 SQL select Project_name from	project_name, ".MDB") write line OF file 3 := CONCAT("DefaultDir=", current_path) write line OF file 3 := "FIL=Microsoft Access" write line OF file 3 := "JetIniPath=msacc20.ini"	Building element) = LOWCASE(GENERAL_Building_component) WHEN FOUND i := 1 WHEN NONE FOUND i := 2
WHEN CHANGED BEGIN sction OF ISodbe 2 IS connect := TRUE EXEC ISodbe 2 SQL select Project_name from Current_projects	project_name, ".MDB") write line OF file 3 := CONCAT("DefaultDir=", current_path) write line OF file 3 := "FIL=Microsoft Access" write line OF file 3 := "JetlaiPath=msacc20.ini" write line OF file 3 := "UID=Admin"	Building element) = LOWCASE(GENERAL_Building_component) WHEN FOUND i := 1 WHEN NONE FOUND
WHEN CHANGED BEGIN action OF ISodbe 2 IS connect := TRUE EXEC ISodbe 2 SQL select Project_name from	project_name, ".MDB") write line OF file 3 := CONCAT("DefaultDir=", current_path) write line OF file 3 := "FIL=Microsoft Access" write line OF file 3 := "JetIniPath=msacc20.ini"	Building element) = LOWCASE(GENERAL_Building_component) WHEN FOUND i := 1 WHEN NONE FOUND i := 2 FIND END
WHEN CHANGED BEGIN sction OF ISodbe 2 IS connect: = TRUE EXEC ISodbe 2 SQL select Project_name from Current_projects END SQL INTO Current_projects END ATTRIBUTE Current_project SIMPLE	project_name, ".MDB") write line OF file 3 := CONCAT("DefaultDir=", current_path) write line OF file 3 := "FIL=Microsoft Access" write line OF file 3 := "JetlniPath=msacc20.ini" write line OF file 3 := "UID=Admin" write line OF file 3 := " action OF file 3 IS close := TRUE !	Building element) = LOWCASE(GENERAL_Building_component) WHEN FOUND i := 1 WHEN NONE FOUND i := 2 FIND END ! !!
WHEN CHANGED BEGIN sction OF ISodbe 2 IS connect: = TRUE EXEC ISodbe 2 SQL select Project name from Current_projects END SQL INTO Current_projects END ATTRIBUTE Current_project SIMPLE WHEN CHANGED	project_name, ".MDB") write line OF file 3 := CONCAT("DefaultDir=", current_path) write line OF file 3 := "FIL=Microsoft Access" write line OF file 3 := "JetlniPath=msacc20.ini" write line OF file 3 := "UID=Admin" write line OF file 3 := " action OF file 3 IS close := TRUE !	Building element) = LOWCASE(GENERAL_Building_component) WHEN FOUND i = 1 WHEN NONE FOUND i = 2 FIND END ! !! !! !! !! !! !! !! !! !! !! !! !!
WHEN CHANGED BEGIN scion OF ISodbe 2 IS connect: = TRUE EXEC ISodbe 2 SQL select Project_name from Current_projects END SQL INTO Current_projects END ATTRIBUTE Current_project SIMPLE WHEN CHANGED BEGIN	project_name, ".MDB") write line OF file 3 := CONCAT("DefaultDir=", current_path) write line OF file 3 := "FIL=Microsoft Access" write line OF file 3 := "JetlniPath=msacc20.ini" write line OF file 3 := "UID=Admin" write line OF file 3 := " action OF file 3 IS close := TRUE !	Building element) = LOWCASE(GENERAL_Building_component) WHEN FOUND i = 1 WHEN NONE FOUND i = 2 FIND END ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! !
WHEN CHANGED BEGIN sction OF ISodbe 2 IS connect: = TRUE EXEC ISodbe 2 SQL select Project name from Current_projects END SQL INTO Current_projects END ATTRIBUTE Current_project SIMPLE WHEN CHANGED	project_name, ".MDB") write line OF file 3 := CONCAT("DefaultDir=", current_path) write line OF file 3 := "FIL=Microsoft Access" write line OF file 3 := "JetlniPath=msacc20.ini" write line OF file 3 := "UID=Admin" write line OF file 3 := " action OF file 3 IS close := TRUE !	Building element) = LOWCASE(GENERAL_Building_component) WHEN FOUND i := 1 WHEN NONE FOUND i := 2 FIND END ! IF i = 1 THEN BEGIN text OF message box 2 := "This Component Already exists"
WHEN CHANGED BEGIN scion OF ISodbe 2 IS connect:= TRUE EXEC ISodbe 2 SQL select Project_name from Current_projects END SQL INTO Current_projects END ATTRIBUTE Current_project SIMPLE WHEN CHANGED BEGIN action OF ISodbe 2 IS connect:= TRUE EXEC ISodbe 2 SQL select Project_name from Current_projects	project_name, ".MDB") write line OF file 3 := CONCAT("DefaultDir=", current_path) write line OF file 3 := "FIL=Microsoft Access" write line OF file 3 := "JetIniPath=msacc20.ini" write line OF file 3 := "UID=Admin" write line OF file 3 := " action OF file 3 IS close := TRUE !	Building element) = LOWCASE(GENERAL_Building_component) WHEN FOUND i = 1 WHEN NONE FOUND i = 2 FIND END ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! !
WHEN CHANGED BEGIN action OF ISodbe 2 IS connect: = TRUE EXEC ISodbe 2 SQL select Project_name from Current_projects END SQL INTO Current_projects END ATTRIBUTE Current_project SIMPLE WHEN CHANGED BEGIN action OF ISodbe 2 IS connect:= TRUE EXEC ISodbe 2 SQL select Project_name from Current_projects END SQL INTO Current_projects	project_name, ".MDB") write line OF file 3 := "CONCAT("DefaultDir=", current_path) write line OF file 3 := "FIL=Microsoft Access" write line OF file 3 := "JetIniPath=msacc20.ini" write line OF file 3 := "UID=Admin" write line OF file 3 := " action OF file 3 IS close := TRUE !	Building element) = LOWCASE(GENERAL_Building_component) WHEN FOUND i := 1 WHEN NONE FOUND i := 2 FIND END IF i = 1 THEN BEGIN text OF message box 2 := "This Component Already exists" put up OF message box 2 := TRUE END
WHEN CHANGED BEGIN sction OF ISodbe 2 IS connect: = TRUE EXEC ISodbe 2 SQL select Project_name from Current_projects END SQL INTO Current_projects END ATTRIBUTE Current_project SIMPLE WHEN CHANGED BEGIN action OF ISodbe 2 IS connect:= TRUE EXEC ISodbe 2 SQL select Project_name from Current_projects END SQL INTO Current_projects END SQL INTO Current_projects	project_name, ".MDB") write line OF file 3 := "CONCAT("DefaultDir=", current_path) write line OF file 3 := "FIL=Microsoft Access" write line OF file 3 := "IetlniPath=msacc20.ini" write line OF file 3 := "UD=Admin" write line OF file 3 := " action OF file 3 IS close := TRUE !	Building element) = LOWCASE(GENERAL_Building_component) WHEN FOUND i = 1 WHEN NONE FOUND i = 2 FIND END ! ! IF i = 1 THEN BEGIN text OF message box 2 := "This Component Already exists" put up OF message box 2 := TRUE END !
WHEN CHANGED BEGIN scion OF ISodbe 2 IS connect := TRUE EXEC ISodbe 2 SQL select Project name from Current projects END SQL INTO Current projects END ATTRIBUTE Current project SIMPLE WHEN CHANGED BEGIN action OF ISodbe 2 IS connect := TRUE EXEC ISodbe 2 SQL select Project name from Current projects END SQL INTO Current projects END ATTRIBUTE back_to_current_project_display	project_name, ".MDB") write line OF file 3 := CONCAT("DefaultDir=", current_path) write line OF file 3 := "FIL=Microsoft Access" write line OF file 3 := "JetIniPath=msacc20.ini" write line OF file 3 := "UID=Admin" write line OF file 3 := " action OF file 3 IS close := TRUE !	Building element) = LOWCASE(GENERAL_Building_component) WHEN FOUND i = 1 WHEN NONE FOUND i = 2 FIND END IF i = 1 THEN BEGIN text OF message box 2 := "This Component Already exists" put up OF message box 2 := TRUE END IF i = 2 THEN
WHEN CHANGED BEGIN sction OF ISodbe 2 IS connect: = TRUE EXEC ISodbe 2 SQL select Project_name from Current_projects END SQL INTO Current_projects END ATTRIBUTE Current_project SIMPLE WHEN CHANGED BEGIN action OF ISodbe 2 IS connect:= TRUE EXEC ISodbe 2 SQL select Project_name from Current_projects END SQL INTO Current_projects END SQL INTO Current_projects	project_name, ".MDB") write line OF file 3 := CONCAT("DefaultDir=", current_path) write line OF file 3 := "FIL=Microsoft Access" write line OF file 3 := "JetIniPath=msacc20.ini" write line OF file 3 := "UID=Admin" write line OF file 3 := " action OF file 3 := " action OF file 3 IS close := TRUE !	Building element) = LOWCASE(GENERAL_Building_component) WHEN FOUND i = 1 WHEN NONE FOUND i = 2 FIND END ! ! IF i = 1 THEN BEGIN text OF message box 2 := "This Component Already exists" put up OF message box 2 := TRUE END !
WHEN CHANGED BEGIN scion OF ISodbe 2 IS connect: = TRUE EXEC ISodbe 2 SQL select Project name from Current projects END SQL INTO Current projects END ATTRIBUTE Current project SIMPLE WHEN CHANGED BEGIN action OF ISodbe 2 IS connect: = TRUE EXEC ISodbe 2 SQL select Project name from Current projects END SQL INTO Current projects END ATTRIBUTE back_to_current_project_display SIMPLE WHEN CHANGED BEGIN	project_name, ".MDB") write line OF file 3 := "CONCAT("DefaultDir=", current_path) write line OF file 3 := "FIL=Microsoft Access" write line OF file 3 := "JetlaiPath=msacc20.ini" write line OF file 3 := "UID=Admin" write line OF file 3 := " action OF file 3 IS close := TRUE !	Building element) = LOWCASE(GENERAL_Building_component) WHEN FOUND i = 1 WHEN NONE FOUND i = 2 FIND END IF i = 1 THEN BEGIN text OF message box 2 := "This Component Already exists" put up OF message box 2 := TRUE END IF i = 2 THEN
WHEN CHANGED BEGIN sction OF ISodbe 2 IS connect := TRUE EXEC ISodbe 2 SQL select Project_name from Current_projects END SQL INTO Current_projects END ATTRIBUTE Current_project SIMPLE WHEN CHANGED BEGIN sction OF ISodbe 2 IS connect := TRUE EXEC ISodbe 2 SQL select Project_name from Current_projects END SQL INTO Current_projects END ATTRIBUTE back_to_current_project_display SIMPLE WHEN CHANGED	project_name, ".MDB") write line OF file 3 := "CONCAT("DefaultDir=", current_path) write line OF file 3 := "FIL=Microsoft Access" write line OF file 3 := "JetIniPath=msacc20.ini" write line OF file 3 := "UD=Admin" write line OF file 3 := " action OF file 3 IS close := TRUE !	Building element) = LOWCASE(GENERAL_Building_component) WHEN FOUND i = 1 WHEN NONE FOUND i = 2 FIND END IF i = 1 THEN BEGIN text OF message box 2 := "This Component Already exists" put up OF message box 2 := TRUE END IF i = 2 THEN BEGIN IF i = 2 THEN BEGIN

```
tion OF file 2 IS close := TRUE
- make a copy of the empty database
CTIVATE "IPU, EXTERN, project.bat"
-- connect the display to the window 
sput OF main window :=
project_selection_display
end of method
DEBUTE path_of_odbc STRING
LIBUTE current_path STRING
LIBUTE III NUMERIC
LIBUTE IIIX STRING
IBUTE get from general database SIMPLE
EN CHANGED
ŒC 15odbc 2 SQL select distinct
ing_element) from [General-AB-attribute-
er]
 END SQL INTO
RAL_Building_element
remove the building elements that are already in
R (i := 1 TO number instances OF navigate
ng element)
EGIN
nstance number OF navigate Building element
TND GENERAL_Building_element
WHERE building_element OF
RAL_Building_element = building_element OF
ng element
WHEN FOUND
 FORGET CURRENT
RAL_Building_element
FIND END
ible OF get from general database window :=
tput OF get from general database window := add
eneral database display
end of method
BUTE GENERAL_Building_component
IG
N CHANGED
EC ISodbe 2 SQL SELECT * from [General-
ribute-designer] where building_element =
n.GENERAL_Building_component
END SQL INTO
CAL_building_element_attribute
BUTE add component from _get from general
IMPLE
N CHANGED
IN
OD Building element
PHERE LOWCASE( building_element OF
rg element) = LOWCASE(
RAL_Building_component)
PHEN FOUND
HEN NONE FOUND
ID END
 - I THEN
EGIN
ext OF message box 2 := "This Component
out up OF message box 2 := TRUE
= 2 THEN
create the database table with its id attribute
```

GENERAL_Building_component, "_id] TEXT(50));")	cipline)	END .
EXEC ISOdic I SQL WITH general statement of sql END SQL		**** add the attributes to the table [AB-attribute-
1	END	designer] in the new project
! **** add the rest of attributes to the created table	sqr	FOR (ii = 1 TO number instances OF naviga
FOR (ii := 1 TO number instances OF navigate GENERAL building element attributes)	END END	GENERAL building element attributes) BEGIN
BEGIN	END	instance number OF navigate GENERAL
instance number OF nevigate GENERAL	!	building element attributes := ii
building element attributes := ii	! *** end of method	IF CONF(linked building element OF
f **** ignore the attribute that has the _id of the table	END ATTRIBUTE return from _get from general datab	GENERAL building element attribute) = 100 THEN BEGIN
IF SEARCHSTR(*_id*,	SIMPLE	1
attribute_of_building_element OF	WHEN CHANGED	! ***** the following step is necessary as da
GENERAL_building_element_attribute) = 0 THEN	BEGIN	can not be got directly from access to this attribute
BEGIN 1 **** put the attributes in the database	EXEC ISodbe I SQL SELECT distinct	EXEC 15odbc 2 SQL SELECT (linked_attribute_of_building_element) from
general statement of sql := CONCAT(*alter	(building_element), graph_name from [AB-attribute-	[GENERAL-AB-attribute-designer]
table [", building_element OF	designer]	where building_element =
GENERAL_Building_element, "] add [",	END SQL INTO Building element	:domain.GENERAL_Building_component and
attribute_of_building_element OF	PVECISARA I COI CEI ECT Émises	attribute_of_building_element = :GENERAL_building_element_attribute_attribute_of_
GENERAL_building_element_attribute, ") ") IF attribute_type OF	EXEC ISodbe 1 SQL SELECT distinct (building element) from [AB-attribute-designer]	uilding element
GENERAL_building_element_attribute = "t" THEN	END SQL INTO Linked building element	END SQL INTO domain (ttt)
general statement of sqi := CONCAT(1	linked_attribute_of_building_element OF
general statement of aql, "TEXT (50);")	visible OF get from general database window :=	GENERAL_building_element_attribute := tttt
IF attribute_type OF GENERAL_building_element_attribute = "a" THEN	FALSE END	EXEC ISodbe 1 SQL INSERT INTO [AB-
general statement of sql := CONCAT(ATTRIBUTE add component and general rule	attribute-designer] (building element,
general statement of sql, "Number;")	SIMPLE	attribute of building element, attribute type,
EXEC ISodbc 1 SQL WITH general	WHEN CHANGED	designing discipline, linked building element,
statement of sql END SQL	BEGIN	linked_attribute_of_building_element) values
END END	i := 0 FIND Building element	:domain.GENERAL_Building_component,
1	WHERE LOWCASE(building element OF	GENERAL building element attribute attribute of
! **** add the attributes to the table [AB-attribute-	Building element) = LOWCASE(uilding_element,
designer] in the new project	GENERAL_Building_component)	:GENERAL_building_element_attribute.attribute_typ
FOR (ii := 1 TO number instances OF navigate	WHEN FOUND	GENERAL building element_attribute.designing_di cipline; GENERAL building_element_attribute.linko
GENERAL building element attributes) BEGIN	i := 1 WHEN NONE FOUND	building element GENERAL building element att
instance number OF navigate GENERAL	i := 2	bute linked attribute of building element)
building element attributes := ii	FIND END	
IF CONF(linked_building_element OF	!	END
GENERAL_building_element_attribute) = 100 THEN BEGIN	! IFi=1 THEN	SQL
!	BEGIN	END
! ***** the following step is necessary as data	text OF message box 2 := "This Component	ELSE
can not be got directly from access to this attribute	Already exists*	BEGIN
EXEC ISodic 2 SQL SELECT	put up OF message box 2 := TRUE	EXEC ISodbe 1 SQL INSERT INTO [AB- attribute-designer] (building_element,
(linked_attribute_of_building_element) from [GENERAL-AB-attribute-designer]	END .	attribute of building element, attribute_type,
where building element =	i i	designing discipline) values
:domain.GENERAL_Building_component and	IF i = 2 THEN	(
attribute_of_building_element = :GENERAL_building_element_attribute.attribute_of_b	BEGIN 1	:domain.GENERAL_Building_component, :GENERAL_building_element_attribute.attribute_of_
uilding_element	! **** create the database table with its id attribute	uilding element
END SQL INTO domain (tttt)	general statement of sql := CONCAT(*CREATE	:GENERAL_building_element_attribute_attribute_typ
linked_attribute_of_building_element OF	TABLE [", GENERAL_Building_component, "] ([",	:GENERAL_building_element_attribute.designing_di
GENERAL_building_element_attribute := tttt	GENERAL_Building_component, "_id] TEXT(50));")	cipline)
! EXEC ISodbe 1 SQL INSERT INTO [AB-	EXEC ISodbe 1 SQL WITH general statement of sql END SQL	i
attribute-designer] (building element,	1	END
attribute_of_building_element, attribute_type,	**** add the rest of attributes to the created table	SQL
designing_discipline, linked_building_element,	FOR (ii := 1 TO number instances OF navigate	END
linked_attribute_of_building_element) values	GENERAL building element attributes) BEGIN	END
:domain.GENERAL_Building_component,	instance number OF navigate GENERAL	! **** Add the General Rules
:GENERAL_building_element_attribute_attribute_of_b	building element attributes := ii	l (
uilding_element,	! **** ignore the attribute that has the _id of the	EXEC 15odbc 2 SQL select * from [GENERAL
:GENERAL_building_element_attribute_attribute_type, :GENERAL_building_element_attribute_designing_dis	table IF SEARCHSTR("_id",	AB-general-rules) where building_element = :domain.GENERAL_Building_component
cipline,:GENERAL_building_element_attribute.linked	attribute_of_building_element OF	END SQL INTO AB_general_rules
building element :GENERAL building element artri	GENERAL_building_element_attribute) = 0 THEN	1 !
bute.linked_attribute_of_building_element)	BEGIN	! **** add the rules to the new project database
	<pre>! **** put the attributes in the database general statement of sq! := CONCAT(*alter</pre>	FOR (ii := 1 TO number instances OF rules
END	table [", building_element OF	navigate 2)
SQL	GENERAL_Building_element, *] add [*.	BEGIN
END -	attribute_of_building_element OF	instance number OF rules navigate 2 := ii
ELSE RECIN	GENERAL building element attribute, "] ")	EXEC ISodbe 1 SQL INSERT INTO [AB- general-rules] (building_element,
BEGIN EXEC 150dbc SQL INSERT INTO [AB-	IF attribute_type OF GENERAL_building_element_attribute = "t" THEN	attribute_of_building_element, affected_discipline,
attribute-designer) (building_element,	general statement of sql := CONCAT(percent_change, result_in_major_modification,
attribute_of_building_element, attribute_type,	general statement of sqi, "TEXT (50);")	cause of the effect) values
designing discipline) values	IF attribute_type OF	(:AB general rules building element
domain.GENERAL_Building_component,	GENERAL_building_element_attribute = "n" THEN general statement of sql := CONCAT(:AB_general_rules.attribute_of_building_element, :AB_general_rules.affected_discipline,
:GENERAL_building_element_attribute_stribute_of_b	general statement of sql, "Number;")	:AB general rules percent change,
uilding element,	EXEC ISodbe 1 SQL WITH general	AB_general_rules.result_in_major_modification.
:GENERAL_building_element_attribute. attribute_type,	statement of sql END SQL	:AB_general_rules.cause_of_the_effect)
GENERAL_building_element_attribute designing_dis	END	1

END INSTANCE border 9 ISA border SOL WITH style IS edit control FND WITH perspective IS in WITH border width := 1 WITH background color := 192,192,192 oos end of IF WITH fill color := 192,192,192 WITH highlight color := 255,255,255 WITH shadow color := 128,128,128 END 1 *** end of method WITH location = 253 50 500 395 END INSTANCE border 10 ISA border INSTANCE border I ISA border WITH style IS shadow WITH style IS group WITH perspective IS in WITH perspective IS in WITH border width := 4 WITH border width := 1 WITH background color := 192,192,192 WITH fill color := 192,192,192 WITH background color := 192,192,192 WITH fill color := 0,0,128 WITH highlight color := 255,255,255 WITH stadow color := 128,128,128 WITH location := 5,45,175,415 WITH highlight color := 0,0,0 WITH shadow color := 0,0,0 WITH location := 5,5,165,100 INSTANCE border 11 ISA border INSTANCE border 2 ISA border WITH style IS group WITH style IS group
WITH perspective IS is
WITH border width := 1 WITH perspective IS in WITH border width := 1 WITH background color := 192,192,192 WITH fill color := 192,192,192
WITH highlight color := 255,255,255
WITH shadow color := 128,128,128 WITH background color := 192,192,192 WITH fill color := 192,192,192 WITH highlight color := 255,255,255 WITH shadow color := 128,128,128 WITH location := 180,45,340,415 WITH location = 5 40 420 135 INSTANCE border 12 ISA border WITH style IS group WITH perspective IS in WITH border width := 1 **INSTANCE border 3 ISA border** WITH style IS group WITH perspective IS in WITH border width := 1 WITH background color := 192,192,192 WITH background color := 192,192,192 WITH fill color := 192,192,192 WITH fill color := 192,192,192 WITH highlight color := 255,255,255 WITH highlight color := 255,255,255 WITH shadow color := 128,128,128 WITH location := 5,140,500,410 WITH shadow color := 128,128,128 WITH location := 345,45,500,355 INSTANCE border 13 ISA border INSTANCE border 4 ISA border WITH style IS shadow WITH style IS group WITH perspective IS in WITH perspective IS in WITH border width := i WITH border width := 4 WITH background color := 192,192,192 WITH background color := 192,192,192 WITH fill color := 0,0,128 WITH fill color := 192,192,192 WITH highlight color := 255,255,255 WITH shadow color := 128,128,128 WITH highlight color := 0,0,0 WITH shadow color := 0,0,0 WITH location := 5,5,165,75 WITH location := 15,295,490,400 INSTANCE border 14 ISA border WITH style IS shadow INSTANCE border 5 ISA border WITH perspective IS in WITH border width := 4 WITH style IS picture frame WITH perspective IS in WITH border width: = 4 WITH background color := 192,192,192 WITH fill color := 0,0,128 WITH highlight color := 0,0,0 WITH shadow color := 0,0,0 WITH background color := 192,192,192 WITH fill color := 192,192,192 WITH highlight color := 255,255,255 WITH shadow color := 128,128,128 WITH location := 10.5.170.60 WITH location := 5,40,595,245 INSTANCE border 15 ISA border INSTANCE border 6 invisible ISA border WITH style IS group WITH style IS picture frame WITH perspective IS in WITH perspective IS in WITH border width := 1 WITH border width := 1 WITH background color: = 192,192,192 WITH background color := 192,192,192 WITH fill color := 192,192,192 WITH fill color := 192,192,192 WITH highlight color := 255,255,255 WITH highlight color := 255,255,255 WITH shadow color := 128,128,128 WITH shadow color := 128,128,128 WITH location := 5,40,490,400 WITH location := 0,0,0,0 INSTANCE border 16 ISA border WITH style IS group WITH perspective IS in WITH border width := 1 INSTANCE border 7 invisible2 ISA border WITH style IS picture frame WITH perspective IS in WITH border width := 1 WITH background color := 192,192,192 WITH fill color := 192,192, 192 WITH highlight color := 255,255,255 WITH shadow color := 128,128,128 WITH background color := 192,192,192 WITH Gateground color := 122,122,12
WITH fill color := 192,192,192
WITH highlight color := 255,255,255
WITH shadow color := 128,128,128 WITH location := 10,45,485,395 WITH location := 0.0.0.0 **INSTANCE border 17 ISA border** INSTANCE border 8 ISA border WITH style IS picture frame WITH perspective IS in WITH style IS edit control WITH perspective IS in
WITH border width := 1
WITH background color := 192,192,192 WITH border width := 3 WITH background color := 192,192,192 WITH fill color := 192,192,192

WITH highlight color := 255,255,255 WITH shadow color := 128,128,128

WITH location := 20,55,475,195

WITH fill color := 192,192,192

WITH location := 5,50,250,395

WITH highlight color := 255,255,255 WITH shadow color := 128,128,128

INSTANCE border 18 ISA border WITH style IS picture frame WITH perspective IS in WITH border width := 3 WITH background color := 192,192,192 WITH fill color := 192,192,192 WITH highlight color := 255,255,255 WITH shadow color := 128,128,128 WITH location := 20,205,475,345 INSTANCE border 19 ISA border WITH style IS group WITH perspective IS in WITH border width := 1 WITH background color := 192,192,192 WITH fill color := 192,192,192 WITH highlight color := 255,255,255 WITH shadow color := 128,128,128 WITH location := 5.40,490,400 INSTANCE border 20 ISA border WITH style IS group WITH perspective IS in WITH border width := 1 WITH background color: 192,192,192 WITH fill color := 192,192,192 WITH highlight color := 255,255,255 WITH shadow color := 128,128,128 WITH location := 10,45,485,395 INSTANCE border 21 ISA border WITH style IS picture frame WITH perspective IS in WITH border width := 3 WITH background color := 192,192,192 WITH fill color := 192,192,192 WITH highlight color := 255,255,255 WITH shadow color := 128,128,128 WITH location := 20,110,240,385 INSTANCE border 22 ISA border WITH style IS picture frame WITH perspective IS in WITH border width := 3 WITH background color := 192,192,192 WITH fill color := 192,192,192 WITH highlight color := 255,255,255 WITH shadow color := 128,128,128 WITH location := 255,110,475,320 INSTANCE border 23 ISA border WITH style IS picture frame WITH perspective IS in WITH border width := 3 WITH background color := 192,192,192 WITH fill color := 0,128,128 WITH highlight color := 255,255,255 WITH shadow color := 128,128,128 WITH location := 20,55,475,100 **INSTANCE border 24 ISA border** WITH style IS group WITH perspective IS in WITH border width := 1 WITH background color := 192,192,192 WITH fill color := 192,192,192 WITH highlight color := 255,255,255 WITH shadow color := 128,128,128 WITH location := 5.40.415.340 INSTANCE border 25 ISA border WITH style IS group WITH perspective IS in WITH border width := 1 WITH background color := 192,192,192 WITH fill color := 192,192,192 WITH highlight color := 255,255,255 WITH shadow color := 128,128,128 WITH location := 10,45,410,335 INSTANCE edit table column I ISA edit table column WITH attachment := attribute_of_building_element OF building_element_attribute WITH heading label :-'attribute_of_building_element" WITH width := 70 WITH justify IS left WITH pen color [1]: "UNDETERMINED WITH fill color [1]: "UNDETERMINED

INSTANCE edit table column 2 ISA edit table column

WITH attachment := attribute_type OF INSTANCE navigate menu_group_items ISA navigate WITH class attachment := Menus_group_items WITH items [7] := UNDETERMINED WITH items [8] := UNDETERMINED building element attribute
WITH heading label := "attribute_type"
WITH width := 70 WITH items [9]:= pushbutton 34
WITH items [10]:= pushbutton 35
WITH fill color:= 192,192,192 INSTANCE navigate building element attributes ISA WITH justify IS left
WITH pes color [1] := UNDETERMINED
WITH fill color [1] := UNDETERMINED WITH class attachment := building_element_attribute WITH menus [1]:= UNDETERMINED INSTANCE navigate GENERAL building elements INSTANCE config ISA display INSTANCE edit table column 3 ISA edit table column ISA navigate
WITH class attachment := WITH wait := FALSE WITH attachment := designing_discipline OF building_element_attribute WITH heading label := "designing_discipline" WITH width := 70 WITH delay changes := FALSE GENERAL_Building elem WITH items [1] := textbox 2 WITH items [2] := border 2 WITH items [3] := listbox 1 INSTANCE navigate GENERAL building element attributes ISA navigate
WITH class attachment := WITH justify IS left
WITH pea color [1]:= UNDETERMINED
WITH fill color [1]:= UNDETERMINED WITH items [4]:= valuebox 1
WITH items [5]:= textbox 3
WITH items [6]:= textbox 4 GENERAL_building_element_attribute WITH items [7] := border 3 WITH items [8] := listbox 2 INSTANCE edit table column 4 ISA edit table column WITH attachment := linked_building_element OF building_element_attribute INSTANCE the application ISA application WITH unknowns fail := TRUE WITH threshold := 50 WITH items [9] := textbox 5
WITH items [10] := valuebox 2 WITH heading label := "linked_building_element"
WITH width := 70 WITH title display := Starting display WITH items [11] := border 4 WITH items [12] := textbox 6 WITH ignore breakpoints := FALSE WITH justify IS left
WITH pan color [1]:= UNDETERMINED
WITH fill color [1]:= UNDETERMINED WITH reasoning on := FALSE WITH items [13]:= radiobutton group 2 WITH numeric precision := 8 WITH items [14] := UNDETERMINED WITH demon strategy IS fire first WITH items [15]: = textbox 7
WITH items [16]: = promptbox 1 INSTANCE edit table column 5 ISA edit table column INSTANCE column 1 ISA column WITH items [17] := textb WITH attachment := attribute_of_building_element OF building_element_attribute WITH items [18] := pushbutton 3
WITH items [19] := pushbutton 4
WITH items [20] := UNDETERMINED linked_attribute_of_building_element OF ng clement attribute WITH width = 160 WITH heading label :-WITH justify IS left WITH items [21] := listbox 3 WITH items [22] := textbox 9 "linked_attribute_of_building_element" WITH width := 70 WITH wordwrap := TRUE WITH heading label := "Antribute"
WITH heading justify IS heading center
WITH heading wordwrap := TRUE WITH items [23] := listbox 4 WITH justify IS left WITH items [24] := valuebox 3 WITH items [25] := valuebox 4 WITH pen color [1]:= UNDETERMINED WITH fill color [1]:= UNDETERMINED WITH items [26] := textbox 10 INSTANCE column 2 ISA column WITH items [27] := textbox | 1 WITH items [28] := valuebox 5 INSTANCE ISodbe 1 ISA ISodbe WITH attachment := attribute_type OF WITH connection prompting := FALSE WITH auto commit := TRUE building element attribute WITH items [29] := valuebox 6 WITH width := 40 WITH items [30] := textbox 12 WITH append := FALSE WITH default error handling := TRUE WITH justify IS left WITH items [31]:= textbox 13 WITH wordwrap := TRUE
WITH heading label := "Type"
WITH heading justify is heading center WITH items [32] := pushbutton 5 WITH items [33] := pushbutton 6 WITH items [34] := textbox 14 INSTANCE ISodbe 2 ISA ISodbe WITH data source := "general" WITH items [35] = rediobution group I
WITH items [36] = pushbution 14
WITH items [37] := pushbution 15
WITH items [38] = UNDETERMINED
WITH items [38] := border 6 invisible WITH heading wordwrap := TRUE WITH connection prompting := FALSE WITH auto commit := TRUE
WITH append := FALSE
WITH default error handling := TRUE INSTANCE column 3 ISA column WITH attachment : designing_discipline OF building_element_attribute WITH width := 60 WITH items [40] := promptbox 2 invisible INSTANCE ISodbe 3 ISA ISodbe WITH items [41] := textbox 17 invisible WITH items [42] := pushbutton 10 invisible WITH items [43] := UNDETERMINED WITH justify IS left WITH connection prompting := FALSE WITH wordwrap := TRUE WITH auto commit := TRUE WITH heading label := "Design WITH append := FALSE
WITH default error handling := TRUE WITH items [44]:= pushbutton 11 invisible WITH items [45]:= border 7 invisible2 WITH items [46]:= promptbox 3 invisible2 WITH heading justify IS heading center WITH heading wordwrap := TRUE INSTANCE message box 1 ISA message box WITH title: "Warning Message" WITH items [47] := textbox 18 invisible2 WITH items [48] = textbox 19 invisible2
WITH items [49] = pushbutton 12 invisible2
WITH items [50] := UNDETERMINED
WITH items [51] := UNDETERMINED
WITH items [52] := pushbutton 16 invisible2 INSTANCE column 4 ISA column WITH the: "Warming message
WITH text: "Are you sure that you want to
DELETE the current building component" WITH attachment := linked_building_element OF building_element_attribute WITH modal style IS sys modal WITH width := 160 WITH icon IS question WITH button IS yes no WITH justify IS left WITH items [53] := pushbutton 16 invisible2
WITH items [53] := pushbutton 26 invisible2
WITH items [54] := pushbutton 52
WITH items [55] := textbox 64
WITH fill color := 192,192,192
WITH menus [1] := UNDETERMINED WITH wordwrap := TRUE WITH heading label := *Linked WITH default button IS one Building INSTANCE message box 2 ISA message box Component WITH title := "INPUT ERROR" WITH heading justify IS heading center WITH modal style IS app modal WITH heading wordwrap := TRUE WITH icon IS stop sign INSTANCE attributes table display ISA display WITH button IS ok WITH default button IS one INSTANCE column 5 ISA column WITH wait := FALSE WITH attachment := WITH delay changes := FALSE WITH items [1] = border 5
WITH items [2] := table 1
WITH items [2] := textbox 15
WITH items [4] := textbox 16
WITH items [5] := UNDETERMINED linked_attribute_of_building_element OF INSTANCE message box 3 ISA message box building_element_attribute WITH title := "Warning Message"
WITH text := "Are you sure that you want to WITH width := 160 WITH justify IS left DELETE the current component attribute wordwrap := TRUE WITH items [6] := pushbutton 8
WITH fill color := 192,192,192
WITH menus [1] := UNDETERMINED WITH heading label := "Linked WITH modal style IS sys modal Component WITH icon IS question WITH button IS yes no WITH heading justify IS heading center WITH heading wordwrap := TRUE WITH default button IS one INSTANCE General rules display ISA display WITH wait := FALSE INSTANCE navigate 1 ISA navigate **INSTANCE Starting display ISA display** WITH delay changes := FALSE WITH class attachment := building_element_attribute WITH items [1] = textbox 20
WITH items [2] = border 8
WITH items [3] := UNDETERMINED
WITH items [4] = UNDETERMINED
WITH items [5] := textbox 23 WITH wait := TRUE WITH delay changes := FALSE WITH items [1]:= UNDETERMINED
WITH items [2]:= border 1
WITH items [3]:= textbox 1
WITH items [4]:= pushbutton 2
WITH items [5]:= UNDETERMINED
WITH items [6]:= UNDETERMINED INSTANCE rules navigate 2 ISA navigate WITH class attachment := AB_general_rules INSTANCE navigate Building element ISA navigate WITH class attachment := Building element WITH items [6] := textbox 24
WITH items [7] := UNDETERMINED
WITH items [8] := promptbox 4

```
WITH items [9]:= border 9
WITH items [10]:= textbox 26
WITH items [11]:= valuebox 7
                                                                                                                                                                                                                           WITH items [24] := border 23
WITH items [25] := textbox 63
WITH items [26] := listbox 17
                                                                                                              INSTANCE CURENT_project_selection_display ISA
                                                                                                              display
WITH wait := FALSE
    WITH items [12]:= promptbox 6
WITH items [13]:= listbox 5
WITH items [14]:= UNDETERMINED
                                                                                                                                                                                                                           WITH items [27] := pushbutton 51
WITH fill color := 192,192,192
WITH menus [1] := UNDETERMINED
                                                                                                               WITH delay changes: = FALSE
WITH items [1]:= border 1
WITH items [2]:= pushbutton 1
WITH items [3]:= pushbutton 17
WITH items [4]:= pushbutton 26
WITH items [6]:= textbox 45
WITH items [6]:= textbox 45
    WITH items [15]:= textbox 27
WITH items [16]:= textbox 28
                                                                                                                                                                                                                        INSTANCE add from general database display ISA
    WITH items [17] := textbox 29
                                                                                                                                                                                                                          WITH wait := FALSE
    WITH items [18] := valuebox 9
WITH items [19] := valuebox 10
WITH items [20] := pushbutton 18
                                                                                                                                                                                                                         WITH wait := FALSE
WITH delay changes := FALSE
WITH items [1] := UNDETERMINED
WITH items [2] := UNDETERMINED
WITH items [3] := UNDETERMINED
WITH items [4] := UNDETERMINED
WITH items [6] := UNDETERMINED
WITH items [6] := UNDETERMINED
WITH items [7] := UNDETERMINED
WITH items [7] := UNDETERMINED
WITH items [8] := UNDETERMINED
                                                                                                                WITH items [6]:= pushbutton 36
WITH fill color:= 192,192,192
   WITH items [22] := pushbutton 19
WITH items [22] := pushbutton 20
WITH items [22] := pushbutton 21
WITH items [24] := listhox 6
WITH items [25] := exerbox 30
WITH items [25] := valuebox 11
WITH items [25] := valuebox 11
                                                                                                                WITH menus [1] := UNDETERMINED
                                                                                                              INSTANCE NEW_project_selection_display ISA
                                                                                                              display
WITH wait := FALSE
                                                                                                                WITH delay changes := FALSE
    WITH items [27] := radiobutton group 3
WITH items [28] := textbox 31
WITH items [29] := pushbutton 22
                                                                                                                                                                                                                           WITH items [8] := UNDETERMINED
WITH items [9] := UNDETERMINED
WITH items [10] := border 24
                                                                                                               WITH items [1]:=border 1
WITH items [2]:=textbox 49
WITH items [3]:=pusibutton 39
                                                                                                               WITH items [4]:= pushbutton 40
WITH items [5]:= pushbutton 41
WITH fill color:= 192,192,192
WITH menus [1]:= UNDETERMINED
    WITH items [30]:= pushbutton 23
WITH items [31]:= pushbutton 24
                                                                                                                                                                                                                          WITH items [11]:= border 25
WITH items [12]:= listbox 18
WITH items [13]:= valuebox 19
    WITH items [32] := textbox 32
   WITH items [33]:= textbox 33
WITH items [34]:= promptbox 7
WITH items [35]:= textbox 69
                                                                                                                                                                                                                          WITH items [14] := tentbox 65
WITH items [15] := tentbox 66
                                                                                                             INSTANCE NEW from another project display ISA
                                                                                                                                                                                                                          WITH items [16]:= listbox 19
                                                                                                                                                                                                                          WITH items [17]:= textbox 67
WITH items [18]:= textbox 68
   WITH items [36]:= promptbox 5
WITH items [37]:= textbox 70
                                                                                                             display
WITH wait := FALSE
                                                                                                                                                                                                                         WITH items [19] := pushbutton 53
WITH items [20] := pushbutton 53
WITH items [20] := pushbutton 54
WITH items [21] := pushbutton 55
WITH fill color := 192,192,192
WITH menus [1] := UNDETERMINED
   WITH items [38] := valuebox 8
                                                                                                               WITH delay changes :- FALSE
                                                                                                             WITH delay changes := FALSE
WITH items [1] := border 13
WITH items [2] := UNDETERMINED
WITH items [3] := textbox 50
WITH items [4] := textbox 51
WITH items [5] := itextbox 12
WITH items [6] := pushbutton 42
WITH items [7] := pushbutton 43
WITH items [8] := pushbutton 44
WITH items [9] := pushbutton 45
WITH fill color := 192,192,192
WITH memus [1] := UNDETERMINED
   WITH fill color := 192,192,192
WITH menus [1] := UNDETERMINED
 INSTANCE Menus display ISA display
   WITH wait := FALSE
   WITH delay changes := FALSE
                                                                                                                                                                                                                       INSTANCE file I ISA file
  WITH delay changes: FALSE
WITH items [1]: border 12
WITH items [2]: border 11
WITH items [3]: border 10
WITH items [3]: border 10
WITH items [5]: listbox 34
WITH items [5]: listbox 36
WITH items [6]: textbox 36
WITH items [8]: textbox 37
WITH items [8]: textbox 37
WITH items [9]: pushbutton 27
WITH items [10]: listbox 10
WITH items [11]: textbox 38
WITH items [12]: whisebox 13
                                                                                                                                                                                                                         WITH filename := "project.ini"
WITH shared := FALSE
                                                                                                                                                                                                                          WITH default error handling := TRUE
                                                                                                                                                                                                                       INSTANCE file 2 ISA file
                                                                                                                                                                                                                         WITH filename := "project.bat"
WITH shared := FALSE
                                                                                                            INSTANCE NEW from components display ISA
                                                                                                                                                                                                                          WITH default error handling := TRUE
                                                                                                            display
                                                                                                              WITH wait := FALSE
                                                                                                              WITH delay changes := FALSE
WITH items [1] := border 15
WITH items [2] := border 16
WITH items [3] := border 17
WITH items [4] := border 18
                                                                                                                                                                                                                       INSTANCE file 3 ISA file
                                                                                                                                                                                                                         WITH shared := FALSE
   WITH items [12]:= valuebox 13
WITH items [13]:= textbox 39
WITH items [14]:= textbox 40
                                                                                                                                                                                                                         WITH default error handling := TRUE
                                                                                                                                                                                                                       INSTANCE listbox 1 ISA listbox
                                                                                                              WITH items [5]:= textbox 53
WITH items [6]:= listbox 13
WITH items [7]:= textbox 54
   WITH items [15]:= pushbutton 28
                                                                                                                                                                                                                         WITH show current instance := TRUE
   WITH items [16]:= pushbutton 29
WITH items [17]:= valuebox 14
                                                                                                                                                                                                                         WITH source := building_element OF Building
   WITH items [18]:= pushbutton 30
WITH items [19]:= pushbutton 31
WITH items [20]:= pushbutton 32
                                                                                                              WITH items [8] := valuebox 16
WITH items [9] := textbox 55
WITH items [10] := UNDETERMINED
                                                                                                                                                                                                                         WITH destination := building_element OF
                                                                                                                                                                                                                       current_data
WITH location := 16,71,210,95
                                                                                                              WITH items [11]: = UNDETERMINED
WITH items [12]: = listbox 14
WITH items [13]: = textbox 57
WITH items [14]: = pushbutton 47
   WITH items [21] := valuebox 15
   WITH items [22] := promptbox 8
WITH items [23] := textbox 41
                                                                                                                                                                                                                       INSTANCE listbox 2 ISA listbox
                                                                                                                                                                                                                         WITH show current instance := TRUE
   WITH items [24] := textbox 42
                                                                                                                                                                                                                         WITH source := attribute_of_building_element OF
   WITH fill color := 192,192,192
WITH menus [1] := UNDETERMINED
                                                                                                              WITH items [15] = pushbutton 48
WITH items [16] = UNDETERMINED
WITH fill color := 192,192,192
WITH menus [1] := UNDETERMINED
                                                                                                                                                                                                                      building element attribute
WITH destination := attribute of building element
                                                                                                                                                                                                                      OF current_data
WITH location := 16,171,235,240
INSTANCE CURENT_project_start_display ISA
display
WITH wait := FALSE
                                                                                                            INSTANCE NEW from systems display ISA display
                                                                                                                                                                                                                       INSTANCE listbox 3 ISA listbox
   WITH delay changes := FALSE
                                                                                                              WITH wait := FALSE
                                                                                                                                                                                                                         WITH show current instance := TRUE
  WITH items [1]:= border 1
WITH items [2]:= UNDETERMINED
WITH items [3]:= UNDETERMINED
WITH items [4]:= UNDETERMINED
                                                                                                              WITH delay changes := FALSE
                                                                                                                                                                                                                         WITH source := building_element OF Linked
                                                                                                              WITH items [1]:= UNDETERMINED WITH items [2]:= UNDETERMINED WITH items [3]:= UNDETERMINED
                                                                                                                                                                                                                       building element
                                                                                                                                                                                                                         WITH destination := linked_building_element OF
  WITH items [5] := UNDETERMINED
WITH items [6] := textbox 43
WITH items [7] := textbox 44
                                                                                                                                                                                                                         WITH location := 25,320,235,345
                                                                                                              WITH items [4] := UNDETERMINED WITH items [5] := UNDETERMINED
                                                                                                              WITH items [6] := UNDETERMINED
                                                                                                                                                                                                                       INSTANCE listbox 4 ISA listbox
  WITH items [8 ] := pushbutton 33
WITH items [9 ] := listbox 11
                                                                                                              WITH items [7]:= UNDETERMINED WITH items [8]:= UNDETERMINED WITH items [9]:= textbox 58
                                                                                                                                                                                                                         WITH show current instance := TRUE
                                                                                                                                                                                                                         WITH source := attribute_of_building_element OF
  WITH items [10]:= pushbutton 37
WITH fill color:= 192,192,192
WITH menus [1]:= UNDETERMINED
                                                                                                                                                                                                                       Linked element attribute
                                                                                                              WITH items [10] := border 19
                                                                                                                                                                                                                         WITH destination :=
                                                                                                                                                                                                                      linked_attribute_of_building_element OF current_data
WITH location := 25,365,235,390
                                                                                                              WITH items [11] := border 20
WITH items [12] := border 21
INSTANCE NEW_project_start_display ISA display WITH wait := FALSE
                                                                                                              WITH items [13] := border 22
WITH items [14] := textbox 59
WITH items [15] := textbox 60
                                                                                                                                                                                                                       INSTANCE listbox 5 ISA listbox
  WITH delay changes := FALSE
                                                                                                                                                                                                                         WITH show current instance := TRUE
  WITH items [1] := border 1
WITH items [2] := textbox 47
                                                                                                              WITH items [16]:= textbox 61
                                                                                                                                                                                                                         WITH source := building_element OF Building
                                                                                                              WITH items [17] := valuebox 17
  WITH items [3] := textbox 47
WITH items [3] := pushbutton 33
WITH items [4] := textbox 48
WITH items [5] := promptbox 9
WITH items [6] := pushbutton 38
WITH fill color := 192,192,192
                                                                                                              WITH items [18]:= textbox 62
                                                                                                                                                                                                                         WITH destination := building_element OF
                                                                                                              WITH items [19] := valuebox 18
WITH items [20] := listbox 15
WITH items [21] := listbox 16
                                                                                                                                                                                                                      Current_rule_data
WITH location := 265,75,489,99
                                                                                                              WITH items [22] := pushbutton 49
WITH items [23] := pushbutton 50
                                                                                                                                                                                                                       INSTANCE listbox 6 ISA listbox
   WITH menus [1] := UNDETERMINED
                                                                                                                                                                                                                         WITH show current instance := TRUE
```

WITH source := attribute_of_building_element OF building_element_attribute WITH destination := attribute_of_building_element OF Current_rule_data WITH location := 265,120,489,144

INSTANCE fistbox 8 ISA fistbox
WITH show current instance: "TRUE
WITH source: "label OF
Menus_unclassified B_comp
WITH destination == current_menu_B_comp OF
Menus_manipulation_data
WITH location: "16.91.165.315

INSTANCE fistbox 9 ISA fistbox
WITH show current instance: "TRUE
WITH source: "label OF Menus_group
WITH destination: "current_menu_group OF
Menus_manipulation_dats
WITH location: "185,90,335,205

INSTANCE listbox 10 ISA listbox
WITH show current instance: TRUE
WITH source: label OF Menus group_items
WITH destination: current_menu_B_comp OF
Menus_manipulation_data
WITH location: 350,90,495,255

INSTANCE listbox !1 ISA listbox
WITH show current instance := TRUE
WITH source := project_name OF Current_projects
WITH destination := project_name
WITH location := 11,140,160,165

INSTANCE listbox 12 ISA listbox
WITH show current instance = TRUE
WITH show current instance = TRUE
WITH destination := another_project_name OF
NEW_to_make a new_project
WITH location := 10.95.160.115

INSTANCE listbox 13 ISA listbox
WITH show current instance: "TRUE
WITH source: "building element OF Building
element
WITH destination: "building element OF
NEW to make a new project
WITH location: "35,95,460,180

INSTANCE listbox 14 ISA listbox
WITH show current instance: TRUE
WITH source: stribute of building_element OF
building_element_strrbute
WITH location:= 115,255,460,330

INSTANCE listbox 15 ISA listbox
WITH show current instance := TRUE
WITH source := building_element OF Building_element
WITH location := 265,200,465,285

INSTANCE listbox 16 ISA listbox
WITH show current instance: "TRUE
WITH source: "building_system OF
NEW_building_systems
WITH destination: "building_system OF
NEW_to_make_a_new_project
WITH location: "30,150,230,270

INSTANCE listbox 17 ISA listbox
WITH show current instance: TRUE
WITH source: discipline OF
NEW_design_disciplines
WITH destination: design_discipline OF
NEW_to_make_a_new_project
WITH location: 265,65,460,87

INSTANCE listbox 18 ISA listbox WITH show current instance: TRUE WITH source: = building_element OF GENERAL_Building_element WITH destination: = GENERAL_Building_component WITH location: = 21.76.400,160

INSTANCE listbox 19 ISA listbox
WITH show current instance:= TRUE
WITH source:= attribute of building_element OF
GENERAL_building_element_attribute
WITH location:= 110.205.400,280

INSTANCE menu c_Foundation ISA menu WITH label := "Foundation" WITH items [1] := UNDETERMINED

INSTANCE menu c_Floor ISA menu WITH label := "Floor" WITH items [1] := UNDETERMINED

INSTANCE menu c_Wall ISA menu WITH label := "Wall" WITH items [1] := UNDETERMINED

INSTANCE menu c_Roof ISA menu WITH label := "Roof" WITH items [1] := UNDETERMINED

INSTANCE menu c_Opening ISA menu WITH label := "Opening" WITH items [1] := UNDETERMINED

INSTANCE menu c_Protection ISA menu WITH label := "Protection" WITH items [1] := UNDETERMINED

INSTANCE menu c_Special ISA menu WITH label := "Special" WITH items [1] := UNDETERMINED

INSTANCE menu c_Finishing ISA menu WITH label := "Finishing" WITH items [1] := UNDETERMINED

INSTANCE menu c_Mechanical ISA menu WITH label := "Mechanical" WITH items [1] := UNDETERMINED

INSTANCE menu c_Electrical ISA menu WITH label := "Electrical" WITH items [1] := UNDETERMINED

INSTANCE menu c_Pier ISA menu WITH label := "Pier" WITH items [1] := UNDETERMINED

INSTANCE menu c_Wood_Pier_Foundation ISA menu WITH label := "Wood_Pier_Foundation"

WITH items [1] := meau item c_Circular_Wood_Pier_Foundation

INSTANCE menu 13 ISA menu WITH label := "menu 13" WITH items [1] := menu item 1

INSTANCE menu item 1 ISA menu item WITH label := "menu item 1"

INSTANCE menu item
c_Circular_Wood_Pier_Foundation ISA menu item
WITH label := "Circular_Wood_Pier_Foundation"

INSTANCE promptbox 1 ISA promptbox
WITH font: = "System"
WITH justify IS center
WITH frame: = TRUE
WITH show current: = TRUE
WITH strachment: = graph_name OF Building
element
WITH location: = 100,100,210,125

INSTANCE promptbox 2 invisible ISA promptbox WITH pen color := 255,255,255 WITH fill color := 0,0,255 WITH font := "System" WITH justify IS center WITH frame := TRUE WITH show current := TRUE WITH statchment := new_component_name WITH location := 0,0,0,0

INSTANCE promptbox 3 invisible2 ISA promptbox WITH pen color := 255,255,255
WITH fill color := 0,0,255
WITH font := "System"
WITH justify IS center
WITH grame := TRUE
WITH show current := TRUE
WITH stackment := new_attribute_name
WITH location := 0,0,0

INSTANCE promptbox 4 ISA promptbox

WITH pen color := 255,00
WITH fill color := 255,255,255
WITH font := "System"
WITH finne := TRUE
WITH seroil := TRUE
WITH show current := TRUE
WITH show current := TRUE
WITH stachment := cause_of_the_effect OF
AB_general_rules
WITH location := 15,230,240,275

INSTANCE promptbox 5 ISA promptbox WITH pen color := 255,00 WITH fill color := 255,205,255. WITH font := "System" WITH justify IS center WITH finne := TRUE WITH show current := TRUE WITH struckment := percent_change OF AB_general_rules WITH location := 110,55,155,80

INSTANCE promptbox 6 ISA promptbox
WITH pen color: = 255,0,0
WITH fill color: = 255,255,255
WITH foat: = "System"
WITH justify IS center
WITH frame: = TRUE
WITH show current: = TRUE
WITH stackment: = result_in_major_modification
OF AB_general_rules
WITH location: = 190,295,240,320

INSTANCE promptbox 7 ISA promptbox WITH pen color := 255,0,0 WITH fill color := 255,255.55 WITH font := "System" WITH justify IS center WITH frame := TRUE WITH show current := TRUE WITH stanchment := rule_ignored OF AB_general_rules WITH location := 190.345,240,370

INSTANCE promptbox & ISA promptbox
WITH pen color := 128,0,0
WITH fill color := 255,255,255
WITH float := "System"
WITH justify IS center
WITH frame := TRUE
WITH show current := TRUE
WITH attachment := new_name_of_menu_group OF
Menus_manipulation_data
WITH location := 185,300,335,325

INSTANCE promptbox 9 ISA promptbox WITH foat: = "System" WITH justify IS center WITH frame: = TRUE WITH show current: = TRUE WITH stackment: = project_name WITH location: = 10,135,160,165

INSTANCE pushbutton 1 ISA pushbutton WITH label: = "Building Components" WITH attribute attachment: = Building Components WITH location: = 10,105,160,130

INSTANCE pushbutton 2 ISA pushbutton WITH label: "EXXIT" WITH attribute attachment: "exit OF application WITH location: "10,185,160,215

INSTANCE pushbutton 3 ISA pushbutton WITH label := "Delete" WITH location := 215,100,290,125

INSTANCE pushbutton 4 ISA pushbutton WITH label := "New Component" WITH location := 290,100,410,125

INSTANCE pushbutton 5 ISA pushbutton WITH label: "END" WITH attribute attachment: "back to current project display WITH location: = 425,5,500,35

INSTANCE pushbutton 6 ISA pushbutton WITH label := *** WITH location := 425,90,500,135 INSTANCE pushbutton 8 ISA pushbutton WITH label := "Return" WITH location := 175,250,395,275

INSTANCE pushbutton 10 invisible ISA pushbutton WITH label := "Done"
WITH attribute attachment := done of new component WITH location := 0.0.0.0

INSTANCE pushbutton 11 invisible ISA pushbutton WITH label:= "Cancel" WITH location:= 0,0,0,0

INSTANCE pushbutton 12 invisible2 ISA pushbutton WITH label := "Done" WITH attribute attackment := done of new attribute WITH location := 0.0 0.0

INSTANCE pushbutton 13 invisible2 ISA pushbutton WITH label: "Next Attribute" WITH attribute attackment: "next new attribute WITH location: "0.0.0

INSTANCE pushbutton 14 ISA pushbutton WITH label := "Delete" WITH location := 180,250,235,270

INSTANCE pushbutton 15 ISA pushbutton WITH label := "New" WITH location := 180,270,235,290

INSTANCE pushbutton 16 invisible2 ISA pushbutton WITH label: ""OK."
WITH attribute attachment: "ok of attribute name WITH location: "0.00.0

INSTANCE pushbutton 17 ISA pushbutton WITH label := "General Rules" WITH attribute attachment := General Rules WITH location := 10,165,160,190

INSTANCE pushbutton 18 ISA pushbutton WITH label := "First" ITH attribute attachment := action OF rules navigate 2 IS first WITH location := 400,240,490,265

INSTANCE pushbutton 19 ISA pushbutton WITH label := "Next" WITH attribute attachment := action OF navigate IS next WITH location := 400,265,490,290

INSTANCE pushbutton 20 ISA pushbutton
WITH label: "Previous"
WITH attribute attachment: "action OF navigate IS
previous
WITH location: "400,290,490.315

INSTANCE pushbutton 21 ISA pushbutton WITH label := "Last"
WITH stribute attachment := action OF navigate IS last
WITH location := 400,315,490,340

INSTANCE pushbutton 22 ISA pushbutton WITH label := "END" WITH attribute attachment := back_to_current_project_display WITH location := 280,400,480,425

INSTANCE pushbutton 23 ISA pushbutton WITH label := "Update Rule" WITH attribute attachment := update rule WITH location := 30,400,230,425

INSTANCE pushbutton 24 ISA pushbutton WITH label: ""New Rule" WITH attribute strachment: "new rule WITH location: "= 400,355,490,380

INSTANCE pushbutton 25 ISA pushbutton WITH location := 10,10,210,35

INSTANCE pushbutton 26 invisible2 ISA pushbutton WITH label: "Done" WITH attribute attackment: "done of ID attribute WITH location: "0.9.0.0

INSTANCE pushbutton 26 ISA pushbutton WITH label := "Menus"

WITH attribute attachment := Menus WITH location := 10,135,160,160

INSTANCE pushbutton 27 ISA pushbutton WITH label:= "END" WITH attribute attachment:= back_to_current_project_display WITH location:= 145,360,500,415

INSTANCE pushbutton 28 ISA pushbutton WITH label := "Add to the Menu Group" WITH stribute stuckment := add build comp OF Menus_manipulation_data WITH (ocation := 10.385, 170,410

INSTANCE pushbutton 29 ISA pushbutton WITH label: = "Del. Component" WITH attribute attachment: = delete build comp OF Menus_manipulation_data WITH location: = 350.325.495.350

INSTANCE pushbutton 30 ISA pushbutton WITH label: = "Add A Menu Group" "Add The attribute attachment: = add menu group OF Menus_manipulation_data WITH location:= 185,360,335,385

INSTANCE pushbutton 31 ISA pushbutton
WITH label: "Delete Menu Group"
WITH attribute attachment: "delete menu group OF
Menus_manipulation_data
WITH location: "185,260,335,285

INSTANCE pushbutton 32 ISA pushbutton WITH label := "Rename Menu Group" WITH attribute strachment := rename menu group OF Menus_manipulation_data WITH location := 185.735.335.410

INSTANCE pushbutton 33 ISA pushbutton
WITH label: " "EXXI"
WITH attribute attachment: " exit OF application
WITH location: " 10,195,160,220

INSTANCE pushbutton 34 ISA pushbutton WITH label: = "New Project"
WITH attribute attachment: = New_project WITH display attachment: = NEW_project_start_display
WITH location: = 10,105,160,135

INSTANCE pushbutton 35 tSA pushbutton WITH label: = "Existing Project" WITH attribute attachment: = Current_project WITH display attachment: = CURENT_project_start_display WITH location: = 10,145,160,175

INSTANCE pushbutton 36 ISA pushbutton WITH label := "EXIT" WITH attribute attachment := exit OF application WITH location := 10,195,160,220

INSTANCE pushbutton 37 ISA pushbutton WITH label := "OK" WITH location := 10,170,160,195

INSTANCE pushbutton 38 ISA pushbutton WITH label := "OK"
WITH location := 10.170.160.195

INSTANCE pushbutton 39 ISA pushbutton WITH label := "from Another Project" WITH display attachment := NEW from another project display WITH location := 10 105 160 140

INSTANCE pushbutton 40 ISA pushbutton WITH label: ""from Systems" WITH attribute attachment: "from systems OF NEW_10_make_a_new_project WITH location: "10,145,160,180

INSTANCE pushbutton 41 ISA pushbutton
WITH label: = "from Components"
WITH attribute attachment: = from components OF
NEW_to_make_a_new_project
WITH location := 10,185,160,220

INSTANCE pushbutton 42 ISA pushbutton WITH label := "Modify components"

WITH location := 10,120,160,145

INSTANCE pushbutton 43 ISA pushbutton WITH label := "Modify Gen. Rules" WITH location := 10,145,160,170

INSTANCE pushbutton 44 ISA pushbutton WITH label := "Config. Menus" WITH location := 10,170,160,195

INSTANCE pushbutton 45 ISA pushbutton WITH label:="Back" WITH display attachment:= NEW_project_selection_display WITH location:= 10,195,160,220

INSTANCE pushbutton 46 ISA pushbutton WITH label: = "OK" WITH attribute attachment: = from systems OF NEW to make a new project WITH location: = 20,195,160,220

INSTANCE pushbutton 47 ISA pushbutton
WITH label := "Add selected component to the new
project"
WITH attribute attachment := add component to the
new project OF NEW to make a new project
WITH location := 20,355,335,380

INSTANCE pushbutton 48 ISA pushbutton WITH label: "Finish" WITH attribute attachment: "finish components of the new project OF NEW_to_make_a_new_project WITH location: "345,355,475,380"

INSTANCE pushbutton 49 ISA pushbutton WITH label: "Add to the new project" WITH attribute attachment: "add system to the new project OF NEW_to_make_a_new_project WITH location: "255.330.475.355

INSTANCE pushbutton 50 ISA pushbutton WITH label: "Finish"
WITH attribute attachment: "finish systems of the new project OF NEW_to_make_a_new_project WITH location: = 255,360,475,385

INSTANCE pushbutton 51 ISA pushbutton WITH label: ""Ignore highlighted compon." WITH attribute attachment: "ignore highlighted component OF NEW_to_make_a_new_project WITH location: "265,285,465,310

INSTANCE pushbutton 52 ISA pushbutton WITH label := " " WITH attribute attachment := get from general database WITH location := 425 40.500.85

INSTANCE pushbutton 53 ISA pushbutton
WITH label: "Add the Component"
WITH attribute attachment: "add component from
get from general datab
WITH location: "20,285,215,305

INSTANCE pushbutton 54 ISA pushbutton
WITH label: ""Return"
WITH attribute attachment: "return from _get from
general datab
WITH location: = 215 285 400 305

INSTANCE pushbutton 55 ISA pushbutton WITH label: ""Add the Component and its related General Rules" WITH attribute attachment: "add component and general rule

WITH location := 20,305,400,325

INSTANCE radiobutton group I ISA radiobutton group
WITH pen color := 0,0,0

WITH pen color: = 0,0,0
WITH fill color: = 192,192,192
WITH firme: = TRUE
WITH group label: = ""
WITH show current: = TRUE
WITH attachment: = discipline OF input data
WITH location: = 235,145,355,295

INSTANCE radiobutton group 2 ISA radiobutton group
WITH pen color := 0,0,0

WITH pen color := 0,0,255 WITH fill color := 192,192,192 WITH justify IS left WITH fill color := 192,192,192 WITH justify IS left WITH frame := TRUE WITH group label := "attribute type" WITH show current := TRUE WITH foat := "System"
WITH text := "Type the name of WITH foot := "System"
WITH text := "Select Linked Component" ig componen WITH attachment := attribute type OF input data WITH location := 355,145,495,225 WITH location := 0.0.0.0 WITH location := 25,300,225,320 INSTANCE textbox 18 invisible2 ISA textbox INSTANCE radiobutton group 3 ISA radiobutton INSTANCE textbox \$ ISA textbox WITH pen color := 0,0,0 WITH fill color := 192,192,192 WITH pen color := 0,0,255
WITH fill color := 192,192,192 group
WITH pen color := 0,0,0
WITH fill color := 192,192,192
WITH frame := TRUE WITH justify IS left WITH justify IS left
WITH font := "System"
WITH text := "Graph name"
WITH location := 15,105,100,125 WITH font := "System"
WITH text := "Type the attribute name of the building component and press OK"
WITH location := 0.0.0.0 WITH group label := " WITH show current := TRUE
WITH attachment := discipline OF input data WITH location := 260,220,395,385 INSTANCE textbox 9 ISA textbox INSTANCE textbox 19 invisible2 ISA textbox WITH pen color := 0,0,255 WITH fill color := 192,192,192 WITH pen color := 0,0,0 WITH fill color := 192,192,192 INSTANCE table I ISA table WITH attachment := building_element_attribute WITH justify IS left WITH justify IS left WITH columns [1]:= column 1 WITH foot := "System"
WITH text := "Select Linked Attribute" WITH font := "System"
WITH text := "Fill the rest of the attribute data WITH columns [2] := column 2 WITH columns [3]:= column 3 WITH location := 25,345,185,365 (discipline, type)* WITH columns [4] := column 4 WITH columns [5] := column 5 WITH location := 0.0.0.0 INSTANCE textbox 10 ISA textbox WITH columns (5 7 := Column WITH heading := TRUE WITH heading height := 50 WITH fill color := 0,255,255 WITH pen color := 0,0,255 WITH fill color := 192,192,192 INSTANCE textbox 20 ISA textbox WITH pen color := 255,255,255 WITH fill color := 128,0,0 WITH justify IS left WITH column lines := TRUE WITH row lines := TRUE WITH font := "System"
WITH text := "Current Linked Component" WITH justify IS center
WITH foot: "Times New Roman"
WITH foot style IS bold, italic CF FALSE, underline WITH frame := TRUE WITH location := 260,300,460,320 WITH show current instance := TRUE WITH location := 10,45,590,240 CF FALSE, strikeout CF FALSE WITH font size := 18 INSTANCE textbox 11 ISA textbox WITH pen color := 0,0,255 WITH fill color := 192,192,192 WITH frame := TRUE WITH text := "GENERAL RULE INSTANCE textbox 1 ISA textbox WITH pen color := 255,255,255 WITH fill color := 0,0,128 WITH justify IS left
WITH font: "System"
WITH tent: "Current Linked Attribute"
WITH location: = 260,345,420,365 CONFIGURATION" WITH location := 5,5,500,45 WITH justify 1S center WITH font := "Times New Roman" INSTANCE textbox 23 ISA textbox WITH pen coior := 0,0,6 WITH fill color := 192,192,192 WITH font style IS bold, italic, underline CF FALSE, strikeout CF FALSE INSTANCE textbox 12 ISA textbox WITH justify IS left WITH font size := 12 WITH pen color := 0,0,255 WITH fill color := 192,192,192 WITH font := "System"
WITH text := "then the discipline"
WITH location := 15,170,140,190 WITH text := "CONFIGURATION MODULE* WITH justify IS left WITH location := 10,30,155,80 WITH font := "System"
WITH text := "Current Type" INSTANCE textbox 2 ISA textbox WITH location := 355,225,445,250 INSTANCE textbox 24 ISA textbox WITH pen color := 255,255,255 WITH fill color := 128,0,0 WITH pen color := 0,0,0 WITH fill color := 192,192,192 INSTANCE textbox 13 ISA textbox WITH pen color := 0,0,255 WITH fill color := 192,192,192 WITH justify IS left WITH justify IS center WITH justify IS left WITH font := "Times New Roman"
WITH font style IS bold, italic CF FALSE, underline
CF FALSE, strikeout CF FALSE WITH font := "System"
WITH text := "If such a change results in major redesign, please type YES"
WITH location := 10,285,190,335 WITH font := "System" WITH foot size := 12 WITH text := "Current WITH frame := TRUE
WITH text := "Configuration of Building Components INSTANCE textbox 26 ISA textbox WITH location := 355,250,425,290 WITH pen color := 0,0,0 and Attributes WITH location := 5,5,420,35 INSTANCE textbox 14 ISA textbox WITH fill color := 192,192,192 WITH pen color := 0,0,0 WITH fill color := 192,192,192 WITH justify IS left WITH font := "System"
WITH text := "Select a Building Component" INSTANCE textbox 3 ISA textbox WITH pen color := 0,0,255 WITH fill color := 192,192,192 WITH justify IS center WITH foat := "System" WITH text := "Review WITH location := 270.60.465.80 WITH justify IS left WITH font := "System"
WITH text := "List of Building Components" INSTANCE textbox 27 ISA textbox WITH location := 430,95,495,130 WITH pen color := 0,0,0 WITH fill color := 192,192,192 WITH location := 15,50,205,70 INSTANCE textbox 15 ISA textbox WITH justify IS left WITH font := "System"
WITH text := "Select a Component Attribute" INSTANCE textbox 4 ISA textbox WITH pen color := 0,0,255 WITH fill color := 192,192,192 WITH pen color := 0,0,255 WITH fill color := 192,192,192 WITH justify IS left
WITH font: ""Times New Roman"
WITH font style IS bold, italic CF FALSE, underline WITH location := 270,105,465,125 WITH justify IS left
WITH font: " "System"
WITH text: " "Current Building Components"
WITH location: = 215,50,410,70 INSTANCE textbox 28 ISA textbox WITH pen color := 0,0,255 WITH fill color := 192,192,192 CF FALSE, strikeout CF FALSE WITH font size := 12 WITH justify IS left WITH text := "Summary of Attributes of the Building WITH font := "System"
WITH text := "# of Current Related Rules:" INSTANCE textbox 5 ISA textbox WITH location := 10,10,355,35 WITH pen color := 0,0,255 WITH fill color := 192,192,192 WITH location := 275,155,455,175 WITH justify IS left
WITH font := "System"
WITH text := "Building Component Attributes" INSTANCE textbox 16 ISA textbox INSTANCE textbox 29 ISA textbox WITH pen color := 128,0,0 WITH fill color := 192,192,192 WITH pen color := 0,0,255 WITH fill color := 192,192,192 WITH location := 15,150,220,170 WITH justify IS left
WITH font := "Times New Roman" WITH justify IS left WITH font: = "System"
WITH text:= "Current Related Rules #:"
WITH location:= 275,180,445,200 INSTANCE textbox 6 ISA textbox WITH font style IS bold, italic, underline CF FALSE, strikeout CF FALSE WITH pen color := 0,0,255 WITH fill color := 192,192,192 WITH foot size := 12 WITH text := " " WITH justify IS left WITH font := "System"
WITH text := "Current Attribute" WITH location = 355,10,590,35 INSTANCE textbox 30 ISA textbox WITH pen color := 0,0,0 WITH fill color := 192,192,192 WITH location := 15,240,130,260 INSTANCE textbox 17 invisible ISA textbox WITH justify IS left WITH font := "System" WITH pen color := 0,0,0 WITH fill color := 192,192,192

INSTANCE textbox 7 ISA textbox

WITH text := "of the following Building Component" WITH fill color := 192 192 192 WITH justify IS center
WITH justify IS center
WITH font: "System"
WITH text: "Type Added or Rena;
WITH location:= 185,325,335,360 WITH location := 10,125,245,145 INSTANCE textbox \$1 ISA textbox WITH pen color := 0,0,0 WITH fill color := 192,192,192 INSTANCE textbox 31 ISA textbox ed Name above WITH pen color := 0.0,255 WITH fill color := 192,192,192 WITH justify IS left WITH font := "System"
WITH text := "Select project"
WITH location := 40,75,140,95 WITH justify IS left INSTANCE textbox 42 ISA textbox WITH pen color: = 0,0,255
WITH fill color:= 192,192,192
WITH justify IS center
WITH font:= "System"
WITH text:= "Selected Building Component"
WITH location:= 355,260,490,295 WITH font := "System"
WITH text := "Related General Rule" WITH location := 405,205,495,240 INSTANCE textbox 52 ISA textbox WITH pen color := 255,255,255 WITH fill color := 0,0,128 INSTANCE textbox 32 ISA textbox WITH pen color := 0,0,0
WITH fill color := 192,192,192
WITH fill color := 192,192,192
WITH justify 1S left
WITH fost := "System"
WITH text := "Affected Discipline"
WITH location := 265,205,390,225 WITH justify IS center
WITH foot := "Times New Roman"
WITH foot style IS bold, italic CF FALSE, underline
CF FALSE, strikeout CF FALSE **INSTANCE** textbox 43 ISA textbox WITH pen color := 0,0,0 WITH fill color := 192,192,192 WITH fost size := 12
WITH text := "Select Participating Disciplines"
WITH location := 15,10,155,55 WITH justify IS center WITH font := "System" INSTANCE textbox 33 ISA textbox WITH text := "Double click to select a project then WITH pen color := 0,0,0 WITH filt color := 192,192,192 click OK* INSTANCE textbox 53 ISA textbox WITH location := 10,100,165,140 WITH pen color := 255,255,255 WITH fill color := 128,0,0 WITH justify IS left WITH int cook: "128,0.0
WITH justify IS center
WITH foot: "Times New Roman"
WITH foot style IS bold, italic CF FALSE, underline
CF FALSE, strikeout CF FALSE INSTANCE textbox 44 ISA textbox WITH pen color:= 255,255,255 WITH fill color:= 0,0,128 WITH font := "System"
WITH text := "If you need this rule to be ignored, use type YES" WITH location := 10,340,180,375 WITH justify IS center
WITH font := "Times New Roman WITH foot size := 14 INSTANCE textbox 34 ISA textbox WITH pen color := 255,255,255 WITH fill color := 128,0,0 WITH font style IS bold, italic CF FALSE, underline CF FALSE, strikeout CF FALSE WITH frame := TRUE WITH text := "Assemble the New Project from the WITH foot size := 16 WITH justify IS center
WITH font := "Times New Roman"
WITH font style IS bold, italic CF FALSE, underline WITH text := "Configure Existing Project"
WITH location := 15.15.150.85 WITH location := 5,5,490,35 **INSTANCE** textbox 54 ISA textbox WITH pen color := 0,0,0
WITH fill color := 192,192,192
WITH justify IS center CF FALSE, strikeout CF FALSE INSTANCE textbox 45 ISA textbox WITH foot size := 16 WITH pen color := 255,255,255 WITH fill color := 0,0,128 WITH frame := TRUE
WITH text := "Configuration of Menu of Building WITH font := "System"
WITH text := "Double click to select a building WITH justify IS center WITH font := "Times New Roman"
WITH font style IS bold, italic CF FALSE, underline WITH location := 5,5,500,40 CF FALSE, strikeout CF FALSE WITH location := 105,65,395,85 INSTANCE textbox 36 ISA textbox WITH font size := 12 WITH pen color := 0,0,0 WITH fill color := 192,192,192 WITH text := "What would you WITH location := 15,25,145,75 d you like to configure? INSTANCE textbox 55 ISA textbox WITH pen color := 0,0,0 WITH fill color := 192,192,192 WITH justify IS center WITH foot := "System"
WITH text := "Unclassified Building Components" WITH justify IS left **INSTANCE** textbox 46 ISA textbox WITH font := "System"
WITH text := "Selected Component" WITH justify IS left WITH font := "System"
WITH text := "textbox 46" WITH location := 15,55,160,90 WITH location := 30,215,110,250 INSTANCE textbox 37 ISA textbox WITH location := 210.65.360.140 WITH pen color := 0,0,0 WITH fill color := 192,192,192 INSTANCE textbox 57 ISA textbox INSTANCE textbox 47 ISA textbox WITH pen color := 0,0,0 WITH pen color := 255,255,255 WITH fill color := 0,0,128 WITH justify IS center WITH fill color := 192,192,192 WITH font := "System"
WITH text := "List of Menu Groups" WITH justify IS left WITH font := "System WITH justify IS center WITH text := "Attributes WITH location := 200.55.320.90 WITH font := "Times New Roman"
WITH font style IS bold, italic CF FALSE, underline of the Component*
WITH location := 35,270,115,325 **INSTANCE** textbox 38 ISA textbox CF FALSE, strikeout CF FALSE WITH pen color := 0,0,0 WITH fill color := 192,192,192 WITH font size := 16 WITH text := "Assemble New Project"
WITH location := 10,20,155,75 INSTANCE textbox 58 ISA textbox WITH justify IS center WITH pen color := 255,255,255 WITH font: " "System"
WITH text: = "Components of Current Menu Group"
WITH location: = 350,50,490,85 WITH fill color := 128,0,0 INSTANCE textbox 48 ISA textbox WITH justify IS center WITH font := "Times New Roman" WITH pen color := 0,0,0 WITH fill color := 192,192,192 WITH foot style IS bold, italic CF FALSE, underline CF FALSE, strikeout CF FALSE INSTANCE textbox 39 ISA textbox WITH justify IS center WITH pen color := 0,0,255 WITH fill color := 192,192,192 WITH font: " "System"
WITH text: " "Type the project same then click OK"
WITH location: "10,100,165,135 WITH font size := 14 WITH frame := TRUE WITH justify IS center WITH text := "Assemble the New Project from WITH font := "System"
WITH text := "Selected Menu Group" **Building Systems** INSTANCE textbox 49 ISA textbox WITH pen color := 255,255,255 WITH fill color := 0,0,128 WITH location := 5.5.490.35 WITH location := 185,210,335,230 INSTANCE textbox 59 ISA textbox INSTANCE textbox 40 ISA textbox WITH justify IS center WITH font := "Times New Roman" WITH pen color := 0,0,0 WITH fill color := 192,192,192 WITH pen color := 0,0,255 WITH font style IS bold, italic CF FALSE, underline CF FALSE, strikeout CF FALSE WITH fill color := 192,192,192 WITH justify IS center WITH IN COMM .- 15015000 WITH INSINITY IN COMMON TO THE WITH font := "System"
WITH text := "Selected Building Component" WITH font := "System" WITH text := "Double click to select a building WITH font size := 14 WITH text := "Select an Assembly Method" WITH location := 15 320 165 355 WITH location := 20.15,145.80 WITH location := 50.115.215.150

INSTANCE textbox 50 ISA textbox

WITH pen color := 255,255,255 WITH fill color := 0,0,128

CF FALSE, Strikeout CF FALSE

WITH font := "Times New Roman"
WITH font style IS bold, italic CF FALSE, underline

WITH text := "Assemble from another project"
WITH location := 25,15,140,60

WITH justify IS center

WITH font size := 12

INSTANCE temp invisible textbox 41 ISA textbox

WITH pen color := 0,0,255 WITH fill color := 192,192,192

WITH location := 0,0,0,0

WITH pen color := 0,0,255

WITH justify IS center
WITH font := "System"
WITH text := "Write the new name"

INSTANCE textbox 41 ISA textbox

INSTANCE textbox 60 ISA textbox

WITH location := 280,115,450,135

INSTANCE textbox 61 ISA textbox

WITH pen color := 0,0,0

WITH font := "System"
WITH text := "Selected Building System"

WITH pen color := 0,0,0 WITH fill color := 192,192,192

WITH justify IS left

WITH justify IS center WITH font := "System'
WITH frame := TRUE WITH font := "System"
WITH text := "Components in the bi
WITH location := 275,165,460,202 WITH location := 455,155,490,175 WITH dipped := TRUE
WITH dipped := TRUE
WITH stachment := building_element OF nts in the building system INSTANCE valuebox 10 ISA valuebox WITH pen color := 0,0,255 WITH fill color := 192,192,192 WITH location := 215,70,410,95 INSTANCE textbox 62 ISA textbox WITH pen color := 0,0,0 WITH fill color := 192,192,192 WITH justify IS left INSTANCE valuebox 2 ISA valuebox WITH font := "System" WITH frame := TRUE WITH pen color := 128,0,0 WITH fill color := 192,192,192 WITH justify IS center
WITH font := "System"
WITH text := "Description of the highlighted building WITH clipped := TRUE WITH justify IS center WITH attachment := instance number OF rules WITH font := "System" navigate 2 WITH location := 455,180,490,200 WITH location := 40,275,225,312 WITH frame :- TRUE WITH clipped := TRUE WITH attachment := attribute of building element INSTANCE textbox 63 ISA textbox INSTANCE valuebox 11 ISA valuebox WITH pen color := 0,0,255 WITH fill color := 192,192,192 WITH justify IS center WITH pen color := 255,255,255 WITH fill color := 0,128,128 OF current_data WITH location := 15 260 170 285 WITH justify IS center WITH font := "System" INSTANCE valuebox 3 ISA valuebox WITH font := "System" WITH frame := TRUE WITH text := "Double click to select a design WITH pen color := 128,0,0 WITH fill color := 192,192,192 WITH clipped := TRUE
WITH attachment := affected_discipline OF WITH location := 30.60.255.95 WITH justify IS center WITH frame := TRUE AB_general_rules WITH location := 155,170,240,190 INSTANCE textbox 64 ISA textbox WITH clipped := TRUE
WITH attachment := linked_building_element OF WITH pen color := 0,0,0 WITH fill color := 192,192,192 INSTANCE valuebox 13 ISA valuebox current_data
WITH location := 260,320,480,345 WITH pen color := 128,0,0 WITH fill color := 192,192,192 WITH justify IS cente WITH font := "System"
WITH text := "From WITH justify IS center WITH foot := "System" WITH frame := TRUE INSTANCE valuebox 4 ISA valu WITH pen color := 128,0,0 WITH fill color := 192,192,192 WITH location := 430,45,495,80 WITH clipped := TRUE
WITH attachment := current menu_B comp OF WITH justify IS center INSTANCE textbox 65 ISA textbox WITH pen color := 0,0,0 WITH fill color := 192,192,192 WITH font := "System"
WITH frame := TRUE Menus_manipulation_data WITH location := 10,355,170,380 WITH clipped := TRUE
WITH attachment :=
linked_attribute_of_building_element OF current_data
WITH location := 260,365,480,390 WITH justify IS left WITH font := "System"
WITH text := "Selected Component INSTANCE valuebox 14 ISA valuebox WITH pen color := 128,0,0 WITH fill color := 192,192,192 WITH location := 20,170,100,205 WITH justify IS center WITH font := "System" WITH frame := TRUE INSTANCE textbox 66 ISA textbox INSTANCE valuebox 5 ISA valuebox INS I ANN.E LECTION 60 ISA textb
WITH pen color: = 0,00
WITH fill color: = 192,192,192
WITH justify IS left
WITH font: = "System"
WITH text: = "Attributes WITH pen color := 128,0,0
WITH fill color := 192,192,192
WITH justify IS center
WITH font := "System"
WITH frame := TRUE WITH clipped: "TRUE
WITH clipped: "TRUE
WITH stackment: "current menu_B_comp OF
Menus_manipulation_data
WITH location: "350,295,495,320 WITH clipped := TRUE
WITH attachment := designing_discipline OF of the Component WITH location := 20,215,100,270 INSTANCE valuebox 15 ISA va WITH pen color := 128,0,0 WITH fill color := 192,192,192 WITH location := 435,260,490,285 INSTANCE textbox 67 ISA textbox WITH pen color := 255,255,255
WITH fill color := 128,0,0
WITH justify IS center
WITH font := "Times New Roman" WITH justify IS center INSTANCE valuebox 6 ISA valuebox WITH foot := "System" WITH frame := TRUE WITH pen color := 128,0,0 WITH fill color := 192,192,192 WITH dipped := TRUE WITH font style IS bold, italic CF FALSE, underline WITH justify IS center WITH font := "System" WITH frame := TRUE WITH attachment := current_menu_group OF Menus_manipulation_data CF FALSE, strikeout CF FALSE WITH font size := 14 WITH location := 185,230,335,255 WITH frame := TRUE
WITH text := "Components is
WITH location := 5,5,415,35 WITH clipped := TRUE
WITH stachment := attribute_type OF current_data nts in the General Database^e INSTANCE valuebox 16 ISA valuebox WITH pen color := 0,0,255 WITH fill color := 192,192,192 WITH location := 465,225,490,250 INSTANCE textbox 68 ISA textbox INSTANCE valuebox 7 ISA valuebox WITH justify IS center WITH pen color := 0,0,0
WITH fill color := 192,192,192
WITH justify IS left WITH pen color := 0,0,255 WITH fill color := 192,192,192 WITH foot := "System"
WITH frame := TRUE WITH justify IS center WITH clipped := TRUE WITH font := "System"
WITH text := "Double Click to Select a Component" WITH font := "System"
WITH frame := TRUE WITH attachment := building_element OF NEW_to_make_a_new_project WITH location := 115,215,460,245 WITH location := 90,55,330,75 WITH dipped := TRUE WITH attachment := building_element OF INSTANCE textbox 69 ISA textbox Current rule data INSTANCE valuebox 17 ISA valuebox WITH pen color := 0,0,0 WITH fill color := 192,192,192 WITH pen color := 0,0,255 WITH fill color := 192,192,192 WITH location := 15,745,240,165 WITH justify IS left WITH justify IS center WITH font : " "System" INSTANCE valuebox 8 ISA valuebox WITH font := "System"
WITH text := "If a change of WITH pen color := 0,0,255 WITH frame := TRUE WITH fill color := 192,192,192 % occurs to the following attribute"
WITH location := 15,65,245,100 WITH justify IS center WITH font := "System" WITH frame := TRUE WITH clipped := TRUE WITH attachment := building_system OF NEW_to_make_a_new_project WITH location := 265,135,465,160 INSTANCE textbox 70 ISA textbox WITH dipped := TRUE WITH pen color := 0,0,0 WITH fill color := 192,192,192 WITH attachment := attribute_of_building_element OF Current_rule_data WITH location := 15,100,240,120 INSTANCE valuebox 18 ISA v WITH justify IS left
WITH font := "System"
WITH text := "need to be notified because it may WITH pen color := 0,0,255 WITH fill color := 192,192,192 INSTANCE valuebox 9 ISA value WITH justify IS left WITH pen color := 0,0,255 WITH fill color := 192,192,192 WITH font := "System"
WITH frame := TRUE WITH location := 15,193,245,230 WITH justify IS left WITH clipped := TRUE WITH attachment := description OF WITH font := "System"
WITH frame := TRUE
WITH clipped := TRUE INSTANCE valuebox 1 ISA valuebox WITH pen color = 128,0,0 WITH fill color = 192,192,192 NEW_building_systems WITH location := 35,315,230,370

WITH justify IS center

WITH fill color := 192,192,192

WITH attachment := number instances OF rules

INSTANCE valuebox 17 ISA valuebox
WITH pen color: = 0,0,255
WITH fill color: = 192,192,192
WITH fill color: = 192,192,192
WITH foat: = "System"
WITH frame: = TRUE
WITH clipped: = TRUE
WITH strachment: = building_system OF
NEW_to_make_a_new_project
WITH location: = 265,135,465,160

INSTANCE valuebox 18 ISA valuebox WITH pen color := 0,0,255 WITH fill color := 192,192,192 WITH flustify IS left WITH font := "System" WITH frame := "TRUE WITH filpped := TRUE WITH attachment := description OF NEW_building_systems WITH location := 35,315,230,370

INSTANCE valuebox 19 ISA valuebox
WITH pen color := 0,0,255
WITH dil color := 192,192,192
WITH justify IS center
WITH foot := "System"
WITH frame := TRUE
WITH dipped := TRUE
WITH attachment :=
GENERAL_Building_component
WITH location := 110,170,400,200

INSTANCE main window ISA window
WITH location: = 207,160,381,413
WITH memus [1]: = UNDETERMINED
WITH style IS moveable, sizeable, closeable
WITH title: = ""
WITH visible: = TRUE

INSTANCE configuration window ISA window WITH location: = 61,11,578,464 WITH menus [1]: = UNDETERMINED WITH style IS moveable, sizeable, closeable CF FALSE WITH title: = **
WITH visible: = FALSE

INSTANCE attribute window ISA window WITH location := 22,62,627,171
WITH menus [1]:= UNDETERMINED
WITH style Is moveable, sizeable, closeable
WITH title := ""
WITH visible := FALSE

INSTANCE get from general database window ISA window
WITH location := 192,114,617,490
WITH menus [1] := UNDETERMINED
WITH style IS moveable, sizeable, closeable
WITH title := * "
WITH visible := FALSE

! DEMON GROUP: affected_discipline OF AB general rules

DEMON 11
IF CONF(affected_discipline OF AB_general_rules)
>= 0
THEN FORGET discipline OF input data

! DEMON GROUP: button selected OF message box

DEMON 5
IF button selected OF message box 1 IS yes
THEN delete a component := TRUE

DEMON 10
IF button selected OF message box 3 IS yes
THEN delete of attribute := TRUE

! DEMON GROUP: sciected OF pushbutton

DEMON I
IF selected OF pushbutton 6
THEN text OF textbox 16: "building_element OF
current_data
AND output OF attribute window: "attributes table
display
AND visible OF attribute window: "TRUE

DEMON 2
IF selected OF pushbutton 8
THEN visible OF attribute window := FALSE

DEMON 4
IF selected OF pushbutton 3
THEN put up OF message box 1 := TRUE

DEMON 6
IF selected OF pushbutton 4
THEN location OF border 6 invisible := 15,50,410,95
AND location OF textbox 17 invisible := 25,55,160,90
AND location OF promptbox 2 invisible := 170,60,340,85
AND location OF pushbutton 10 invisible := 355,50,410,70
AND location OF pushbutton 11 invisible :=

355,75,410,95

AND location OF pushbutton 12 invisible2 := 0,0,0,0

AND location OF promptbox 3 invisible2 := 0,0,0,0

AND location OF border 7 invisible2 := 0,0,0,0

AND location OF textbox 18 invisible2 := 0,0,0,0

AND location OF textbox 19 invisible2 := 0,0,0,0

AND location OF textbox 19 invisible2 := 0,0,0,0

AND location OF pushbutton 13 invisible2 := 0,0,0,0

AND FORGET current data

AND FORGET building_element_attribute

DEMON 7
IF selected OF pushbutton 11 invisible
THEN location OF border 6 invisible:= 0,0,0,0
AND location OF tentbox 17 invisible:= 0,0,0,0
AND location OF promptbox 2 invisible:= 0,0,0,0
AND location OF pushbutton 10 invisible:= 0,0,0,0
AND location OF pushbutton 11 invisible:= 0,0,0,0

DEMON 8

IF selected OF pushbutton 15

THEN location OF pushbutton 16 invisible2:=
180,195,220,220

AND location OF prompthox 3 invisible2:=
25,195,220,220

AND location OF border 7 invisible2:=
15,150,235,290

AND location OF textbox 18 invisible2:=
25,125,220,190

AND location OF textbox 19 invisible2:=
25,225,220,260

AND FORGET attribute_of_building_element OF current_data

AND FORGET attribute_type OF current_data

AND FORGET tinked_building_element OF current_data

AND FORGET linked_building_element OF current_data

AND FORGET linked_building_element OF current_data

AND FORGET linked_building_element OF current_data

AND FORGET linked_attribute_of_building_element

OF current_data

DEMON 9
IF selected OF pushbutton 14
THEN put up OF message box 3 := TRUE

I DEMON GROUP: double clicked OF table

DEMON 3
IF double clicked OF table 1
THEN visible OF attribute window := FALSE
AND attribute_of_building_element OF current_data
:= attribute_of_building_element OF
building_element_attribute

END

C.3. MESSAGE MODULE

```
SVERSION35
                                                                   WITH data source STRING
                                                                                                                                            pen color OF valuebox 1 := 0,0,255
                                                                                                                                  IF been_read OF discipline_message = "Yes"

AND been_acted_upon OF discipline_message = "Yes"
                                                                    WITH userid STRING
 SLOCATIONS ARE PIXELS
                                                                    WITH password STRING
                                                                    WITH connection string STRING
                                                                    WITH connection prompting SIMPLE
                                                                                                                                            pen color OF valuebox 1 := 0,0,0
                                                                    INIT FALSE
                                                                                                                                           attachment OF valuebox 1 := message builder
 CLASS AB_general_rules
                                                                    WITH auto cor
                                                                                                                                  OF required_message
                                                                    INIT TRUE
  WITH AB general rules ID NUMERIC
                                                                    WITH action COMPOUND
                                                                                                                                   IF (general_specific OF discipline_message = "GENERAL") THEN
  WITH building element STRING
                                                                                                                                           BEGIN
                                                                   WITH records NUMERIC
                                                                                                                                           general rule message := TRUE
END
  WITH attribute_of_building_element STRING
                                                                   WITH transaction COMPOUND
  WITH affected_discipline STRING
                                                                    roliback
                                                                                                                                  IF (general_specific OF discipline_message = "SPECIFIC") THEN
  WITH percent change NUMERIC
WITH result in major modification STRING
WITH cause_of_the_effect STRING
                                                                   WITH append SIMPLE
INIT FALSE
                                                                                                                                           BEGIN
                                                                    WITH status STRING
                                                                                                                                             specific rule message := TRUE
                                                                   WITH show error SIMPLE
CLASS AB_specific_rules
WITH AB_specific_rules_id NUMERIC
WITH building_element STRING
                                                                   WITH default error handling SIMPLE
                                                                                                                                         END
                                                                    INTT TRUE
                                                                                                                                      END
                                                                   WITH error message STRING
                                                                                                                                    WITH instance number NUMERIC
  WITH building element id STRING
WITH attribute of building element
                                                                   WITH pative error NUMERIC
                                                                                                                                    WITH number instances NUMERIC
                                                                    WITH trace file STRING
  WITH related_class STRING
                                                                                                                                  CLASS required_message
WITH discipline name COMPOUND
                                                                 CLASS message box INHERITS add on WITH title STRING
  WITH related instance id STRING
WITH related attribute STRING
                                                                                                                                     Architecture
  WITH relationship STRING
                                                                   WITH text STRING
                                                                                                                                     HVAC
  WITH writting_date STRING
WITH relation COMPOUND
                                                                                                                                     Illumination,
                                                                   WITH variable text STRING
                                                                   WITH model style COMPOUND
                                                                                                                                     Envelope.
                                                                    app modal,
   Equal To,
                                                                                                                                     Structure,
   Less than,
More than
                                                                                                                                     Interior
                                                                    svs modal.
                                                                    not modal
                                                                                                                                    WITH message status COMPOUND
                                                                    INIT app modal
                                                                                                                                     All Messages,
CLASS border INHERITS add on, display item
                                                                   WITH icon COMPOUND
                                                                                                                                     InRead Yet
  WITH style COMPOUND
                                                                    exclamation,
                                                                                                                                     UnActed Upon
                                                                    information,
   picture frame
                                                                                                                                   WITH choosen_discipline STRING
                                                                    auestion
                                                                                                                                   WITH message builder STRING
   group,
shadow
                                                                    stop sign
                                                                                                                                  ATTRIBUTE NoS STRING
ATTRIBUTE Yes STRING
                                                                    INIT exclumation
   INIT picture frame
                                                                   WITH button COMPOUND
  WITH perspective COMPOUND
                                                                                                                                  ATTRIBUTE helping SIMPLE
                                                                    abort retry ignore,
   in,
                                                                                                                                   WHEN CHANGED
                                                                    ok cancel
                                                                                                                                    BEGIN
   INIT in
                                                                    retry cancel,
                                                                                                                                      visible OF helping window := TRUE
  WITH border width NUMERIC
                                                                    yes no.
                                                                                                                                      output OF helping window := helping display
  INIT I
                                                                    ves no cancel
                                                                                                                                    END
  WITH background color COLOR
                                                                                                                                  ATTRIBUTE show rule SIMPLE
  INIT 192 192 192
                                                                   WITH default button COMPOUND
                                                                                                                                   WHEN CHANGED
  WITH fill color COLOR
                                                                   one
                                                                                                                                    BEGIN
  INIT 192,192,192
                                                                   two,
 WITH highlight color COLOR
                                                                                                                                    **** in case of a Genreal rule
  INIT 255,255,255
                                                                    INIT one
                                                                                                                                  IF (general_specific OF discipline_message = "GENERAL") THEN
 WITH shadow color COLOR
                                                                   WITH button selected COMPOUND
  INIT 128,128,128
                                                                                                                                       BEGIN
                                                                                                                                        EXEC ISodbc 1 SQL select * from [AB-general-
                                                                    cancel
                                                                                                                                  rules] where AB_general_rules_ID =
:discipline_message.violated_rule_id
CLASS Current_projects
                                                                    ignore,
 WITH Project_name STRING
                                                                                                                                                  END SQL INTO AB_general_rules
                                                                   ok.
CLASS discipline_message
                                                                                                                                        visible OF helping window := TRUE
                                                                    retry.
 WITH changed_attribute STRING
WITH changed_class STRING
WITH changed_class STRING
WITH changed_instance_id STRING
WITH old_value STRING
                                                                                                                                        output OF helping window := general rule
                                                                  WITH put up SIMPLE
                                                                                                                                  display
                                                                                                                                       ÉND
                                                                 CLASS pavigate INHERITS add on
                                                                                                                                 ! **** in case of a specific rule
IF (general specific OF discipline_message =
"SPECIFIC") THEN
 WITH new_value STRING
                                                                   WITH class attachment CLASS REFERENCE
 WITH affected_attribute STRING
WITH affected_class STRING
                                                                   WITH action COMPOUND
                                                                    first,
 WITH affected_instance_id STRING
                                                                   previous.
                                                                                                                                       BEGIN
 WITH relationship STRING
                                                                                                                                        EXEC ISodbc 1 SQL select * from [AB-specific-
                                                                    next,
 WITH writing_date TIME
                                                                                                                                  rules] where AB_specific_rules_id = :discipline_message.violated_rule_id
                                                                    last
 WITH reading date TIME
WITH reading date TIME
WITH been read STRING
WITH been acted upon STRING
WITH id NUMERIC
                                                                    WHEN CHANGED
                                                                     BEGIN
                                                                                                                                                  END SQL INTO AB_specific_rules
                                                                      IF instance number OF navigate 1 ◆ 0 THEN
                                                                                                                                       visible OF specific rule window := TRUE output OF specific rule window := specific rule
 WITH discipline_to STRING
                                                                                                                                  display
                                                                  of the message color

IF been_read OF discipline_message = "Yes"
  WITH discipline from STRING
                                                                                                                                       END
 WITH sent_because_change_ID NUMERIC WITH general_specific STRING
                                                                 AND been_acted_upon OF discipline_message = "No"
                                                                                                                                   •••• end of method
 WITH rule_writing_date STRING
WITH violated_rule_id NUMERIC
WITH cause_of_the_effect STRING
                                                                 THEN
                                                                                                                                    END
                                                                           pen color OF valuebox 1 := 255,0,0
                                                                                                                                  ATTRIBUTE update of rule SIMPLE
                                                                 IF been_read OF discipline_message = "No"
AND been_acted_upon OF discipline_message = "No"
                                                                                                                                   WHEN CHANGED
                                                                                                                                    BEGIN
CLASS ISodbc INHERITS add on
```

```
! *** in case of a Genreal rule
                                                                              action OF ISodbc 1 IS connect := TRUE
                                                                                                                                                       title OF message window := CONCAT( "Reading
 IF (general_specific OF discipline_message = "GENERAL") THEN
                                                                                                                                                   Automated Messages for Project : ", project_name)
                                                                              visible OF main window := FALSE
                                                                                                                                                       start the module := TRUE
                                                                              visible OF message window := TRUE
                                                                              output OF message window: message display
      BEGIN
                                                                                                                                                      FND
 IF (result_in_major_modification OF
AB_general_rules = "Yes" OR
                                                                                                                                                   ATTRIBUTE include_initial_design SIMPLE
                                                                            END
                                                                          ATTRIBUTE general rule message SIMPLE WHEN CHANGED
                                                                                                                                                     WHEN CHANGED
 result_in_major_modification OF AB_general_rules = "No") THEN
                                                                                                                                                      BEGIN
                                                                            BEGIN
                                                                                                                                                       FORGET attackment OF valuebox 1
                                                                                                                                                       FORGET message status OF required mess
IF include_initial_design = TRUE THEN
include_initial_design_value := ** * **
         BEGIN
                                                                         IF ( CONF(cause of the effect OF discipline_message) < 0) THEN cause of the effect OF discipline_message :=
           EXEC i5odbc 1 SQL update [AB-general-
                  percent_change
:AB_general_rules percent_change,
result_in_major_modification =
:AB_general_rules.result_in_major_modification,
                                                                                                                                                         include_initial_design_value := " "
                                                                                                                                                      END
                                                                         message builder OF required message :=
CONCAT( "The attribute ", changed attribute OF
discipline_message, " of the building component ",
changed_class OF discipline_message, " (",
                                                                                                                                                   ATTRIBUTE include_initial_design_value STRING
 cause of the effect =: AB_general rules cause of the effect
                                                                                                                                                   INSTANCE border 1 ISA border
                                                                                                                                                    WITH style IS shad
                  where AB_general_rules_ID =
:AB_general_rules.AB_general_rules_ID
END SQL
                                                                         changed_instance_id OF discipline_message, "Y has been changed from the value (", old_value OF
                                                                                                                                                    WITH perspective IS in WITH border width := 5
          visible OF helping window := FALSE
FORGET output OF helping window
                                                                                                                                                    WITH background color := 192,192,192
                                                                          discipline_message, ") to the value (", new_value OF
                                                                         discipline_message, ").
Such a change may affect the design of the ".
                                                                                                                                                    WITH fill color := 0.0.128
                                                                                                                                                    WITH his color := 0,0,722
WITH highlight color := 255,255,255
WITH shadow color := 0,0,0
         END
                                                                         discipline to OF discipline message, "system.
The possible cause of the effect is ",
        ET SE
         BEGIN
                                                                                                                                                    WITH location := 15.15.280.225
          text OF message box 1 := "Last box value
                                                                          cause of the effect OF discipline message, "
 should be Yes or No"
                                                                                                                                                  INSTANCE border 2 ISA border
        put up OF message box 1 := TRUE END
                                                                         It is advisable to take some of the following steps:
                                                                                                                                                    WITH style IS edit control
                                                                             Redesign the affected building system.
                                                                                                                                                    WITH perspective IS in
      END
                                                                             Negotiate the change with design
                                                                                                                                                    WITH border width := 1
                                                                                                                     ing discipline.
                                                                            * Modefy the rule that caused this m
                                                                                                                                                    WITH background color := 192,192,192
WITH fill color := 192,192,192
! **** in case of a specific rule
                                                                         If you need more information, please push the HELP
                                                                                                                                                    WITH highlight color := 255,255,255
IF (general_specific OF discipline_message = "SPECIFIC") THEN
                                                                                                                                                    WITH shadow color := 128,128,128
                                                                                                                                                    WITH location := 390,20,610,140
      BEGIN
                                                                         If you want to see and edit the rule that caused this
       IF ( CONF(relation OF AB_specific_rules) >= 0)
                                                                         automated message
to be sent, please push the RULE botton")
                                                                                                                                                  INSTANCE border 3 ISA border
WITH style IS shadow
THEN
        BEGIN
                                                                                                                                                    WITH perspective IS in
relationship OF AB_specific_rules := TO
STRING(relation OF AB_specific_rules)
                                                                         ATTRIBUTE specific rule message SIMPLE WHEN CHANGED
                                                                                                                                                    WITH border width := 5
                                                                                                                                                    WITH background color := 192,192,192
        FND
                                                                            BEGIN
                                                                                                                                                    WITH fill color := 192,192,192
                                                                        message builder OF required message:

CONCAT( "The attribute", changed attribute OF discipline message, " of the building component ", changed class OF discipline message, "(, changed instance id OF discipline message, ") has been changed from the value (', old value OF discipline message, ") has been changed from the value (', old value OF discipline message, ").
                                                                                                                                                    WITH highlight color := 255,255,255
WITH shadow color := 128,128,128
       EXEC ISodbe 1 SQL update [AB-specific-rules]
                  relationship =
                                                                                                                                                    WITH location := 10.235.615.385
:AB_specific_rules.relationship
                  where AB_specific_rules_id =
                                                                                                                                                  INSTANCE border 4 ISA border
:AB_specific_rules_AB_specific_rules_id
END SQL
                                                                                                                                                    WITH style IS picture frame
                                                                                                                                                   WITH perspective IS in
WITH border width := 1
                                                                         discipline message, ") to the value (", new_value OF discipline message, ").
      END
                                                                         According to a previously define SPECIFIC rule, such
                                                                                                                                                    WITH background color := 192,192,192
! *** end of method
                                                                         a change may affect the design of the attribute
", affected_attribute OF discipline_message,"
                                                                                                                                                   WITH fill color := 192,192,192
   END
                                                                                                                                                    WITH highlight color := 255,255,255
ATTRIBUTE ignore rule SIMPLE
                                                                        building component ", affected class OF
discipline message, " (", affected instance id OF
discipline message, ")".
                                                                                                                                                    WITH shadow color := 128,128,128
  WHEN CHANGED
                                                                                                                                                    WITH location := 5.50.300.345
   REGIN
                                                                                                                                                  INSTANCE border 5 ISA border
  *** in case of a Genreal rule
                                                                         It is advisable to take some of the following steps:
                                                                                                                                                   WITH style IS edit control
                                                                           * Redesign the affected building component
IF (general_specific OF discipline_message ==
"GENERAL") THEN
                                                                                                                                                   WITH perspective IS in
WITH border width := 1

    Negotiate the change with designing discipline.

     BEGIN

    Modery the rule that caused this message.

                                                                                                                                                   WITH background color := 192,192,192
       EXEC ISodbc 1 SQL update [AB-general-rules]
                                                                                                                                                   WITH fill color := 192,192,192
                                                                         If you need more information, please push the HELP
                                                                                                                                                   WITH highlight color := 255,255,255
WITH shadow color := 128,128,128
                  rule ignored = :domain.YesS
                  where AB_general_rules_ID =
                                                                                                                                                    WITH location := 5,50,300,345
:AB_general_nules.AB_general_nules_ID
END SQL
                                                                         If you want to see and edit the rule that caused this
                                                                                                                                                  INSTANCE border 6 ISA border
                                                                           tomated message
       visible OF helping window := FALSE
FORGET output OF helping window
                                                                                                                                                   WITH style IS edit control
                                                                         to be sent, please push the RULE botton")
                                                                           END
                                                                                                                                                   WITH perspective IS in
                                                                        ATTRIBUTE go_shead SIMPLE WHEN CHANGED
      END
                                                                                                                                                    WITH border width := 1
                                                                                                                                                   WITH background color := 192,192,192
! *** in case of a Specific! rule
                                                                           BEGIN
                                                                                                                                                   WITH fill color := 192,192,192
IF (general_specific OF discipline_message = "SPECIFIC") THEN
                                                                                                                                                   WITH highlight color := 255,255,255
WITH shadow color := 128,128,128
                                                                              action OF ISodbc 2 GENERAL IS connect :=
                                                                        TRUE
     BEGIN
                                                                                                                                                   WITH location := 5.5.290.265
       EXEC ISodbc 1 SQL update [AB-specific-rules]
202
                                                                             EXEC ISodbe 2 GENERAL SQL SELECT
                                                                                                                                                  INSTANCE border 7 ISA border
                                                                        Project_name from Current_projects
END SQL INTO Current_projects
                  rule_ignored = :domain_Yes$
                                                                                                                                                   WITH style IS picture frame
where AB_specific_rules_id =:
AB_specific_rules.AB_specific_rules_id
END SQL
                                                                                                                                                   WITH perspective IS in
                                                                         (Project_name)
                                                                                                                                                    WITH border width := 1
                                                                                                                                                   WITH background color := 192,192,192
       visible OF specific rule window := FALSE
                                                                              visible OF main window := FALSE
                                                                                                                                                    WITH fill color := 192,192,192
                                                                                                                                                   WITH highlight color := 255,255,255
WITH shadow color := 128,128, 128
       FORGET output OF specific rule window
                                                                             visible OF Select a project window := TRUE
      END
                                                                             output OF Select a project window :=
                                                                         select_project_display
                                                                                                                                                    WITH location := 10,10,240,195
! **** end of method
                                                                           FND
                                                                         ATTRIBUTE project_name STRING
  END
                                                                                                                                                  INSTANCE border 8 ISA border
ATTRIBUTE start the module SIMPLE
                                                                           WHEN CHANGED
                                                                                                                                                    WITH style IS shadow
  WHEN CHANGED
                                                                                                                                                   WITH perspective IS in WITH border width := 6
                                                                           REGIN
                                                                             data source OF ISodbc 1 := project_name
   BEGIN
```

```
WITH background color: # 192,192,192
                                                                                                    WITH items [19]:= textbox $
                                                                                                                                                                                                   WITH fill color := 192,192,192
                                                                                                                                                                                                   WITH menus [1]:= UNDETERMINED
                                                                                                    WITH items [20] := textbox 9
WITH items [21] := UNDETERMINED
WITH items [22] := UNDETERMINED
    WITH fill color := 0.0,128
   WITH highlight color := 255,255,255
WITH shadow color := 0,0,0
                                                                                                                                                                                                 INSTANCE listbox 1 ISA listbox
                                                                                                  WITH items [22]: - UNDETERMINED
WITH items [23]: - UNDETERMINED
WITH items [24]: - UNDETERMINED
WITH items [25]: - pushbutton 10
WITH items [25]: - textbox 10
WITH items [26]: - textbox 13
WITH items [28]: - textbox 13
WITH items [29]: - UNDETERMINED
WITH items [10]: - UNDETERMINED
WITH items [10]: - UNDETERMINED
WITH items [32]: - UNDETERMINED
WITH items [32]: - UNDETERMINED
WITH items [33]: - checkbox 1
WITH fill color: - 192,192,192
WITH menus [1]: - UNDETERMINED
                                                                                                                                                                                                   WITH show current instance := TRUE
WITH source := Project_name OF Current_projects
   WITH location := 25 20 220 140
                                                                                                                                                                                                   WITH destination := project_name
WITH location := 30,150,210,180
 INSTANCE the domain ISA domain
   WITH Nos := "No"
   WITH Yes! := "Yes"
WITH Yes! := "Yes"
WITH include initial_design := FALSE -
WITH include_initial_design_value := " "
                                                                                                                                                                                                 INSTANCE promptbox 1 ISA promptbox
                                                                                                                                                                                                   WITH foot := "System"
                                                                                                                                                                                                   WITH justify IS center
 INSTANCE ISodbe 1 ISA ISodbe
                                                                                                                                                                                                   WITH frame := TRUE
   WITH data source := "build"
                                                                                                                                                                                                   WITH show current := TRUE
   WITH connection prompting := FALSE
                                                                                                                                                                                                   WITH attachment := percent_change OF
   WITH auto commit := TRUE
                                                                                                                                                                                                 AB_general_rules
   WITH append := FALSE
WITH default error handling := TRUE
                                                                                                                                                                                                  WITH location := 240,170,290,200
                                                                                                                                                                                                INSTANCE promptbox 2 ISA promptbox
WITH font := "System"
WITH justify IS left
                                                                                                 INSTANCE helping display ISA display WITH wait := FALSE
 INSTANCE ISodbe 2 GENERAL ISA ISodbe
   WITH data source := "general"
                                                                                                    WITH delay changes := FALSE
  WITH connection prompting := FALSE
WITH auto commit := TRUE
                                                                                                   WITH items [1]:= border 4
                                                                                                                                                                                                   WITH frame := TRUE
                                                                                                   WITH items [2] := textbox 11
WITH items [3] := textbox 12
WITH items [4] := UNDETERMINED
                                                                                                                                                                                                  WITH show current := TRUE
   WITH append := FALSE
WITH default error handling := TRUE
                                                                                                                                                                                                   WITH attachment := result_in_major_modification
                                                                                                                                                                                                OF AB_general_rules
WITH location := 240,225,290,255
                                                                                                   WITH items [5] := pushbutton 11
WITH items [6] := pushbutton 12
WITH fill color := 192,192,192
 INSTANCE message box 1 ISA message box
  WITH title: "Input Error"
WITH modal style IS app modal
WITH icon IS stop sign
                                                                                                                                                                                                INSTANCE promptbox 3 ISA promptbox
                                                                                                   WITH menus [1] := UNDETERMINED
                                                                                                                                                                                                  WITH font := "System"
WITH justify IS left
   WITH button IS ok
                                                                                                                                                                                                  WITH frame := TRUE
                                                                                                 INSTANCE general rule display ISA display
   WITH default button IS one
                                                                                                   WITH wait := FALSE
                                                                                                                                                                                                  WITH show current := TRUE
                                                                                                   WITH delay changes := FALSE
WITH items [1] := textbox 14
                                                                                                                                                                                                  WITH attachment := cause_of_the_effect OF
INSTANCE navigate 1 ISA navigate
WITH class attachment := discipline_message
                                                                                                                                                                                                 AB_general_nules
                                                                                                  WITH items [1] := textbox 14
WITH items [2] := border 5
WITH items [3] := UNDETERMINED
WITH items [4] := UNDETERMINED
WITH items [6] := UNDETERMINED
WITH items [6] := UNDETERMINED
WITH items [7] := UNDETERMINED
WITH items [8] := textbox 15
WITH items [9] := textbox 16
WITH items [10] := UNDETERMINED
WITH items [10] := UNDETERMINED
WITH items [11] := textbox 18
                                                                                                                                                                                                  WITH location := 15.285.290.335
                                                                                                                                                                                                INSTANCE OK botton ISA pushbutton
WITH label := "Go Ahead"
WITH attribute attachment := go_aheac
 INSTANCE the application ISA application
  WITH unknowns fail := TRUE
   WITH threshold := 50
  WITH title display := opening display WITH ignore breakpoints := FALSE
                                                                                                                                                                                                  WITH location := 90,230,195,255
   WITH reasoning on := FALSE
                                                                                                                                                                                                INSTANCE pushbutton 3 ISA pushbutton
  WITH numeric precision := 8
WITH demon strategy IS fire first
                                                                                                                                                                                                  WITH label := "Next"
WITH attribute attachment := action OF navigate 1 IS
                                                                                                 WITH items [10] := UNDETERMINED
WITH items [11] := textbox 18
WITH items [12] := promptbox 1
WITH items [13] := UNDETERMINED
WITH items [14] := textbox 19
WITH items [15] := promptbox 2
WITH items [16] := pushbutton 14
WITH items [17] := pushbutton 15
WITH items [18] := pushbutton 16
WITH items [19] := textbox 27
WITH items [20] := promptbox 3
INSTANCE checkbox 1 ISA checkbox
                                                                                                                                                                                                  WITH location := 400,70,460,100
  WITH label := "Include initial designs messages"
   WITH pen color := 0,0,255
                                                                                                                                                                                                INSTANCE pushbutton 4 ISA pushbutton
  WITH fill color := 192,192,192
WITH show current := TRUE
                                                                                                                                                                                                 WITH label := "Prev."
WITH attribute attachment := action OF navigate 1 IS
  WITH attachment := include_initial_design WITH location := 380,200,615,230
                                                                                                                                                                                                  WITH location := 470,70,530,100
                                                                                                  WITH items [20] := promptbox 3
WITH items [21] := valuebox 4
INSTANCE opening display ISA display
WITH wait := TRUE
WITH delay changes := FALSE
                                                                                                                                                                                                INSTANCE pushbutton 5 ISA pushbutton
                                                                                                  WITH items [22] := valuebox 5
WITH fill color := 192,192,192
                                                                                                                                                                                                 WITH label := "Last"
WITH attribute attachment := action OF navigate 1 IS
 WITH items [1] := border 6
WITH items [2] := border 1
WITH items [3] := etectboox 1
WITH items [3] := etectboox 1
WITH items [5] := OK botton
WITH items [5] := UNDETERMINED
WITH items [7] := UNDETERMINED
WITH items [7] := UNDETERMINED
WITH items [9] := UNDETERMINED
WITH items [1] := UNDETERMINED
  WITH items [1]:=border 6
                                                                                                   WITH menus [1] :- UNDETERMINED
                                                                                                                                                                                                  WITH location := 540,70,600,100
                                                                                                INSTANCE specific rule display ISA display
                                                                                                   WITH wait := FALSE
                                                                                                                                                                                                INSTANCE pushbutton 6 ISA pushbutton
                                                                                                  WITH delay changes := FALSE
WITH items [1] := textbox 17
                                                                                                                                                                                                 WITH label := "Read First Message"
WITH attribute attachment := action OF navigate 1 IS
                                                                                                   WITH items [2] := valuebox 6
                                                                                                  WITH items [3 ] := valuebox 7
WITH items [4 ] := valuebox 8
                                                                                                                                                                                                  WITH location := 400,30,600,62
                                                                                                   WITH items [5] := valuebox 9
                                                                                                                                                                                                INSTANCE message has been read botton ISA
                                                                                                  WITH items [6] := valuebox 10
WITH items [7] := valuebox 11
                                                                                                                                                                                               pushbutton
WITH label := "Message has been read"
INSTANCE message display ISA display WITH wait := FALSE
                                                                                                   WITH items [8 ] := valuebox 12
                                                                                                                                                                                                  WITH location := 15,390,305,415
                                                                                                  WITH items [9] := textbox 20
WITH items [10] := textbox 21
  WITH delay changes := FALSE
                                                                                                                                                                                                INSTANCE message has been acted upon botton ISA
 WITH items [1] := border 3
WITH items [2] := valuebox 1
WITH items [3] := radiobutton group 1
WITH items [4] := radiobutton group 2
WITH items [5] := textbox 4
WITH items [6] := valuebox 2
WITH items [7] := textbox 2
                                                                                                   WITH items [11] := textbox 22
                                                                                                  WITH items [12] := textbox 23
WITH items [13] := textbox 24
                                                                                                                                                                                                  WITH label := "Message has been read and acted
                                                                                                                                                                                                upon"
WITH location := 310,390,610,415
                                                                                                   WITH items [14] := textbox 25
                                                                                                  WITH items [15] := textbox 26
WITH items [16] := UNDETERMINED
                                                                                                                                                                                                INSTANCE pushbutton 9 ISA pushbutton
                                                                                                  WITH items [10]:—UNDETERMINED
WITH items [17]:—pushbutton 17
WITH items [18]:—UNDETERMINED
WITH items [19]:—pushbutton 19
WITH fill color:—192,192,192
WITH menus [1]:—UNDETERMINED
  WITH items [7] := textbox 5
WITH items [8] := valuebox 3
WITH items [9] := border 2
                                                                                                                                                                                                  WITH label := "EXIT"
                                                                                                                                                                                                  WITH attribute attachment := exit OF application
                                                                                                                                                                                                  WITH location := 300,20,370,65
  WITH items [10] := pushbutton 3
WITH items [11] := pushbutton 4
                                                                                                                                                                                                INSTANCE pushbutton 10 ISA pushbutton
  WITH items [12] := pushbutton 5
WITH items [13] := pushbutton 6
WITH items [14] := message has been read botton
                                                                                                                                                                                                  WITH label := "
                                                                                                INSTANCE select_project_display ISA display WITH wait := FALSE
                                                                                                                                                                                                  WITH attribute attachment := helping
                                                                                                                                                                                                  WITH location := 300,75,370,105
                                                                                                  WITH delay changes := FALSE
WITH delay changes := FALSE
WITH items [1] := border ?
WITH items [2] := border $
WITH items [3] := textbox 28
WITH items [4] := listbox 1
  WITH items [15]:= message has been acted upon
                                                                                                                                                                                                INSTANCE pushbutton 11 ISA pushbutton
  WITH items [16] := pushbutton 9
                                                                                                                                                                                                  WITH label := "OK"
  WITH items [17] := textbox 6
                                                                                                                                                                                                  WITH location := 30,350,135,380
  WITH items [18] := textbox 7
```

INSTANCE pushbutton 12 ISA pushbutton WITH label := "RULE" WITH attribute attachment := show rule WITH location := 165,350,270,380

INSTANCE pushbutton 13 ISA pushbutton WITH label := ** WITH attribute attachment := show rule WITH location := 300,110,370,140

INSTANCE pushbutton 14 ISA pushbutton WITH label := "Cancel" WITH location := 5,350,70,380

INSTANCE pushbutton 15 ISA pushbutton WITH label := "Update Rule" WITH stiribute strachment := update of rule WITH location := 75,350,185,380

INSTANCE pushbutton 16 ISA pushbutton WITH label:= "Ignore Rule" WITH attribute attachment:= ignore rule WITH location:= 190,350,298,380

INSTANCE pushbutton 17 ISA pushbutton WITH label := "Cancel" WITH location := 215,340,425,370

INSTANCE pushbutton 19 ISA pushbutton WITH label: = "Ignore Rule" WITH attribute attachment: = ignore rule WITH location: = 10,340,210,370

INSTANCE radiobution group I ISA radiobution

group
WITH pen color := 0,0,0
WITH fill color := 192,192,192
WITH frame := TRUE
WITH group label := "Discipline"
WITH show current := TRUE
WITH stachment := discipline name OF
required_message
WITH location := 5,15,140,230

INSTANCE radiobutton group 2 ISA radiobutton

group
WITH pen color := 0,0,0
WITH fill color := 192,192,192
WITH fines: = TRUE
WITH group label := "Message status"
WITH show current := TRUE
WITH state/ments := message status OF
required message
WITH location := 140,15,290,150

INSTANCE textbox 1 ISA textbox
WITH pen color: = 255,255,255
WITH fill color: = 0,0,128
WITH fill color: = 0,0,128
WITH fill color: = "Times New Roman"
WITH font syle IS bold, italic, underline CF FALSE, strikeout CF FALSE
WITH font size: = 20
WITH text: = "MODULE FOR READING
AUTOMATED MESSAGES"
WITH location: = 25,55,265,185

INSTANCE textbox 4 ISA textbox WITH pen color := 255,0,0 WITH fill color := 192,192,192 WITH justify IS left WITH font := "System" WITH text:= "# of Messages" WITH location := 390,150,510,180

INSTANCE textbox 5 ISA textbox WITH pen color := 255,0,0 WITH fill color := 192,192,192 WITH justify IS left WITH font := "5ystem" WITH text := "5 of Current Message" WITH location := 390,175,570,203

INSTANCE textbox 6 ISA textbox
WITH pen color := 0,0,0
WITH fill color := 192,192,192
WITH justify IS left
WITH font := "System"
WITH font style IS bold CF FALSE, italic CF
FALSE, underline CF FALSE, strikeout CF FALSE

WITH frame := TRUE WITH text := "Read and Acted Upon Message" WITH location := 145,150,370,172

INSTANCE textbox 7 ISA textbox
WITH pen color:= 255,0,0
WITH fill color:= 192,192,192
WITH justify IS left
WITH font:= "System"
WITH frame:= "RUE
WITH text:= "Read but Un Acted Upon Message"
WITH location:= 145,200,370,220

INSTANCE textbox 8 ISA textbox
WITH pen color: = 0,0,255
WITH fill color: = 192,192,192
WITH justify IS left
WITH font: = "System"
WITH font style IS bold CF FALSE, italic CF
FALSE underline CF FALSE, strikeout CF FALSE
WITH frame: = TRUE
WITH text: = "Un Read and Un Acted Upon
Message"
WITH location: = 145,175,370,195

INSTANCE textbox 9 ISA textbox WITH pen color := 0,0,128 WITH fill color := 192,192,192 WITH justify IS left WITH font := "System" WITH text := "Select Message" WITH location := 440,110,580,130

INSTANCE testbox 10 ISA testbox WITH pen color:= 255,0,0 WITH fill color:= 192,192,192 WITH justify IS left WITH foot:= "System" WITH test:= "HELP" WITH location:= 315,80,355,100

INSTANCE textbox 11 ISA textbox
WITH pen color:= 255,255,255
WITH fill color:= 128,0,0
WITH justify IS center
WITH font:= "Times New Roman"
WITH font myle IS bold, italic CF FALSE, underline
CF FALSE, strikeout CF FALSE
WITH font size:= 18
WITH frame:= TRUE
WITH text:= "HELP DISPLAY"
WITH location:= 5,5,300,45

INSTANCE textbox 12 ISA textbox
WITH pen color := 0,0,0
WITH fill color := 192,192,192
WITH justify IS left
WITH font := "System"
WITH text := "This automated message has been sent
by the system to inform you about a specific design
change.

Such a change has violated a GENERAL or SPECIFIC rule that is built in the system. The components of that rule can be seen if you click on the RULE botton.

If you feel that this rule does not reflect your building system design requirement, you may modify the rule." WITH location := 25,60,280,340

INSTANCE textbox 13 ISA textbox
WITH pen color:= 0,0,128
WITH fill color:= 192,192,192
WITH justify IS left
WITH font:= "System"
WITH font style IS italic
WITH text:= "RULE"
WITH location:= 310,115,360,135

INSTANCE textbox 14 ISA textbox

WITH pen color := 255,255,255
WITH fill color := 128,0,0
WITH justify IS center
WITH font := "Times New Roman"
WITH font style IS bold, italic CF FALSE, underline
CF FALSE, strikeout CF FALSE
WITH font size := 18
WITH frame := TRUE
WITH text := "GENERAL RULE"
WITH location = 5,5,300,45

INSTANCE textbox 15 ISA textbox
WITH pen color: = 0,0,0
WITH fill color: = 192,192,192
WITH justify IS left
WITH font: = "System"
WITH text: = "Changed Building Component"
WITH lostion: = 15,55,140,90

INSTANCE textbox 16 ISA textbox
WITH pen color := 0,0,0
WITH fill color := 192,192,192
WITH justify IS left
WITH fost := "System"
WITH fost := "Changed Component Attribute"
WITH location := 15,95,160,130

INSTANCE textbox 18 ISA textbox
WITH pen color:= 0,0,0
WITH fill color:= 192,192,192
WITH justify IS left
WITH font:= "System"
WITH text:= "Percent change in value that
requires design change from the
affected discipline: (i.e. 0.3)"
WITH location:= 15,140,230,195

INSTANCE textbox 19 ISA textbox
WITH pen color:= 0,0,0
WITH fill color:= 192,192,192
WITH justify IS left
WITH font:= "System"
WITH text:= "Will there be major
modifications with such a
design change (Yes/No)"
WITH location:= 15,205,205,270

INSTANCE textbox 17 ISA textbox
WITH pers color: = 255,255,255
WITH fill color: = 128,0,0
WITH justify IS center
WITH font: = "Times New Roman"
WITH font style IS bold, italic CF FALSE, underline
CF FALSE, strikeout CF FALSE
WITH font size: = 18
WITH frame: = TRUE
WITH text: = "Specific Rule"
WITH location: = 5,5,425,45

INSTANCE tentbox 20 ISA tentbox
WITH pen color: = 0,0,0
WITH fill color: = 192,192,192
WITH justify IS left
WITH foxt: = "System"
WITH text: = "Building Component:"
WITH text: = "Building Component:"

INSTANCE textbox 21 ISA textbox
WITH per color: = 0,0,0
WITH fill color: = 192,192,192
WITH justify IS left
WITH fost: = "System"
WITH text: = "Building Component ID:"
WITH location: = 10,85,170,110

INSTANCE tentbox 22 ISA tentbox WITH pen color: = 0,0,0 WITH fill color: = 192,192,192 WITH justify IS left WITH font: = "System" WITH tent: = "Component Attribute:" WITH location: = 10,115,170,140

INSTANCE textbox 23 ISA textbox
WITH pen color: = 0,0,0
WITH fill color: = 192,192,192
WITH justify IS left
WITH font: = "System"
WITH text: = "Related Building Component:"
WITH location: = 10,165,220,190

INSTANCE textbox 24 ISA textbox
WITH pen color: = 0,0,0
WITH fill color: = 192,192,192
WITH justify IS left
WITH foot: = "System"
WITH text: = "Related Component ID:"
WITH location: = 10,195,220,220

INSTANCE textbox 25 ISA textbox

WITH pen color := 0,0,255
WITH fill color := 192,192,192 WITH justify IS left WITH location := 5,8,634,460
WITH menus [1] := UNDETERMINED
WITH style IS moveable, sizeable, closeable CF WITH fort := "System"
WITH text := "Related Component Attribute:" WITH justify IS left WITH font := "System' WITH frame := TRUE WITH location := 10,226,235,250 FALSE WITH title := "Reading Autom WITH visible := FALSE WITH dipped := TRUE INSTANCE textbox 26 ISA textbox WITH attachment := building_element OF WITH pen color := 0,0,0 WITH fill color := 192,192,192 AB_specific_rules WITH location := 235,55,425,80 INSTANCE beloing window ISA wind WITH location := 149,43,463,459
WITH menus [1] := UNDETERMINED WITH justify IS left WITH fort := "System"
WITH text := "Current Relationship:" INSTANCE valuebox 7 ISA valuebox NSTANCE valuebox 7 ISA value WITH pen color := 0,0,255 WITH fill color := 192,192,192 WITH justify IS left WITH font := "System" WITH style IS moveable, sizeable, closeable WITH location := 10,255,155,275 WITH title := " WITH visible := FALSE INSTANCE textbox 27 ISA textbox WITH pen color := 0,0,0 WITH fill color := 192,192,192 WITH frame := TRUE INSTANCE specific rule window ISA window WITH location := 97,37,543,447 WITH menus [1] := UNDETERMINED WITH dipped := TRUE WITH justify IS left WITH attachment := building_element_id OF WITH font := "System"
WITH text := "Effect on the building system" WITH style IS moveable, sizeable, closeable WITH title := ** AB_specific_nules WITH location := 235.85.425.110 WITH location := 15,265,280,285 WITH visible := FALSE INSTANCE valuebox & ISA valuebox INSTANCE textbox 28 ISA textbox WITH pen color := 0,0,255 WITH fill color := 192,192,192 INSTANCE Select a project window ISA window WITH location := 151,149,407,383 WITH pen color := 255,255,255 WITH fill color := 0,0,128 WITH justify IS left WITH menus [1] := UNDETERMINED WITH justify IS center
WITH foot := "Times New Roman"
WITH foot style IS bold, italic CF FALSE, underline WITH font := "System WITH frame := TRUE WITH style IS moveable, sizeable, closeable WITH title := "(Untitled)" WITH clipped := TRUE
WITH strachment := attribute_of_building_element WITH visible = FALSE CF FALSE, strikeout CF FALSE WITH font size := 18
WITH text := "Select a Project from the List" OF AB_specific_rules ! DEMON GROUP: discipline name OF WITH location := 235,115,425,140 required_message WITH location := 40,45,205,110 INSTANCE valuebox ! ISA valuebox INSTANCE valuebox ! ISA valuebox
WITH pen color := 0,0,0
WITH fill color := 192,192,192
WITH justify IS left
WITH font := "Times New Roman"
WITH font style IS bold CF FALSE, italic CF
FALSE, underline CF FALSE, strikeout CF FALSE WITH pen color := 0,0,255 WITH fill color := 192,192,192 IF discipline name OF required_message IS WITH justify IS left THEN choosen_discipline OF required_message := WITH font := "System"
WITH frame := TRUE AND FORGET attachment OF valuebox ! WITH clipped := TRUE AND FORGET message status OF required_message WITH attachment := related class OF WITH font size := 12 AB_specific_rules WITH location := 235,165,425,190 **DEMON 2** IF discipline name OF required_message IS HVAC THEN choosen_discipline OF required_message :* WITH frame := FALSE WITH clipped := TRUE WITH scroll := TRUE INSTANCE valuebox 10 ISA valuebox WITH pen color := 0,0,255 WITH fill color := 192,192,192 WITH horizontal scroil := TRUE AND FORGET attachment OF valuebox I WITH location = 20 245 600 170 AND FORGET message status OF required_message WITH justify IS left INSTANCE valuebox 2 ISA WITH font := "System" WITH pen color := 0,0,255 WITH fill color := 192,192,192 WITH frame := TRUE IF discipline name OF required message IS WITH clipped := TRUE
WITH attachment := related_instance_id OF WITH justify IS center WITH foat := "System" THEN choosen_discipline OF required_message := AB_specific_rules WITH location := 235,195,425,220 *ՈԽ* WITH frame := TRUE AND FORGET attachment OF valuebox I WITH clipped := TRUE AND FORGET message status OF required_message WITH attachment := number instances OF navigate 1 INSTANCE valuebox 11 ISA valuebox WITH pen color := 0,0,255 WITH fill color := 192,192,192 WITH location := 565,150,595,170 IF discipline name OF required_message IS Envelope THEN choosen_discipline OF required_message := INSTANCE valuebox 3 ISA valuebox WITH justify IS left WITH font := "System" WITH frame := TRUE WITH pen color := 0,0,128 WITH fill color := 192,192,192
WITH justify IS center
WITH font := "System"
WITH frame := TRUE AND FORGET attachment OF valuebox i WITH clipped := TRUE WITH attachment := related_attribute OF AND FORGET message status OF required_message AB_specific_rules WITH clipped := TRUE WITH location := 235,225,425,250 IF discipline name OF required message IS Structure THEN choosen discipline OF required message := WITH attachment := instance number OF navigate 1
WITH location := 565,175,595,195 INSTANCE valuebox 12 ISA va AND FORGET attachment OF valuebox I WITH pen color := 0,0,255 WITH fill color := 192,192,192 INSTANCE valuebox 4 ISA valu AND FORGET message status OF required_message WITH justify IS left WITH foot := "System" WITH frame := TRUE WITH pen color := 0,0,255 WITH fill color := 192,192,192 DEMON 13 WITH justify IS center IF discipline name OF required_message IS Interior WITH font := "System" WITH frame := TRUE WITH dipped := TRUE THEN choosen_discipline OF required_message := WITH attachment := relationship OF WITH dipped := TRUE AND FORGET attachment OF valuebox 1 AB_specific_rules WITH attachment := building_element OF WITH location := 10,275,425,330 AND FORGET message status OF required_message AB_general_rules I DEMON GROUP: message status OF WITH location := 100,75,290,95 INSTANCE main window ISA window WITH location := 150,107,454,406
WITH menus [1] := UNDETERMINED
WITH style IS moveable, sizeable, closeable required_message INSTANCE valuebox 5 ISA valuebox WITH pen color:= 0,0,255
WITH fill color:= 192,192,192
WITH justify IS center
WITH font:= "System"
WITH frame:= TRUE **DEMON 5** WITH title := "Automated Messages Module" WITH visible := TRUE. IF message status OF required_message IS All Messages AND choosen_discipline OF required_message < "UNDETERMINED"
THEN EXEC ISodbe I SOL select * from INSTANCE expand window ISA window WITH dipped := TRUE WITH location := 13.8.633.458 WITH attachment := attribute_of_building_element WITH menus [1]: - UNDETERMINED message_elements OF AB_general_rules WITH location := 100,115,290,135 WITH style IS moveable, sizeable, closeable WITH title: "(Untitled)" where discipline_to = required_message.choosen_discipline AND rule_ignored = :domain.No\$ WITH visible := FALSE Ahmed Mokhtar, Ph.D Thesis230

INSTANCE valuebox 6 ISA valuebox

WITH pen color := 0,0,0

WITH fill color := 192,192,192

INSTANCE message window ISA window

AND old_value 🗢 :domain include initial design_value END SQL INTO discipline message AND FORGET attachment OF valuebox 1

IF message status OF required message IS UnRead Yet AND choosen discipline OF required message
"UNDETERMINED" THEN EXEC ISodoc I SQL select * from message_elements where discipline_to = where discipline to "
required_message.choosen_discipline
AND been_read = :No\$
AND rule_ignored = :domain.No\$
END SQL INTO discipline_message
AND FORGET_attachment OF valuebox 1

DEMON 7

IF message status OF required_message IS UnActed Upon
AND choosen_discipline OF required_message
"UNDETERMINED" THEN EXEC ISodbc i SQL select * from

message elements where discipline_to = :required_message.choosen_discipline AND been_read = :YesS AND been_exted_upon =: NoS
AND nule_ignored =: domain.NoS
END SQL_INTO discipline_message
AND FORGET attachment OF valuebox 1

! DEMON GROUP: selected OF pushbutton

DEMON 17 IF selected OF message has been read botton
THEN EXEC 15odbc 1 SQL update message_elements been_read = :Yes\$

where id = :discipline_message.id END SOL

DEMON 18

IF selected OF message has been acted upon botton THEN EXEC 15odbc 1 SQL update message_elements been_acted_upon = :Yes\$, been_read = :Yes\$

where id = :discipline_message.id END SQL

DEMON &

IF selected OF pushbutton 9
ThEN action OF ISodbe I IS disconnect

DEMON 9

F selected OF pushbutton 11
THEN visible OF helping window := FALSE
AND FORGET output OF helping window

DEMON 11
If selected OF pushbutton I4
THEN visible OF helping window := FALSE
AND FORGET output OF helping window

DEMON 12 IF selected OF pushbutton 17 THEN visible OF specific rule window := FALSE AND FORGET output OF specific rule window

C.4. TRACKING MODULE

```
SVERSIONIS
                                                                                    attribute OF A_monitored_object_input := **
                                                                                                                                                      INIT 255 255 255
 SLOCATIONS ARE PIXELS
                                                                                   FORGET source OF listbox I
                                                                                                                                                     WITH shadow color COLOR
                                                                                                                                                      INIT 128,128,128
                                                                                   FORGET source OF listbox 2
                                                                                   Change_ID OF A_monitored_object_input := "*
CLASS A_attributes_of_monitored_object
WITH name STRING
                                                                                                                                                    CLASS combobox INHERITS add on, display item
                                                                                                                                                    WITH attachment ATTRIBUTE REFERENCE
WITH selected SIMPLE
CLASS A_change_monitoring_results
                                                                               IF select_discipline OF A_monitored_object_input
                                                                         IS Illumination THEN
                                                                                                                                                     WITH enabled SIMPLE
  WITH Change_ID STRING
                                                                                 BEGIN
                                                                                                                                                      INIT TRUE
                                                                                                                                                    WITH sort SIMPLE
INIT FALSE
                                                                                  discipline OF A_monitored_object_input :=
  WITH date time STRING
WITH discipline STRING
                                                                          "illu"
                                                                                                                                                    WITH update with current instance SIMPLE INIT TRUE
  WITH object STRING
WITH object_ID STRING
WITH attribute STRING
                                                                                     et all the other query data
                                                                                  object OF A monitored object input := "
object_ID OF A monitored object_input := "
attribute OF A monitored object_input := "
  WITH old_value STRING
                                                                                                                                                   CLASS Current_projects
                                                                                                                                                    WITH Project_name STRING
  WITH new value STRING
WITH preceeding change ID STRING
                                                                                  FORGET source OF listbox 1
FORGET source OF listbox 2
                                                                                   Change_ID OF A_monitored_object_input := ""
                                                                                                                                                   CLASS discipline_m
                                                                                                                                                    CLASS discipline message
WITH changed_stribute STRING
WITH changed_class STRING
WITH changed_instance_id STRING
WITH old_value STRING
                                                                                 END
CLASS A_monitored_object_input
WITH object STRING
WITH object_ID STRING
                                                                         IF select_discipline OF A_monitored_object_input IS Interior THEN
  WITH attribute STRING
WITH start_time STRING
                                                                                                                                                    WITH new_value STRING
  WITH end_time STRING
                                                                                  discipline OF A_monitored_object_input :=
                                                                                                                                                    WITH affected attribute STRING
                                                                                                                                                    WITH affected class STRING
  WITH select_discipline COMPOUND
                                                                               -reast all the other query data
object OF A_monitored_object_input := **
object_ID OF A_monitored_object_input := **
attribute OF A_monitored_object_input := **
                                                                                                                                                    WITH affected_class 5 TRING
WITH affected_instance_id STRING
WITH relationship STRING
WITH writing_date TIME
WITH reading_date TIME
   Architecture,
   Structure,
   Egyelope
                                                                                                                                                    WITH been read STRING
WITH been acted upon STRING
WITH Id NUMERIC
   Hirmination
                                                                                  FORGET source OF listbox 1
                                                                                  FORGET source OF listbox 2
   Interior
   WHEN CHANGED
                                                                                  Change ID OF A monitored object_input := ""
                                                                                                                                                    WITH discipline to STRING
WITH discipline from STRING
    REGIN
                                                                                                                                                    WITH asseptime_trost STRING
WITH sense because change ID NUMERIC
WITH general_specific STRING
WITH vide_writing_date STRING
WITH videated_rule_id NUMERIC
WITH cause_of_the_effect STRING
                                                                         IF select_discipline OF A_monitored_object_input = FALSE THEN
      IF select_discipline OF A_monitored_object_input
IS Architecture THEN
       BEGIN
                                                                                BEGIN
         discipline OF A_monitored_object_input :=
                                                                                  discipline OF A_monitored_object_input := "
"arch"
                                                                                END
      rener all the other query data
object OF A_monitored_object_input := ""
object_ID OF A_monitored_object_input := ""
attribute OF A_monitored_object_input := ""
                                                                         |**** end of method
                                                                                                                                                   CLASS ISodbe INHERITS add on
                                                                                                                                                    WITH data source STRING
                                                                             FND
                                                                           WITH discipline STRING
                                                                                                                                                    WITH userid STRING
                                                                          WITH selection_statement STRING
WITH Change_ID STRING
WHEN CHANGED
                                                                                                                                                    WITH pessword STRING
         FORGET source OF listbox I
                                                                                                                                                    WITH connection string STRING
         FORGET source OF listbox 2
         Change_ID OF A_monitored_object_input := "
                                                                                                                                                    WITH connection prompting SIMPLE INIT FALSE
                                                                             BEGIN
                                                                             **** reset all the other query data

IF Change_ID OF A_monitored_object_input 	
                                                                                                                                                    WITH auto commit SIMPLE
     IF select_discipline OF A_monitored_object_input
                                                                                                                                                     INIT TRUE
                                                                                                                                                    WITH action COMPOUND
IS Structure THEN
                                                                         "THEN
                                                                                BEGIN
       BEGIN
                                                                                                                                                     connect.
         discipline OF A_monitored_object_input :=
                                                                                  discipline OF A monitored_object_input :=
                                                                                 object OF A_monitored_object_input := ""
object_ID OF A_monitored_object_input := ""
                                                                                                                                                    WITH records NUMERIC WITH transaction COMPOUND
       reset all the other query data
        object OF A_monitored_object_input := ""
object_ID OF A_monitored_object_input := ""
attribute OF A_monitored_object_input := ""
FORGET source OF listbox 1
                                                                                  attribute OF A_monitored_object_input := **
                                                                                  FORGET source OF listbox I
                                                                                                                                                     milback
                                                                                                                                                    WITH append SIMPLE
INIT FALSE
                                                                                  FORGET source OF listbox 2
                                                                         select_discipline OF A_monitored_object_input
:= UNDETERMINED
                                                                                                                                                    WITH status STRING
        FORGET source OF listbox 2
        Change_ID OF A_monitored_object_input := "
                                                                                                                                                    WITH show error SIMPLE
                                                                                END
       END
                                                                                                                                                    WITH default error handling SIMPLE
                                                                                                                                                     INIT TRUE
                                                                             FND
     IF select_discipline OF A_monitored_object_input
                                                                                                                                                    WITH error message STRING
IS Envelope THEN
                                                                        CLASS A_object_ID_of_monitored_object
WITH ID STRING
                                                                                                                                                    WITH native error NUMERIC
                                                                                                                                                    WITH trace file STRING
       BEGIN
        discipline OF A_monitored_object_input :=
                                                                                                                                                  CLASS message box INHERITS add on WITH title STRING
                                                                          LASS border INHERITS add on, display item 
WITH style COMPOUND
      - reset all the other query data
object OF A_monitored_object_input := ""
object_ID OF A_monitored_object_input := ""
                                                                                                                                                    WITH text STRING
                                                                            picture frame.
                                                                            edit control,
                                                                                                                                                    WITH variable text STRING
        attribute OF A monitored object input := ""
FORGET source OF listbox 1
                                                                                                                                                    WITH modal style COMPOUND
                                                                            group,
shadow
                                                                                                                                                     app modal,
                                                                                                                                                     sys modal.
         FORGET source OF listbox 2
                                                                          INIT picture frame
WITH perspective COMPOUND
                                                                                                                                                     not modal
        Change_ID OF A_monitored_object_input := **
                                                                                                                                                    INIT app modal
WITH icon COMPOUND
     IF select_discipline OF A_monitored_object_input
                                                                                                                                                     exclamation
                                                                            INTT in
IS HVAC THEN
                                                                           WITH border width NUMERIC
                                                                                                                                                      information,
       BEGIN
                                                                            INIT I
                                                                                                                                                      question.
        discipline OF A_monitored_object_input :=
                                                                           WITH background color COLOR.
                                                                                                                                                     stop sign
                                                                            INIT 192,192,192
                                                                                                                                                     INIT exclamation
                                                                           WITH 6II color COLOR
                                                                                                                                                    WITH button COMPOUND
     - reset all the other query data
        object OF A_monitored_object_input := ""
object_ID OF A_monitored_object_input := ""
                                                                            INIT 192,192,192
                                                                                                                                                     abort retry ignore,
                                                                           WITH highlight color COLOR
```

ok cancel,	WITH designers of related class attributes STRING	selection_statement OF
retry cancel, yes no,	ARRAY SIZE I	A_monitored_object_input := CONCAT(selection_statement OF A_monitored_object_input, *
yes no cancel	CLASS required message	AND attribute = : A monitored object input attribute")
INIT ok WITH default button COMPOUND	WITH discipline name COMPOUND Architecture	END 1
one,	HVAC,	! **** To select based on time period
two, three	Illumination, Envelope,	! Case of start time alone IF start_time OF A_monitored_object_input
INIT one	Structure,	◇ ™ THEN
WITH button selected COMPOUND abort.	Interior WITH message status COMPOUND	BEGIN IF LENGTH(selection_statement OF
cancel,	All Messages,	A monitored object input) > 47 THEN
igaore, no,	UnRead Yet, UnActed Upon	selection_statement OF A_monitored_object_input := CONCAT(
ok,	WITH choosen_discipline STRING	selection_statement OF A_monitored_object_input, *
retry, yes,	WITH message builder STRING	AND") selection statement OF
cancell	CLASS temp_current_menu_items	A_monitored_object_input := CONCAT(
WITH put up SIMPLE	WITH label STRING	selection_statement OF A_monitored_object_input, * date_time >= :A_monitored_object_input.start_time*)
CLASS mavigate INHERITS add on	ATTRIBUTE IN NUMERIC	END
WITH class attachment CLASS REFERENCE WITH action COMPOUND	ATTRIBUTE number of table columns NUMERIC ATTRIBUTE show results SIMPLE	! Case of end time with or without start time If end time OF A_monitored_object_input <>
first,	WHEN CHANGED	"THEN
previous, next,	BEGIN	BEGIN IF LENGTH(selection statement OF
last	! **** to test that there are input data	A_monitored_object_input) > 47 THEN
WHEN CHANGED	IF ((object OF A_monitored_object_input <> ***)	selection_statement OF
BEGIN IF instance number OF navigate 1 O 0 THEN	OR (discipline OF A_monitored_object_input <> "") OR (start_time OF A_monitored_object_input <> "")	A_monitored_object_input := CONCAT(selection_statement OF A_monitored_object_input, *
BEGIN	OR (end_time OF A_monitored_object_input <> "")	AND")
! **** attach the discipline to the value box	OR (Change_ID OF A_monitored_object_input <> ***)) THEN	selection_statement OF A monitored object input := CONCAT(
attachment OF valuebox 16 := discipline_to OF	BEGIN	selection_statement OF A monitored object_input, *
discipline_message	f ***** to test if end time is bigger than start time if	date_time <= :A_monitored_object_input_end_time*) END
I **** Show the message on the screen	there is an end time	l t
IF been_read OF discipline_message = "Yes" AND been_acted upon OF discipline message = "No"	IF end_time OF A_monitored_object_input <> ** AND end_time OF A_monitored_object_input <	I ****** Running the select statement EXEC ISodbe 1 SQL WITH
THEN	start_time OF A_monitored_object_input THEN	selection_statement OF A_monitored_object_input
pen color OF valuebox 15 := 255,0,0 IF been_read OF discipline_message = "No"	BEGIN put up OF message box 2 := TRUE.	END SQL INTO A_change_monitoring_results
AND been_acted_upon OF discipline_message = "No"	END	i
THEN pen color OF valuebox 15 := 0,0,255	ELSE BEGIN	output OF expand window := Results visible OF expand window := TRUE
IF been_read OF discipline_message = "Yes"	1	visible OF Query window := FALSE
AND been_acted_upon OF discipline_message = "Yes" THEN	!**** Start of SELECT statement selection statement OF	visible OF main window := FALSE
pen color OF valuebox 15 := 0,0,0	A_monitored_object_input := "SELECT * FROM [A-	(***** end of second IF (check start abd end time)
attachment OF valuebox 15 := message builder OF required message	charge-monitoring-data] WHERE *	END !**** end of IF (check that there are inputs
i Ot. tedemen merrida	1 **** To select based on the discipline	END END
IF general_specific OF discipline_message = "GENERAL" THEN	IF discipline OF A_monitored_object_input THEN	ELSE BEGIN
general rule := TRUE	BEGIN	put up OF message box 1 := TRUE
! IF general specific OF discipline message =	selection_statement OF A_monitored_object_input := CONCAT(END 1 ***** end of method
"SPECIFIC" THEN	selection_statement OF A_monitored_object_input,	END
specific rule := TRUE	"discipline = :A_monitored_object_input.discipline")	ATTRIBUTE numeric_Change_ID NUMERIC
. END	END !	ATTRIBUTE return to query screen SIMPLE WHEN CHANGED
END	lesses To select based on the ID	BEGIN
WITH instance number NUMERIC WITH number instances NUMERIC	IF Change_ID OF A_monitored_object_input ◇ *** THEN	visible OF Query window := TRUE visible OF expand window := FALSE
CLASS relations building	BEGIN	END ATTRIBUTE return to results screen SIMPLE
WITH current class STRING	numeric_Change_ID := TO NUMERIC(Change_ID OF A_monitored_object_input)	WHEN CHANGED
WITH current instance id STRING WITH current attribute name STRING	selection_statement OF	BEGIN
WITH current display INSTANCE REFERENCE	A_monitored_object_input := CONCAT(selection_statement OF A_monitored_object_input,	FORGET attachment OF valuebox 16 FORGET discipling_message
WITH related class STRING WITH related attribute name STRING	"Change_ID = :numeric_Change_ID ")	FORGET required_message
WITH related instance id STRING	END	output OF expand window := Results visible OF expand window := TRUE
WITH Relationship types COMPOUND	100000 To select based on object data	visible OF messages window := FALSE
Equal to, More than,	IF object OF A_monitored_object_input <> ** THEN	END ATTRIBUTE go to message screen SIMPLE
Less than,	BEGIN	WHEN CHANGED
Exist with, Between values	selection_statement OF A_monitored_object_input := CONCAT(BEGIN output OF messages window := messages
WITH Chosen Relationship type STRING	selection_statement OF A_monitored_object_input,	visible OF messages window := TRUE
WITH columns headings STRING ARRAY SIZE I	"object =: A_monitored_object_input.object") IF object_ID OF A_monitored_object_input	visible OF expand window := FALSE END
WITH all id of related class STRING		ATTRIBUTE sample time STRING
ARRAY SIZE I WITH current class id attribute name STRING	selection_statement OF A_monitored_object_input := CONCAT(ATTRIBUTE complete as monitored object SIMPLE WHEN CHANGED
WITH related class id attribute name STRING	selection_statement OF A_monitored_object_input, "	BEGIN
WITH current attribute designer STRING WITH old value STRING	AND object_ID =	! !***** reset all the other query data
WITH new value STRING	:A_monitored_object_input.object_ID") IF attribute OF A_monitored_object_input	Change_ID OF A_monitored_object_input := ""
WITH related attribute designer STRING	O"THEN	discipline OF A monitored object input := **

```
WHEN CHANGED
     select discipline OF A monitored object input :=
UNDETERMINED
                                                                            IF ( CONF(cause_of_the_effect OF
                                                                                                                                                 BEGIN
                                                                                                                                                   data source OF ISodbc 1 := project_name
    object_ID OF A_monitored_object_input := **
attribute OF A_monitored_object_input := **
                                                                       discipline_message) < 0) THEN
cause_of_the_effect OF discipline_message :=
                                                                                                                                                   title OF Query window := CONCAT( *Change
                                                                                                                                              Monitoring for Project : ", project_name)
start the module := TRUE
                                                                       *Undefined
  **** get the ID of the object from the database
    general_statement_of_selection := CONCAT(
lect ", object OF A_monitored_object_input, "_id
                                                                      message builder OF required_message :=
CONCAT( "The attribute ", changed_attribute OF
discipline_message, " of the building component ",
changed_class OF discipline_message, " (",
                                                                                                                                              INSTANCE A_monitored_object_input 1 ISA
from ", object OF A monitored_object_input)
EXEC | Sodbe | 1 SQL WITH
                                                                                                                                               A_monitored_object_input
     ral_statement_of_selection END SQL INTO
                                                                       changed_instance_id OF discipline_message, ")
                                                                       has been changed from the value (*, old_value OF discipline_message, *) to the value (*, new_value OF
                                                                                                                                              INSTANCE border 1 ISA border
 A_object_ID_of_monitored_object (ID)
                                                                                                                                                WITH style IS group
                                                                      discipline message, ").
Such a change may affect the design of the ",
discipline to OF discipline message, " system.
The possible cause of the effect is ".
                                                                                                                                                WITH perspective IS in WITH border width := 1
    source OF listbox 1 := ID OF
A_object_ID_of_monitored_object
                                                                                                                                                WITH background color := 192,192,192
          et the attributes of the object from the database
                                                                                                                                                WITH fill color := 192,192,192
                                                                       cause of the effect OF discipline message,
                                                                                                                                                WITH highlight color := 255,255,255
WITH shadow color := 128,128,128
    EXEC ISodbc 1 SQL select
 attribute_of_building_element from [AB-attribute-
designer] where building_element =:
A_monitored_object_input.object
END SQL INTO
                                                                                                                                                WITH location := 10.60.145.155
                                                                       It is advisable to take some of the following steps:
                                                                          Redesign the affected building system
                                                                           Negotiate the change with designing discipline.
                                                                                                                                              INSTANCE border 2 ISA border
A_attributes_of_monitored_object (name)
                                                                         * Modery the rule that caused this message.
                                                                                                                                                WITH style IS group
                                                                                                                                                WITH perspective IS in
                                                                       If you need more information, please push the HELP
                                                                                                                                                WITH border width := 1
    source OF listbox 2 := name OF
                                                                                                                                                WITH background color := 192,192,192
   _attributes_of_monitored_object
                                                                                                                                                WITH fill color := 192,192,192
                                                                                                                                               WITH highlight color := 255,255,255
WITH shadow color := 128,128,128
!*** end of method
                                                                       If you want to see and edit the rule that caused this
                                                                          tomated message
  END
ATTRIBUTE general_statement_of_selection STRING
ATTRIBUTE start the module SIMPLE
                                                                       to be sent, please push the RULE botton. ")
                                                                                                                                                WITH location := 10,165,320,385
                                                                       ATTRIBUTE specific rule SIMPLE WHEN CHANGED
                                                                                                                                              INSTANCE border 3 ISA border
  WHEN CHANGED
                                                                                                                                                WITH style IS group
   BEGIN
                                                                                                                                                WITH perspective IS in WITH border width := 1
                                                                      message builder OF required message: "
CONCAT( "The attribute ", changed attribute OF discipline message, " of the building component ", changed class OF discipline message, ", changed instance id OF discipline message, ") has been changed from the value (", old value OF discipline message,") but be solved (" old value OF discipline message,") but be solved (" out when OF discipline message,")
    action OF ISodbc 1 IS connect := TRUE
                                                                                                                                                WITH background color := 192,192,192
                                                                                                                                                WITH fill color := 192,192,192
WITH highlight color := 255,255,255
WITH shadow color := 128,128,128
  **** Load the menus in the screens
    EXEC |Sodbc | SOL SELECT
DISTINCT(main_menu_label) from [AB-menu_item]
                                                                                                                                                WITH location := 330 245 550 385
                                                                      discipline message, ") to the value (", new_value OF discipline message, ").

According to a previously define SPECIFIC rule, such
        END SQL INTO menu (label)
                                                                                                                                              INSTANCE border 4 ISA border
    EXEC ISodbc I SQL SELECT label from [AB-
                                                                                                                                                WITH style IS edit control
                                                                                                                                               WITH perspective IS in
                                                                       a change may affect the design of the attribute
", affected attribute OF discipline message, " of the
                                                                                                                                                WITH border width := 1
        END SOL INTO menu item.
                                                                      building component ", affected class OF
discipline message, " (", affected instance id OF
discipline message, "y.
                                                                                                                                                WITH beckground color := 192,192,192
                                                                                                                                               WITH fill color := 192,192,192
    IF (number instances OF navigate menu > 0) THEN
                                                                                                                                                WITH his color := 192,192,192
WITH highlight color := 255,255,255
WITH shadow color := 128,128,128
     BEGIN
                                                                                                                                                WITH location := 5,50,555,175
       RESIZE menus OF Query window := number
                                                                       It is advisable to take some of the following steps:
                                                                          Redesign the affected building system.
instances OF navigate menu

    Negotiate the change with designing discipline.

                                                                                                                                              INSTANCE border 5 ISA border
                                                                                                                                               WITH style IS picture frame

    Modefy the rule that caused this message.

                                                                                                                                                WITH perspective IS in
       FOR (no := 1 TO number instances OF
                                                                       If you need more information, please push the HELP
                                                                                                                                                WITH border width := 6
navigate menu)
                                                                                                                                                WITH background color := 192,192,192
        BEGIN
                                                                                                                                                WITH fill color := 192,192,192
          instance number OF navigate menu : nn
                                                                                                                                               WITH highlight color := 255,255,255
WITH shadow color := 128,128,128
                                                                       If you want to see and edit the rule that caused this
  **** load the menu items in every n
                                                                      automated message
to be sent, please push the RULE botton.")
         EXEC 15odbc 1 SQL SELECT label from
                                                                                                                                                WITH location := 5,180,555,355
                                                                         END
[AB-menu_item] WHERE (main_menu_label) =
                                                                       ATTRIBUTE effect of mensu item SIMPLE
                                                                                                                                              INSTANCE border 6 ISA border
 menu lahel
                                                                        WHEN CHANGED
                                                                                                                                                WITH style IS shadow
                             END SQL INTO
                                                                                                                                                WITH perspective IS in WITH border width := 6
                                                                         BEGIN
temp_current_menu_items (label)
         RESIZE items OF menu := number insuances
                                                                                                                                                WITH background color := 192,192,192
OF navigate temp_current_menu_items
FOR (%nnn := 1 TO number instances OF
                                                                           object OF A_monitored_object_input := label OF
                                                                                                                                                WITH fill color := 0.0.128
                                                                                                                                                WITH highlight color := 255,255,255
                                                                           complete as monitored object := TRUE
navigate temp_current_menu_items)
BEGIN
                                                                                                                                                WITH shadow color := 0,0,0
                                                                                                                                                WITH location := 15.15.190.150
                                                                        *** end of method
            instance number OF navigate
                                                                         END
temp_current_menu_items := %nnn
                                                                       ATTRIBUTE %nnn NUMERIC
                                                                                                                                              INSTANCE border 7 ISA border
            FIND menu item
              WHERE label OF menu item = label OF
                                                                       ATTRIBUTE go_ahead SIMPLE WHEN CHANGED
                                                                                                                                                WITH style IS picture frame
                                                                                                                                                WITH perspective IS in
temp_current_menu_items
             FIND END
                                                                                                                                                WITH border width := 1
                                                                         BEGIN
                                                                                                                                               WITH background color := 192,192,192
            items[ %nnn] OF menu := menu item
                                                                                                                                                WITH fill color := 192,192,192
                                                                           action OF ISodbc 2 GENERAL IS connect :=
           END
                                                                                                                                                WITH highlight color := 255,255,255
WITH shadow color := 128,128,128
                                                                       TRUE
          menus(nn) OF Query window := menu
                                                                                                                                                WITH location := 340,65,545,165
                                                                           EXEC ISodbe 2 GENERAL SQL SELECT
        END
                                                                      Project_name from Current_projects
END SQL INTO Current_projects
                                                                                                                                              INSTANCE border & ISA border
     END
                                                                                                                                                WITH style IS shadow
                                                                       (Project name)
                                                                                                                                               WITH perspective IS in WITH border width := 4
     visible OF Query window := TRUE
                                                                            visible OF main window := FALSE
     visible OF main window := FALSE
                                                                                                                                                WITH background color := 192,192,192
                                                                            visible OF Select a project window := TRUE
 end of method
                                                                            output OF Select a project window :=
                                                                                                                                                WITH fill color := 192,192,192
                                                                                                                                               WITH highlight color := 255,255,255
WITH shadow color := 128,128,128
  EMD
                                                                       select_project_display
ATTRIBUTE general rule SIMPLE
  WHEN CHANGED
                                                                                                                                                WITH location := 10,220,545,372
                                                                       ATTRIBUTE project_name STRING
   BEGIN
```

INSTANCE border 9 ISA border
WITH style IS edit control
WITH perspective IS in
WITH border width := 1
WITH border width := 1
WITH blockground color := 192,192,192
WITH fill color := 192,192,192
WITH shadow color := 255,255,255
WITH shadow color := 128,128,128
WITH location := 5,5,200,200

INSTANCE border 10 ISA border
WITH style IS picture frame
WITH perspective IS in
WITH border width: = 1
WITH border width: = 1
WITH bight color: = 192,192,192
WITH fill color: = 192,192,192
WITH highlight color: = 255,255,255
WITH shadow color: = 128,128,128
WITH location: = 465,65,530,80

INSTANCE border 11 ISA border
WITH syle IS picture frame
WITH perspective IS in
WITH border width := 1
WITH background color := 192,192,192
WITH fill color := 192,192,192
WITH highlight color := 255,255,255
WITH shadow color := 128,128,128
WITH location := 465,220,550,235

INSTANCE border 12 ISA border
WITH style IS picture frame
WITH perspective IS is
WITH border width:= 1
WITH border width:= 1
WITH blookground color:= 192,192,192
WITH fill color:= 192,192,192
WITH highlight color:= 255,255,255
WITH shadow color:= 128,128,128
WITH location:= 10,10,240,195

INSTANCE border 13 ISA border
WITH style IS shadow
WITH perspective IS in
WITH border width:=6
WITH border width:=6
WITH blockground color:=192,192,192
WITH fill color:=0,0,128
WITH highlight color:=255,255,255
WITH shadow color:=0,0,0
WITH location:=25,25,225,135

INSTANCE combobox I ISA combobox WITH enabled := TRUE WITH son: = FALSE WITH update with current instance := TRUE WITH location := 155,245,295,265

INSTANCE the domain ISA domain WITH sample time := "1996/01/20 14:00"

INSTANCE ISodbe I ISA ISodbe
WITH data source: "build"
WITH connection prompting:= FALSE
WITH auto commit:= TRUE
WITH append:= FALSE
WITH default error handling:= TRUE

INSTANCE ISodbe 2 GENERAL ISA ISodbe WITH data source := "general"
WITH connection prompting := FALSE WITH auto commit := TRUE
WITH append := FALSE
WITH default error handling := TRUE

INSTANCE message box ! ISA message box WITH title := "INPUT ERROR"
WITH text := "You need to input query data"
WITH text := "You need to input query data"
WITH dood style IS app modal
WITH icon IS stop sign
WITH button IS ok
WITH default button IS one

INSTANCE message box 2 ISA message box WITH title: ""INPUT ERROR" WITH text: ""End Time must be bigger than Start Time"
WITH modal style IS app modal WITH icon IS stop sign
WITH button IS ok

WITH default button IS one
INSTANCE navigate 1 ISA navigate

WITH class attachment := discipline message

INSTANCE navigate menu ISA navigate WITH class attachment := menu

INSTANCE navigate temp_current_menu_items ISA navigate WITH class attachment := temp_current_menu_items

INSTANCE the application USA application WITH unknowns fail: TRUE WITH threshold: 50 WITH title display: Scarting Display WITH title display: Scarting Display WITH ignore breakpoints: FALSE WITH reasoning on: FALSE WITH numeric precision: 8 WITH demon strategy IS fire first

INSTANCE column 1 ISA column
WITH strachment: "Change_ID OF
A_change_monitoring_results
WITH width: "55
WITH justify IS center
WITH wordwrap: "TRUE
WITH heading label: "Change
ID"

WITH heading justify IS heading center WITH heading wordwrap := TRUE

INSTANCE column 2 ISA column
WITH attachment := date_time OF
A_change_monitoring_results
WITH width := 120
WITH justify IS center
WITH wordwrap := TRUE
WITH heading label := "Date/Time"
WITH heading justify IS heading center
WITH heading wordwrap := TRUE

INSTANCE column 3 ISA column
WITH attachment := discipline OF
A_change_monitoring_results
WITH width := 40
WITH justify IS center
WITH wordwrap := TRUE
WITH heading label := "Dopt."
WITH heading justify IS heading center
WITH heading wordwrap := TRUE

INSTANCE column 4 ISA column
WITH attachment := object OF
A_change_monitoring_results
WITH width := 100
WITH justify IS center
WITH wordwrap := TRUE
WITH heading label := "component"
WITH heading justify IS heading center
WITH heading wordwrap := TRUE

INSTANCE column 5 ISA column
WITH strachment := object ID OF
A_change_monitoring_results
WITH width := 70
WITH justify IS center
WITH wordwrap := TRUE
WITH heading label := "Comp.ID"
WITH heading justify IS heading center
WITH heading wordwrap := TRUE

INSTANCE column 6 ISA column
WITH strachment: "= attribute OF
A_change_monitoring_results
WITH width: "= 120
WITH justify IS center
WITH wordwrap: "= TRUE
WITH heading label: = "Attribute"
WITH heading justify IS heading center
WITH heading wordwrap: "= TRUE

INSTANCE column 7 ISA column
WITH attachment := old_value OF
A_change_monitoring_results
WITH width := 100
WITH justify IS center
WITH wordwrap := TRUE
WITH heading label := "Old
Value"
WITH heading justify IS heading center
WITH heading wordwrap := TRUE

INSTANCE column 8 ISA column

WITH attachment := new_value OF
A_change monitoring_results
WITH width := 100
WITH justify IS center
WITH wordwrap := TRUE
WITH heading label := "New
Value"
WITH heading justify IS heading center
WITH heading wordwrap := TRUE

INSTANCE column 9 ISA column
WITH attachment := preceeding_change_ID OF
A_change monitoring_results
WITH width := 65
WITH justify IS left
WITH wordwrap := TRUE
WITH heading label := "Prev.
ChangeID"
WITH beading justify IS heading center
WITH heading wordwrap := TRUE

WITH wait: = TRUE
WITH delay changes: = FALSE
WITH items [1]: = border 9
WITH items [2]: = border 6
WITH items [3]: = textbox 1
WITH items [4]: = UNDETERMINED
WITH items [5]: = pushbotton 6
WITH items [6]: = UNDETERMINED
WITH items [7]: = UNDETERMINED
WITH items [7]: = UNDETERMINED
WITH items [7]: = UNDETERMINED
WITH items [8]: = UNDETERMINED
WITH menus [1]: = UNDETERMINED

INSTANCE Starting Display ISA display

INSTANCE query ISA display WITH wait := TRUE WITH delay changes := FALSE
WITH items [1] := radiobutton group 1
WITH items [2] := UNDETERMINED
WITH items [3] := border 3 WITH items [4]:= border 2
WITH items [5]:= textbox 2
WITH items [6]:= border 1
WITH items [7]:= UNDETERMINED WITH items [8]:= textbox 4
WITH items [9]:= textbox 5 WITH items [10]:= textbox 6
WITH items [11]:= UNDETERMINED
WITH items [12]:= UNDETERMINED WITH items [13] := UNDETERMINED
WITH items [14] := valuebox 3
WITH items [15] := textbox 7 WITH items [16] = UNDETERMINED
WITH items [17] = UNDETERMINED
WITH items [18] = UNDETERMINED
WITH items [18] = UNDETERMINED
WITH items [19] = UNDETERMINED WITH items [20] := promptbox 1 WITH items [21] := textbox 9 WITH items [21] := textbox 9
WITH items [22] := textbox 10
WITH items [23] := textbox 11
WITH items [24] := UNDETERMINED
WITH items [25] := promptbox 2
WITH items [26] := UNDETERMINED
WITH items [27] := UNDETERMINED
WITH items [27] := UNDETERMINED WITH items [28] := UNDETERMINED WITH items [29] := UNDETERMINED WITH items [30] := pushbutton 2 WITH items [31]:= pushbutton 3
WITH items [32]:= UNDETERMINED
WITH items [33]:= UNDETERMINED WITH items [34] = UNDETERMINED
WITH items [34] = UNDETERMINED
WITH items [35] = UNDETERMINED
WITH items [36] := UNDETERMINED
WITH items [37] := UNDETERMINED
WITH items [38] := textbox 28
WITH items [39] := promptbox 5
WITH items [40] := UNDETERMINED
WITH items [41] := UNDETERMINED
WITH items [42] := UNDETERMINED
WITH items [43] := textbox 33
WITH items [43] := textbox 33 WITH items [43]:= textbox 33 WITH items [44]:= textbox 34 WITH items [45] = UNDETERMINED
WITH items [46] = UNDETERMINED
WITH items [47] = textbox 35
WITH items [48] = UNDETERMINED WITH items [49]:= textbox 37 WITH items [50]:= textbox 38 WITH items [51] := UNDETERMINED WITH items [52] := textbox 40 WITH items [53] := promptbox 7 WITH items [54] := listbox 1

```
WITH location := 415,360,550,400
   WITH items [55] := listbox 2
                                                                                        WITH show current instance := TRUE
  WITH items [56]:= textbox 36
WITH items [57]:= valuebox 17
WITH items [58]:= valuebox 18
                                                                                        WITH destination := attribute OF
                                                                                       \_monitored_object_input
WITH location := 150,315,310,342
                                                                                                                                                                         INSTANCE pushbutton 14 ISA pushbutton WITH label := "END"
  WITH items [59] := border 10
WITH items [60] := border 11
WITH fill color := 192,192,192
                                                                                                                                                                           WITH attribute attachme
                                                                                                                                                                                                              nt := exit OF the application
                                                                                                                                                                           WITH location := 215.40.365.78
                                                                                     INSTANCE listbox 3 ISA listbox
                                                                                       WITH show current instance := TRUE
                                                                                                                                                                        INSTANCE pushbutton 11 ISA pushbutton WITH label := "OK"
  WITH means [1]:= UNDETERMINED
                                                                                       WITH source := Project_name OF Current_projects
                                                                                       WITH destination := project_name
WITH location := 30,150,215,180
                                                                                                                                                                           WITH attribute attach
 INSTANCE Results ISA display
                                                                                                                                                                                                             at := return to results screen
  WITH wait := TRUE
WITH delay changes := FALSE
WITH items [1] := border 4
                                                                                                                                                                           WITH location := 340,170,545,210
                                                                                     INSTANCE promothox 1 ISA promothox
                                                                                       WITH font := "System"
                                                                                                                                                                         INSTANCE radiobutton group 1 ISA radiobutton
  WITH items [2 ] := textbox 13
WITH items [3 ] := textbox 14
WITH items [4 ] := textbox 15
                                                                                       WITH justify IS center
                                                                                                                                                                        group
WITH pen color := 0,0,0
                                                                                       WITH frame := TRUE
                                                                                                                                                                           WITH fill color := 192,192,192
                                                                                       WITH show current := TRUE
  WITH items [5] := textbox 16
WITH items [6] := textbox 17
WITH items [7] := UNDETERMINED
                                                                                       WITH attachment := start_time OF
                                                                                                                                                                           WITH frame := TRUE
                                                                                                                                                                          WITH group label := "Discipline"
WITH show current := TRUE
                                                                                      \_monitored_object_input
WITH location := 385,295,540,320
  WITH items [8] := UNDETERMINED
                                                                                                                                                                           WITH attachment := select_discipline OF
  WITH items [9] := valuebox 8
WITH items [10] := UNDETERMINED
                                                                                                                                                                          monitored_object_input
WITH location := 320,55,465,245
                                                                                     INSTANCE promptbox 2 ISA promptbox
WITH font := "System"
  WITH items [11] := valuebox 7
WITH items [12] := valuebox 9
                                                                                       WITH justify IS center
                                                                                                                                                                        INSTANCE table 2 ISA table
                                                                                       WITH frame := TRUE
 WITH items [12] := valuebox 9
WITH items [13] := valuebox 10
WITH items [14] := itexibox 18
WITH items [15] := UNDETERMINED
WITH items [16] := itexibox 19
                                                                                                                                                                          WITH attachment := A_change_monitoring_results
                                                                                      WITH show current := TRUE
                                                                                      WITH attachment := end_time OF
                                                                                                                                                                          WITH columns [1] := column 1
                                                                                      A_monitored_object_input
WTTH location := 385,325,540,350
                                                                                                                                                                          WITH columns [2] := column 2
WITH columns [3] := column 3
  WITH items [17] := textbox 20
                                                                                                                                                                          WITH columns [4] := column 4
  WITH items [18]:= valuebox 11
WITH items [19]:= valuebox 12
WITH items [20]:= UNDETERMINED
                                                                                    INSTANCE promptbox 5 ISA promptbox WITH font := "System"
                                                                                                                                                                          WITH columns [5] := column 5
WITH columns [6] := column 6
                                                                                      WITH justify IS center
                                                                                                                                                                          WITH columns [7] := column 7
 WITH items (21) - UNDETERMINED
WITH items (21) - table 2
WITH items (22) - table 2
WITH items (23) - UNDETERMINED
WITH items (24) - UNDETERMINED
WITH items (25) - UNDETERMINED
                                                                                                                                                                          WITH columns [8] := column 8
WITH columns [9] := column 9
                                                                                      WITH frame := TRUE
                                                                                      WITH show current := TRUE
                                                                                                                                                                         WITH heading := TRUE
WITH heading height := 40
WITH fill color := 0,255,255
                                                                                      WITH attachment := Change_ID OF
                                                                                     A_monitored_object_input
WITH location := 20,115,135,140
  WITH items [26]:= pushbutton 15
WITH items [27]:= UNDETERMINED
WITH items [28]:= textbox 29
                                                                                                                                                                         WITH column lines := TRUE
WITH row lines := TRUE
                                                                                    INSTANCE promptbox 7 ISA promptbox
WITH pen color := 0,0,0
WITH fill color := 192,192,192
WITH font := "System"
                                                                                                                                                                          WITH frame := TRUE
  WITH items [29]:= tenthox 31
WITH fill color:= 192,192,192
WITH mesus [1]:= UNDETERMINED
                                                                                                                                                                          WITH show current instance := TRUE
                                                                                                                                                                          WITH location := 10,185,550,350
                                                                                      WITH justify IS center
                                                                                      WITH frame := FALSE
WITH show default := TRUE
                                                                                                                                                                        INSTANCE text style I ISA text style
                                                                                                                                                                          WITH font := Times New Roman
INSTANCE messages ISA display
  WITH wait := TRUE
                                                                                      WITH show current := TRUE
                                                                                                                                                                          WITH size := 10
 WITH delay changes: = FALSE
WITH items [1]: = border 8
WITH items [2]: = border 7
WITH items [3]: = textbox 21
WITH items [4]: = UNDETERMINED
WITH items [5]: = UNDETERMINED
                                                                                                                                                                        WITH font style IS bold CF FALSE, italic CF
FALSE, underline CF FALSE, strikeout CF FALSE
                                                                                      WITH attachment := sample time
WITH location := 400,355,540,380
                                                                                    INSTANCE pushbutton 2 ISA pushbutton
                                                                                                                                                                       INSTANCE textbox 1 ISA textbox
WITH pen color := 255,255,255
WITH fill color := 0,0,128
                                                                                      WITH label := "SHOW RESULTS"
                                                                                      WITH attribute attachment := show results
                                                                                                                                                                       WITH justify IS center
WITH font: "Times New Roman"
WITH font style IS bold, italic CF FALSE, underline
CF FALSE, strikeout CF FALSE
 WITH items [6]:= pushbutton 7
WITH items [7]:= pushbutton 8
WITH items [8]:= pushbutton 9
                                                                                      WITH location = 155 60 320 105
                                                                                    INSTANCE pushbutton 3 ISA pushbutton
 WITH items [9] := pushbutton 10
WITH items [10] := textbox 22
WITH items [11] := textbox 23
                                                                                      WITH label - "END"
                                                                                      WITH attribute attachment := exit OF the application
                                                                                                                                                                          WITH font size := 14
                                                                                                                                                                          WITH frame := FALSE
                                                                                      WITH location := 155,110,320,155
 WITH items [12] = textbox 24
WITH items [13] = textbox 25
WITH items [14] = textbox 25
WITH items [14] = textbox 26
WITH items [15] = textbox 27
WITH items [16] = valuebox 13
WITH items [17] = valuebox 14
                                                                                                                                                                         WITH text **
                                                                                                                                                                       CHANGE TRACKING MODULE*
                                                                                    INSTANCE pushbutton 6 ISA pushbutton
                                                                                      WITH label := "Go Ahead"
                                                                                                                                                                          WITH location := 25,25,170,130
                                                                                      WITH attribute attachment := go_ahead
                                                                                      WITH location := 25,160,175,192
                                                                                                                                                                        INSTANCE textbox 2 ISA textbox
                                                                                                                                                                         WITH pen color := 255,255,255
WITH items [17] := valuebox 14
WITH items [18] := valuebox 15
WITH items [18] := UNDETERMINED
WITH items [20] := UNDETERMINED
WITH items [21] := UNDETERMINED
WITH items [21] := pushbutton 11
WITH items [22] := pushbutton 11
WITH items [24] := valuebox 16
WITH items [24] := valuebox 16
WITH fill color := 192,192,192
WITH menus [1] := UNDETERMINED
                                                                                    INSTANCE pushbutton 7 ISA pushbutton WITH label := "Read First Message"
                                                                                                                                                                         WITH fill color = 128.0.0
                                                                                                                                                                         WITH justify IS center
WITH foat := "Times New Rom
                                                                                      WITH attribute attachment := action OF navigate 1 IS
                                                                                    first
WITH location := 350,70,535,105
                                                                                                                                                                         WITH font style IS bold, italic CF FALSE, underline
                                                                                                                                                                        CF FALSE, strikeout CF FALSE
                                                                                                                                                                         WITH font size := 18
WITH frame := TRUE
                                                                                    INSTANCE pushbutton $ ISA pushbutton
                                                                                                                                                                          WITH text := "SELECTION OF DESIGN
                                                                                      WITH label := "Next"
                                                                                      WITH attribute attachment := action OF navigate 1 IS
                                                                                                                                                                        CHANGES*
                                                                                                                                                                          WITH location := 10.10.550.50
INSTANCE select_project_display ISA display WITH wait := FALSE
                                                                                      WITH location := 350,105,410,135
                                                                                                                                                                        INSTANCE textbox 4 ISA textbox
 WITH delay changes := FALSE
WITH items [1] := border 12
                                                                                    INSTANCE pushbutton 9 ISA pushbutton
                                                                                                                                                                         WITH pen color := 0,0,0
WITH fill color := 192,192,192
                                                                                      WITH label := "Prev."
 WITH items [2]:= border 13
WITH items [3]:= textbox 41
WITH items [4]:= listbox 3
                                                                                                                                                                         WITH justify IS left
WITH font := "System"
WITH text := "Building Comp.:"
                                                                                      WITH attribute attachment := action OF navigate 1 IS
                                                                                      WITH location := 410,105,470,135
  WITH fill color := 192,192,192
                                                                                                                                                                          WITH location := 15,180,155,200
 WITH menus [1] := UNDETERMINED
                                                                                    INSTANCE pushbutton 10 ISA pushbutton
                                                                                      WITH label := "Last"
                                                                                                                                                                        INSTANCE textbox 5 ISA textbox
                                                                                                                                                                         WITH pen color := 0,0,0
WITH fill color := 192,192,192
INSTANCE listbox I ISA listbox
                                                                                      WITH attribute attachment := action OF navigate 1 IS
 WITH show current instance := TRUE
                                                                                      WITH location := 470,105,533,135
                                                                                                                                                                          WITH justify IS left
 WITH destination := object_ID OF
A_monitored_object_input
WTTH location := 150,235,310,260
                                                                                                                                                                         WITH font := "System"
WITH text := "Building Comp. ID:"
                                                                                    INSTANCE pushbutton 15 ISA pushbutton
                                                                                      WITH label -= "OK"
                                                                                                                                                                          WITH location := 20,240,180,260
INSTANCE listbox 2 ISA listbox
                                                                                      WITH attribute attachment := return to query screen
```

INSTANCE textbox 6 ISA textbox WITH pen color := 0,0,255 WITH fill color := 192,192,192 WITH justify IS left
WITH font := "System"
WITH text := "Time Period:" WITH location := 40,70,120,90 INSTANCE textbox 29 ISA textbox WITH justify IS left
WITH font := "System"
WITH font style IS bold, italic, underline CF FALSE, WITH location := 20,115,125,140 WITH pen color := 0,0,0 WITH fill color := 192,192,192 INSTANCE textbox 19 ISA textbox WITH justify IS left strikeout CF FALSE WITH pen color := 0,0,0 WITH fill color := 192,192,192 WITH font := "System"
WITH text := "From project start time OR" WITH font size := 5 WITH text := "Define from the upper Menu" WITH justify IS left WITH location := 20,140,270,160 WITH location := 20,205,210,225 WITH font := "System"
WITH text := "From:" INSTANCE textbox 31 ISA textbox WITH pen color := 0,0,0 WITH fill color := 0,255,255 WITH justify IS center **INSTANCE** textbox 7 ISA textbox WITH location := 275,115,335,140 WITH pen color := 0,0,0 WITH fill color := 192,192,192 INSTANCE textbox 20 ISA textbox WITH pen color := 0,0,0 WITH fill color := 192,192,192 WITH justify IS left WITH font := "System"
WITH font style IS bold
WITH frame := TRUE WITH justify IS left WITH font := "System"
WITH text := "Comp. Attribute:"
WITH location := 20,315,165,340 WITH font := "System WITH text := "To:" WITH text := "Double click to retreive the massages WITH location := 275,140,305,165 INSTANCE textbox 9 ISA textbox by the system as a response to a design change WITH location := 5,360,410,400 WITH pen color := 0,0,255 WITH fill color := 192,192,192 INSTANCE textbox 21 ISA textbox WITH pen color := 255,255,255 WITH fill color := 128,0,0 INSTANCE textbox 32 ISA textbox WITH justify IS left WITH font := "System"
WITH font style IS italic WITH pen color := 255,255,255 WITH fill color := 255,0,0 WITH justify IS center WITH font := "Times New Roman"
WITH font style IS bold, italic CF FALSE, underline
CF FALSE, strikeout CF FALSE WITH text := "Time Period" WITH justify IS center WITH foot: = "Times New Roman"
WITH foot style IS bold, italic CF FALSE, underline
CF FALSE, strikeout CF FALSE WITH location := 340,250,430,270 WITH font size := 18
WITH frame := TRUE
WITH text := "RELATED MESSAGES" INSTANCE textbox 10 ISA textbox WITH pen color := 0,0,0 WITH fill color := 192,192,192 WITH font size := 36 WITH frame := TRUE
WITH text := "END OF MONITORING MODULE" WITH justify IS left WITH location := 10,10,545,50 WITH font := "System WITH text := "From" WITH location := 77.87.548.307 INSTANCE textbox 22 ISA textbox Will pen color := 0,0,255 WITH fill color := 192,192,192 INSTANCE textbox 33 ISA textbox WITH location := 340,300,380,320 WITH pen color := 0,0,0 WITH fill color := 192,192,192 WITH justify IS left
WITH font := "System"
WITH text := "Select Message INSTANCE textbox 11 ISA textbox INSTANCE textbox II ISA textb
WITH pen color:= 0,0,0
WITH fill color:= 192,192,192
WITH justify IS left
WITH font:= "System"
WITH text:= "To" WITH justify IS left WITH font := "System"
WITH text := "From Project Start Time" WITH location := 390,140,500,160 WITH location := 340,270,500,290 INSTANCE textbox 23 ISA textbox WITH location := 340,330,375,350 WITH justify IS left INSTANCE textbox 34 ISA textbo WITH font := "System"
WITH frame := TRUE
WITH text := " Read and Acted Upon Message (Black WITH pen color := 255,0,0 WITH fill color := 192,192,192 INSTANCE textbox 13 ISA textbox WITH pen color := 255,255,255 WITH fill color := 128,0,0 WITH justify IS left WITH font := "System"
WITH text := "OR" WITH justify IS center
WITH font := "Times New Roman"
WITH font style IS bold, italic CF FALSE, underline
CF FALSE, strikeout CF FALSE WITH location := 10,95,330,120 WITH location := 515,270,540,295 INSTANCE textbox 24 ISA textbox WITH pen color := 0,0,255 WITH fill color := 255,255,255 INSTANCE textbox 35 ISA textbox WITH foat size := 18 WITH pen color := 0,0,255 WITH fill color := 192,192,192 WITH frame := TRUE
WITH text := "QUERY RESULTS" WITH justify IS left WITH font := "System WITH justify IS left WITH frame := TRUE
WITH text := * Un Read and Un Acted Upon Message WITH font := "System"
WITH font style IS bold, italic, underline CF FALSE, WITH location := 5,5,555,45 INSTANCE textbox 14 ISA textbox strikeout CF FALSE NSTANCE textion 14 150,00
WITH pen color := 0,0,0
WITH fill color := 192,192,192
WITH justify IS left
WITH font := "System"
WITH text := "Discipline"
WITH location := 20,60,90,85 WITH location := 10,125,330,150 WITH font size := 5 WITH text := "Select by double click" INSTANCE textbox 25 ISA textbox WITH pen color := 255,0,0 WITH fill color := 255,255,255 WITH location := 20,260,175,305 WITH justify IS left INSTANCE textbox 37 ISA textbox WITH font := "System'
WITH frame := TRUE WITH pen color := 0,0,255 WITH fill color := 192,192,192 INSTANCE textbox 15 ISA textbox WITH pen color := 0,0,0 WITH fill color := 192,192,192 WITH text := * Read but Un Acted Upon Message WITH justify IS left WITH font := "System"
WITH font style IS bold, italic, underline CF FALSE, WITH justify IS left
WITH font := "System"
WITH text := "Building Component" WITH location := 10.155,330,180 strikeout CF FALSE WITH foot size := 5
WITH text := "Input" INSTANCE textbox 26 ISA textbox WITH location := 100,60,240,80 WITH pen color := 0,0,255 WITH fill color := 192,192,192 WITH location := 40,90,90,110 INSTANCE textbox 16 ISA textbox WITH justify IS left INSTANCE textbox 38 ISA textbox WITH pen color := 0,0,0 WITH fill color := 192,192,192 WITH font := "System"
WITH text := "# of Messages" WITH pen color := 0,0,128 WITH fill color := 0,255,255 WITH justify IS left WITH location := 10,185,140,210 WITH justify IS left
WITH font := "Times New Roma WITH font := "System"
WITH text := "Component ID"
WITH location := 255,60,350,85 INSTANCE textbox 27 ISA textbox WITH font style IS bold CF FALSE, italic CF WITH pen color := 0,0,255 WITH fill color := 192,192,192 FALSE, underline CF FALSE, strikeout CF FALSE INSTANCE textbox 17 ISA textbox WITH justify IS left WITH font size := 10 WITH pen color := 0,0,0 WITH fill color := 192,192,192 WITH font := "System"
WITH text := "# of Current Message" WITH frame := TRUE WITH text := WITH justify IS left WITH location := 155,185,300,210 WITH font := "System"
WITH text := "Component Attribute"
WITH location := 395,60,530,85 INSTANCE textbox 28 ISA textbox - Discipline WITH pen color := 0,0,0 WITH fill color := 192,192,192 - Change ID OR

WITH justify IS left

WITH font := "System"
WITH text := "Change ID:"

INSTANCE textbox 18 ISA textbox WITH pen color := 0,0,0 WITH fill color := 192,192,192

WITH location := 465,90,550,210

INSTANCE textbox 39 ISA textbox
WITH pen color := 0,0,255
WITH fill color := 192,192,192
WITH justify IS left
WITH font := "System"
WITH text := "Discipline which received the
message:"
WITH location := 10,65,335,90

INSTANCE textbox 40 ISA textbox WITH pen color := 0,0,0 WITH fill color := 192,192,192 WITH justify IS left WITH font := "System" WITH text := "Sample:" WITH location := 340,360,405,380

INSTANCE textbox 36 ISA textbox
WITH pen color := 0,0,255
WITH fill color := 192,192,192
WITH justify IS left
WITH font := "System"
WITH font style IS bold, italic, underline CF FALSE, strikout CF FALSE
WITH font size := 5
WITH sext := "Select
by double click"
WITH location := 20,335,135,375

INSTANCE textbox 41 ISA textbox
WITH pen color: = 255,255,255
WITH fill color: = 0,0,128
WITH justify IS center
WITH foat: = "Times New Roman"
WITH foat style IS bold, italic CF FALSE, underline
CF FALSE, strikeout CF FALSE
WITH foat size: = 18
WITH foat size: = 18
WITH text: = "Select a Project from the List"
WITH location: = 35,50,210,115

INSTANCE valuebox 2 ISA valuebox WITH justify IS left WITH foot := "System" WITH frame := TRUE WITH clopped := TRUE WITH clopped := TRUE WITH clopped := 160,170,290,203

INSTANCE valuebox 3 ISA valuebox
WITH pen color := 0,0,128
WITH fill color := 192,192,192
WITH justify IS center
WITH font := "System"
WITH film := "TRUE
WITH clipped := TRUE
WITH stachment := object OF
A monitored object input
WITH location := 150,175,310,200

INSTANCE valuebox 8 ISA valuebox WITH pen color: = 0,0,255 WITH pen color: = 192,192,192 WITH justify IS center WITH font: = "System" WITH frame: = TRUE WITH clipped: = TRUE WITH strachment: = object_ID OF A monitored_object_input WITH location: = 240,80,370,105

INSTANCE valuebox 7 ISA valuebox WITH pen color: = 0,0,255
WITH fill color: = 192,192,192
WITH justify IS center
WITH fint: = "System"
WITH firame: = TRUE
WITH clipped: = TRUE
WITH attachment: = stribute OF
A_monitored_object_input
WITH location: = 375,80,545,105

INSTANCE valuebox 9 ISA valuebox WITH pen color := 0,0,255 WITH fill color := 192,192,192 WITH justify IS center
WITH foot: = "System"
WITH frame: = TRUE
WITH clipped:= TRUE
WITH stackment: = object OF
A_monitored_object_input
WITH location:= 95,80,235,105

INSTANCE valuebox 10 ISA valuebox
WITH pen color: = 0,0,255
WITH fill color: = 192,192,192
WITH justify IS center
WITH font: = "System"
WITH frame: = TRUE
WITH clipped: = TRUE
WITH stackment: = discipline OF
A_monitored_object_input
WITH location: = 20,80,90,105

INSTANCE valuebox 11 ISA valuebox WITH pen color: = 255,0,0
WITH fill color: = 192,192,192
WITH justify IS left
WITH front: = "System"
WITH front: = "TRUE
WITH stachment: = end_time OF
A_monitored_object_input
WITH location: = 350,140,345,165

INSTANCE valuebox 12 ISA valuebox
WITH pen color: = 255,0,0
WITH fill color: = 192,192,192
WITH justicy IS left
WITH font: = "System"
WITH frame: = TRUE
WITH disped: = TRUE
WITH attachment: = start_time OF
A_monitored_object_input
WITH location: = 350,110,545,135

INSTANCE valuebox 13 ISA valuebox
WITH pen color := 0,0,255
WITH fill color := 192,192,192
WITH justify IS center
WITH foot := "System"
WITH frame := TRUE
WITH clipped := TRUE
WITH attachment := number instances OF navigate 1
WITH location := 115,185,145,210

WITH pen color:=0,0,255
WITH fill color:=192,192,192
WITH justify IS center
WITH font:="System"
WITH frame:=TRUE
WITH clipped:=TRUE
WITH dipped:=TRUE
WITH attachment:=instance number OF navigate i
WITH location:=300,185,330,210

INSTANCE valuebox 14 ISA valuebox

INSTANCE valuebox 15 ISA valuebox
WITH pen color := 0,0,0
WITH fill color := 192,192,192
WITH justify IS left
WITH font := "System"
WITH font style IS bold, italic CF FALSE, underline
CF FALSE, strikeout CF FALSE
WITH font size := 10
WITH frame := FALSE
WITH clipped := TRUE
WITH scroll := TRUE
WITH borizontal scroll := TRUE
WITH location := 20,230,535,360

INSTANCE valuebox 16 ISA valuebox WITH justify IS center WITH font: "" "System" WITH frame: "" TRUE WITH clipped: "" TRUE WITH location: " 275,65,330,90

INSTANCE valuebox 17 ISA valuebox

WITH pen color := 0,0,128
WITH fill color := 192,192,192
WITH justify IS center
WITH font := "System"
WITH frame := TRUE
WITH clipped := TRUE
WITH attachment := object_ID OF
A_monitored_object_input
WITH location := 150,265,310,290

INSTANCE valuebox 18 ISA valuebor
WITH pen color := 0,0,128
WITH fill color := 192,192,192
WITH justify IS center
WITH font := "System"
WITH frame := TRUE
WITH clipped := TRUE
WITH attachment := attribute OF
A monitored object_input
WITH location := 150,345,310,370

INSTANCE main window ISA window WITH location: = 172,137,386,370 WITH menus [1]: = UNDETERMINED WITH style IS moveable, sizeable, closeable WITH title: = "Change-Monitoring Module" WITH visible: = TRUE

INSTANCE expand window ISA window WITH location: = 33,15,604,454 WITH menus [1]:= UNDETERMINED WITH style IS moveable, sizeable, closeable CF FALSE WITH title: = "Change-Monitoring Module" WITH visible: = FALSE

INSTANCE messages window ISA window WITH location: = 41,43,601,448 WITH menus [1]:= UNDETERMINED WITH style IS moveable, sizeable, closeable CF FALSE WITH title: = "Change-Monitoring Module" WITH visible:= FALSE

INSTANCE Query window ISA window
WITH location: 33,15,604,454
WITH menus [1]:= UNDETERMINED
WITH style IS moveable, sizeable, closeable CF
FALSE
WITH title: "Change-Monitoring Module"
WITH visible: = FALSE
WITH output: "query

INSTANCE Select a project window ISA window WITH location := 151,149,407,383 WITH menus [1]:= UNDETERMINED WITH style IS moveable, sizeable, closeable WITH title := ""
WITH visible := FALSE

! DEMON GROUP: selected OF menu item

DEMON 1
IF selected OF menu item
THEN effect of menu item := TRUE

! DEMON GROUP: double clicked OF table

DEMON 2

IF double clicked OF table 2

THEN numeric_Change_ID := TO NUMERIC(
Change_ID OF A_change_monitoring_results)

AND EXEC ISodbe 1 SQL select * from
message_elements WHERE sent_because_change_ID

= :numeric_Change_ID

END SQL INTO

discipline_message
AND go to message screen := TRUE

END

C.5. VIRTUAL CHANGE MODULE

```
SVERSION35
                                                                          Structure
                                                                                                                                               WITH attribute of building el
                                                                                                                                               WITH affected discipline STRING
WITH statement of selection STRING
 SLOCATIONS ARE PIXELS
                                                                           Envelope.
                                                                          HVAC
 CLASS A_attributes_of_monitored_object
WITH name STRING
                                                                                                                                               WITH AB specific rules id NUMERIC
                                                                          Interior
                                                                           WHEN CHANGED
                                                                                                                                              CLASS border INHERITS add on, display item
 CLASS A_change_data
WITH CHANGE STRING
                                                                            BEGIN
                                                                                                                                               WITH style COMPOUND
                                                                                                                                                picture frame
  WITH Required man hours NUMERIC
WITH Difference in Build Cost NUMERIC
WITH Recommendation NUMERIC
                                                                             ** reset for accepting a first change
CONF(yes for second alternative) := -1
                                                                                                                                                 edit control.
                                                                             CONF(first_alternative_value) := -1
                                                                                                                                                 chadow
  WITH Reason_for_recomm STRING
WITH object STRING
                                                                                                                                                INIT nicture frame
                                                                                                                                               WITH perspective COMPOUND
                                                                             IF select_discipline OF A_Responding_Data IS
  WITH object_ID STRING
WHEN CHANGED
                                                                                  ure THEN
                                                                              BEGIN
                                                                                                                                                INIT in
                                                                                discipline OF A_Responding_Data := "arch"
     BEGIN
IF object_ID OF A_change_data <> ** AND attribute OF A_change_da\
ta <> ** THEN
                                                                                                                                               WITH border width NUMERIC
                                                                                                                                                INIT I
                                                                                                                                               WITH background color COLOR
INIT 192,192,192
                                                                             IF select_discipline OF A_Responding_Data IS
       BEGIN
                                                                               m THÊN
                                                                                                                                               WITH fill color COLOR
         check existing data := TRUE
                                                                              BEGIN
                                                                                                                                                INIT 192,192,192
                                                                                discipline OF A_Responding_Data := "stru"
       END
                                                                                                                                               WITH highlight color COLOR.
                                                                                                                                                INIT 255,255,255
  WITH attribute STRING
WHEN CHANGED
                                                                                                                                               WITH shadow color COLOR
INIT 128,128,128
                                                                             IF select_discipline OF A_Responding_Data IS
     BEGIN
                                                                             ioos THEN
                                                                              BEGIN
       IF object_ID OF A_change_data <> ** AND
attribute OF A_change_da\
ta <> ** THEN
                                                                                                                                                 ASS combobox INHERITS add on, display item
                                                                                discipline OF A_Responding_Data := "enve"
                                                                                                                                              WITH attachment ATTRIBUTE REFERENCE
WITH selected SIMPLE
       RECIN
                                                                       IF select_discipline OF A_Responding_Data IS HVAC THEN
                                                                                                                                               WITH enabled SIMPLE
         check existing data := TRUE
        END
                                                                                                                                                INTETRUE
                                                                                                                                               WITH sort SIMPLE
     END
                                                                              BEGIN
  WITH sending time STRING
WITH old value STRING
WITH new value STRING
WITH Case ID NUMERIC
                                                                                                                                                INIT FALSE
                                                                                discipline OF A Responding Data := "hvac"
                                                                                                                                               WITH update with current instance SIMPLE
                                                                                                                                                INIT TRUE
                                                                             IF select_discipline OF A_Responding_Data IS nination THEN
                                                                                                                                             CLASS Current_projects
WITH Project_name STRING
  WITH PREDECESSOR STRING
  WITH DURATION NUMERIC
                                                                              BEGIN
  WITH alternative CHANGE STRING
                                                                                discipline OF A_Responding_Data := "illu"
                                                                                                                                             CLASS discipline_message
  WITH date_time STRING
                                                                              END
                                                                                                                                               WITH changed_attribute STRING
                                                                       .
IF select_discipline OF A_Responding_Data IS
Interior THEN
CLASS A_change_data_retrieved
WITH CHANGE STRING
                                                                                                                                              WITH changed_class STRING
WITH changed_instance_id STRING
  WITH Required man hours NUMERIC WITH Difference in Build_Cost NUMERIC
                                                                              BEGIN
                                                                                                                                               WITH old_value STRING
                                                                                discipline OF A_Responding_Data := "inte"
                                                                                                                                              WITH new_value STRING
WITH affected_attribute STRING
  WITH Recommendation NUMERIC
 WITH Reason for recomm STRING
WITH object STRING
WITH object ID STRING
WITH attribute STRING
                                                                                                                                               WITH affected_class STRING
                                                                                                                                              WITH affected instance id STRING
WITH relationship STRING
                                                                       IF select_discipline OF A_Responding_Data = FALSE THEN
                                                                                                                                              WITH writing date TIME
WITH reading date TIME
                                                                              BEGIN
  WITH DURATION NUMERIC
                                                                               discipline OF A_Responding_Data := **
                                                                                                                                               WITH been read STRING
  WITH CASE_ID NUMERIC
                                                                                                                                              WITH been_acted_upon STRING
WITH id NUMERIC
  WITH date_time STRING
  WITH old_value STRING
WITH new_value STRING
                                                                         **** retrieve the messages from
                                                                                                                                               WITH discipline_to STRING
WITH discipline_from STRING
                                                                       [A_virtual_message_elements] table
EXEC ISodbc 1 SQL select * from
CLASS A message components
WITH changed attribute STRING
WITH changed class STRING
WITH changed instance id STRING
WITH old value STRING
WITTH and waker STRING
                                                                                                                                              WITH rule writting date STRING WITH violated rule id NUMERIC
                                                                           virtual_message_elements who
                                                                        e discipline_to = :A_Responding_Data.discipline AND
                                                                                                                                              WITH general specific STRING
WITH Case ID NUMERIC
                                                                       been_acted_upon = :d\
                                                                       omain.SNo
                                                                                                                                              WITH sent_because_change_ID STRING
WITH cause_of_the_effect STRING
                                                                                        END SQL INTO discipline_message
  WITH new_value STRING
WITH affected_attribute STRING
                                                                         **** put the first message as default action OF navigate 1 IS first := TRUE
                                                                                                                                             CLASS ISodbc INHERITS add on
  WITH affected class STRING
WITH affected instance id STRING
WITH relationship STRING
                                                                                                                                               WITH data source STRING
                                                                                                                                               WITH userid STRING
                                                                        leeeee end of method
                                                                           END
                                                                                                                                               WITH password STRING
  WITH writting_date STRING
WITH discipline_to STRING
                                                                         WITH CASE_ID NUMERIC
                                                                                                                                               WITH connection string STRING
                                                                                                                                               WITH connection prompting SIMPLE
  WITH discipline_from STRING
                                                                         WITH PREDECESSOR STRING
                                                                                                                                                INIT FALSE
  WITH general_specific STRING
WITH rule_writting_date STRING
                                                                                                                                               WITH auto com
                                                                        CLASS A_Violated_GENERAL_rule
  WITH violated_rule_id NUMERIC WITH Case_ID NUMERIC
                                                                         WITH percent_change NUMERIC
WITH affected_discipline STRING
                                                                                                                                               WITH action COMPOUND
  WITH sent_because_change_ID STRING
WITH cause_of_the_effect STRING
                                                                         WITH cause_of the_effect STRING
                                                                                                                                                 disconnect
                                                                                                                                               WITH records NUMERIC
                                                                        CLASS A Violated SPECIFIC rule
CLASS A_object_ID_of_monitored_object WITH ID STRING
                                                                         WITH related_class STRING
                                                                                                                                               WITH transaction COMPOUND
                                                                         WITH related instance id STRING
WITH related attribute STRING
                                                                                                                                                commit.
CLASS A_Responding_Data
WITH discipline STRING
WITH select_discipline COMPOUND
                                                                         WITH relationship STRING
                                                                                                                                               WITH append SIMPLE INIT FALSE
                                                                         WITH writting_date STRING
WITH building_element STRING
                                                                                                                                               WITH SEEUS STRING
```

WITH building_element_id STRING

Architecture

WITH show error SIMPLE

```
WITH default error handling SIMPLE
                                                            WITH all id of related class STRING
                                                                                                                       action OF ISodbe 1 IS connect := TRUE
  DUT TRUE
                                                             ARRAY SIZE I
  WITH error message STRING
                                                            WITH current class id attribute name STRING
                                                            WITH related class id attribute name STRING
                                                                                                                     ***** Load the menus in the screens
  WITH native error NUMERIC
  WITH trace file STRING
                                                            WITH current attribute designer STRING
                                                            WITH old value STRING
                                                                                                                       EXEC ISodbe 1 SOL SELECT
 CLASS message box INHERITS add on WITH title STRING
                                                            WITH new value STRING
                                                                                                                    DISTINCT(main_menu_label) from [AB-menu_il
 CLASS m
                                                           WITH related attribute designer STRING
WITH designers of related class attributes STRING
                                                                                                                    tem]
  WITH text STRING
                                                                                                                            END SQL INTO menu (label)
  WITH variable text STRING
                                                             ARRAY SIZE I
  WITH modal style COMPOUND
                                                                                                                       EXEC ISodbc 1 SQL SELECT label from [AB-
                                                          CLASS required_message
WITH discipline name COMPOUND
  app modal,
                                                                                                                    menu item i
   sys model,
                                                                                                                            END SQL INTO menu item
  not model
                                                             Architecture
 INIT app model
WITH icos COMPOUND
                                                             HVAC.
                                                                                                                       IF (number instances OF navigate menu > 0) THEN
                                                             Envelope,
  exclamation
  information,
                                                                                                                         RESIZE menus OF new case window := number
                                                             Structure
  question,
                                                            WITH message status COMPOUND
                                                                                                                    instances OF navigar\
                                                             All Messages,
   stop sign
                                                                                                                    e menu
  INIT exclamation
                                                             UnRead Yes.
  WITH button COMPOUND
                                                             UnActed Upon
                                                           WITH choosen_discipline STRING
WITH message builder STRING
                                                                                                                         FOR (nn := 1 TO number instances OF
  abort retry ignore.
                                                                                                                    navigate menu)
  ok cancel
                                                                                                                          BEGIN
                                                          CLASS temp_current_menu_items
WITH label STRING
  netry cancel.
                                                                                                                           instance number OF navigate menu :- on
  yes no.
  yes no cancel
INIT ok
                                                                                                                    ! **** load the menu items in
                                                          ATTRIBUTE on NUMERIC
                                                                                                                           EXEC (Sodbe 1 SOL SELECT label from
 WITH defa
             ult button COMPOUND
                                                          ATTRIBUTE complete as monitored object SIMPLE WHEN CHANGED
                                                                                                                    [AB-menu_item] WHERE\
                                                                                                                    (main_menu_label) = :menu.label
                                                                                                                                             END SOL INTO
  two.
                                                            BEGIN
                                                                                                                    temp current mel
                                                         %yyy := 0
! **** in case of (RESPONDING) check that a
  INIT one
                                                                                                                    nu_items (label)
 WITH button selected COMPOUND
                                                                                                                           RESIZE items OF menu : number instances
                                                          discipline has been selected
                                                                                                                   OF navigate temp_\
current_menu_items
  abort,
                                                             IF (new case or responding " "responding") THEN
                                                                                                                           FOR (%ann := 1 TO number instances OF
  ignore.
                                                              BEGIN
                                                               IF ( CONF(discipline OF A Responding Data)
  no,
                                                                                                                    mavigrate temp our\
  ok,
                                                          <0) THEN
                                                                                                                   rent_menu_items)
                                                                REGIN
  retry,
                                                                                                                             REGIN
                                                                  text OF message box 2 := "Select a discipline
                                                                                                                             instance number OF navigate
  yes,
                                                                                                                    temp_current_menu_items :=\
  cancell
 WITH put up SIMPLE
                                                                  put up OF message box 2 := TRUE
                                                                                                                             FIND menu item
                                                                   %ууу := I
CLASS paviente INHERITS add on
                                                                 END
                                                                                                                               WHERE label OF menu item = label OF
 WITH class attachment CLASS REFERENCE
                                                              END
                                                                                                                   temp current mel
 WITH action COMPOUND
                                                                                                                   nu_items
  first,
                                                                                                                              FIND END
                                                             IF (%yyy = 0) THEN
BEGIN
                                                                                                                             items[ %nnn] OF menu := menu item
  previous
  next,
                                                           **** get the ID of the object from the database
  le st
  WHEN CHANGED
                                                               general_statement_of_selection := CONCAT(
                                                                                                                           menus[nn] OF new case window := menu
   BEGIN
                                                           select ", object OF\
    IF instance number OF navigate 1 > 0 THEN
                                                          A_change_data, "_id from ", object OF
      BEGIN
                                                          A_change_data)
                                                                                                                        END
                                                               EXEC ISodbe I SQL WITH
***** reset for accepting a first alternative CONF(yes for second alternative) := -1
                                                         general_statement_of_selection END SQL \
INTO A_object_ID_of_monitored_object (ID)
                                                                                                                      visible OF Select a project window := FALSE visible OF new case window := TRUE
       CONF(first_alternative_value) := -1
                                                                source OF listbox 1 := ID OF
                                                                                                                        ew case or responding := "NEW"
IF general_specific OF discipline_message = "GENERAL" THEN
                                                          A_object_ID_of_monitored_object
                                                               source OF listbox 3 := ID OF
                                                                                                                   ! **** get the latest case id number and increase it by
       general message := TRUE
                                                           _object_ID_of_monitored_object
                                                                                                                      EXEC |Sodbe | SOL select MAX (CASE |ID) from
                                                                                                                   IF general_specific OF discipline_message = "SPECIFIC" THEN
                                                                get the attributes of the object from the database
                                                               EXEC ISodbc | SQL select
       specific message := TRUE
                                                          attribute of building element from [A\
                                                                                                                   (Case ID)
                                                          B-attribute-designer] where building_element =
                                                                                                                      IF CONF(Case_ID) < 0 THEN
     END
                                                          :A_change_data.object
                                                                                                                      Case_ID := 0
Case_ID := Case_ID + 1
   END
                                                                             END SQL INTO
 WITH instance number NUMERIC
                                                            _attributes_of_monitore\
                                                                                                                   end of method
 WITH number instances NUMERIC
                                                          d object (name)
                                                                                                                     END
CLASS relations building
                                                                 ource OF listbox 2 := name OF
                                                                                                                    ATTRIBUTE responding to change SIMPLE
 WITH current class STRING
                                                                                                                    WHEN CHANGED
                                                           attributes_of_monitored_objec\
 WITH current instance id STRING
 WITH current attribute name STRING
WITH current display INSTANCE REFERENCE
                                                                 ource OF listbox 4 := name OF
                                                                                                                      action OF ISodbe 1 IS connect := TRUE
                                                           _attributes_of_monitored_objec\
 WITH related class STRING
 WITH related attribute name STRING
                                                                                                                   ! **** Load the menus in the screens
 WITH related instance id STRING
 WITH Relationship types COMPOUND
                                                               Case_ID OF A_change_data := Case_ID
  Equal to,
                                                              END
                                                                                                                       EXEC ISodbe 1 SQL SELECT
                                                          lees end of method
  More than.
                                                                                                                   DISTINCT(main_menu_label) from [AB-menu_il
  Less than,
                                                           END
  Exist with
                                                          ATTRIBUTE general_statement_of_selection STRING
                                                                                                                           END SOL INTO menu (label)
                                                         ATTRIBUTE start new case SIMPLE
WHEN CHANGED
  Between values
 WITH Chosen Relationship type STRING
                                                                                                                       EXEC ISodbc I SQL SELECT label from [AB-
 WITH columns headings STRING
                                                            BEGIN
                                                                                                                     nenu_item]
END SQL INTO menu item
  ARRAY SIZE I
```

1	the database (tel	where
IF (number instances OF navigate menu > 0) THEN	xt format)	PREDECESSOR =
BEGIN	current_time := NOW	:A_change_data_PREDECESSOR_A\
•	#year := YEAR(current_time)	ND CHANGE = :domain.first alternative value
RESIZE menus OF responding window:	#month := MONTH(current_time)	END SOL
number instances OF navig\	#day := DAY(current time)	CONF(first_alternative_value) := -1
ate menu	#hour := HOUR(current_time)	END
	#minute := MINUTE(current time)	ELSE
· ·	date time OF A change data := CONCAT(BEGIN
FOR (nn := 1 TO number instances OF	TO STRING(#year)	EXEC ISodbe 1 SQL insert into [A-
navigate menu)), 7/7)	CPM-Alternativ
BEGIN -	! to add zero before the number for months	ves! (CASE_ID, PREDECESSOR, CHANGE) values
instance number OF navigate menu := na	less than \	(:A_change_data.CASE_ID,
PROPER SANDAL	t	:A_change_data.PREDECESSOR, :A_change_dat
! **** load the menu items in every menu	IF #month < 10 THEN	ta.CHANGE)
EXEC ISodbe 1 SQL SELECT label from	date_time OF A_change_data :=	END SQL
[AB-menu_kem] WHERE\	CONCAT(date_time OF A\	END
(main_menu_label) = :menu_label	_change_data, "0", TO STRING(#month), "/", TO	END
END SQL INTO	STRING(#day), " ", TO ST\	į t
temp_current_me\	RING(#hour), ":", TO STRING(#minute))	† †
nu_items (label)	ELSE	! **** send the messages and later close the data
RESIZE items OF menu := number instances	date_time OF A_change_data :=	display
OF navigate temp_\	CONCAT(date_time OF A\	FORGET
CUITERE menu items	_change_data, TO STRING(#month), "/", TO	A_attributes_of_monitored_object
FOR (%nnn := 1 TO number instances OF	STRING(#day), * *, TO STRING(\	FORGET
navigate temp_cur\	#hour), ":", TO STRING(#minute))	A_object_ID_of_monitored_object
rent_menu_items)	[1	send message components := TRUE
BEGIN	IF (new case or responding = "responding")	END
instance number OF navigate	THEN	ELSE
temp_current menu_items :-\	BEGIN	BEGIN
%ann	Case ID OF A change data :=	text OF message box 2 := "Please tell why
FIND menu item	CASE_ID OF A_Respondin\	you set the \
WHERE label OF menu item = label OF	g_Deta	recommendation level to ZERO*
	END	put up Of message box 2 := TRUE
temp_current_me\		END
nu_items	1	
FIND END	! **** input the data in [A-change-data] table	END
items (%ann) OF menu := menu item		ELSE
END	IF (CONF(Reason_for_recomm OF	BEGIN
!	A_change_data) < 0) THE\	text OF message box 2 := "Recommendation
menus [na] OF responding window := menu	N	level should be
END	Reason_for_recomm OF A_change_data	between 0 and 4°
t	:= * *	put up OF message box 2 := TRUE
END	EXEC ISodbc 1 SQL insert into [A-change-	END
1	data] (CASE_ID\	END
visible OF Select a project window := FALSE	,CHANGE, object, object_ID, attribute,	ELSE
visible OF responding window := TRUE	Required_man_hours, Difference_in\	BEGIN
1	Build_Cost, Recommendation, Reason_for_recomm,	text OF message box 2 := "Some Data are
new case or responding := "responding"	DURATION, date_time, old\	Missing*
1	_value, new_value) VALUES	put up OF message box 2 := TRUE
! *** end of method	(:A_change_data.CASE_ID,	END
END	:A_change_data.CHANGE, :A_change_data.ob\	1
ATTRIBUTE another change in new case SIMPLE	ject, :A change data object ID,	1**** end of method
ATTRIBUTE close change data display SIMPLE	:A_change_data.attribute, :A_change_data\	END
WHEN CHANGED	Required man hours	ATTRIBUTE check existing data SIMPLE
BEGIN	:A_change_data.Difference_in_Build_Cost,	WHEN CHANGED
1	:A_change_\	BEGIN
IF CONF(Required_man_hours OF	data.Recommendation	1
A_change_data) > -1 AND CONF(Diffe\	:A_change_data.Reason_for_recomm,	! load the current value of the attribute to be
rence_in_Build_Cost OF A_change_data) > -1 AND	:A change data DU\	changed
CONF(Recommendation OF A\	RATION, :A_change_data.date_time,	! === check the attribute type first ("n" for numeric or
_change_data) > -1 AND CONF(DURATION OF	th channe date old value at channe del	
_change_data)>-I AND CONF(DURATION OF A_change_data)>-I AND CONF(n\	:A_change_data_old_value, :A_change_da\	"t" for text) EYEC (Sodhe 1 SO), select exclusive time from
	ta.new_value)	EXEC ISodbc 1 SQL select attribute_type from
ew_value OF A_change_data) > -1 THEN	END SQL	[AB-attribute-designe\
BEGIN	1 9940 inner the days in fa CODE Atomostic Dayler	r] where building_element = :A_change_data.object
IF (0 <= Recommendation OF A_change_data	! **** input the data in [A-CPM-Alternatives] table	and attribute_of_build\
AND Recommendation OF\ A_change_data <= 4) THEN	IF (new case or responding = "NEW") THEN	ing_element = :A_change_data.attribute
		END SQL INTO domain (Sttt)
BEGIN		general_statement_of_selection := CONCAT(
	BEGIN	
IF NOT (Recommendation OF A_change_data	PREDECESSOR OF A_change_data :=	"select ", attribute OF \
= 0 AND CONF(Reas)	PREDECESSOR OF A_change_data := "NULL"	"select ", attribute OF \ A_change_data, " from ", object OF A_change_data, "
= 0 AND CONF(Reast on_for_recomm OF A_change_data) = -1) THEN	PREDECESSOR OF A_change_data := "NULL" EXEC ISodbc 1 SQL insert into [A-CPM-	"select ", attribute OF \ A_change_data, " from ", object OF A_change_data, " where ", object OF A\
= 0 AND CONF(Reas)	PREDECESSOR OF A_change_data := "NULL" EXEC ISodbc SQL insert into [A-CPM-Alternatives]\	"sclect ", attribute OF \\ A_change_data, " from ", object OF A_change_data, " where ", object OF A\\ _change_data, " id = : A_change_data.object_ID ")
= 0 AND CONF(Reas\ on_for_recomm OF A_change_data) = -1) THEN BEGIN !	PREDECESSOR OF A_change_data := "NULL" EXEC ISodbe I SQL insert into [A-CPM-Alternatives]\ (CASE_ID, PREDECESSOR, CHANGE) values	"sclect ", stribute OF \ A_change_data, " from ", object OF A_change_data, " where ", object OF A\ change_data, " id = :A_change_data_object_ID ") IF Sttt = "\" THEN
= 0 AND CONF(Reas\ on_for_recomm OF A_change_data) = -1) THEN BEGIN ! ! **** define the CHANGE OF A_change_data	PREDECESSOR OF A_change_data := "NULL" EXEC ISodbe 1 SQL insert into [A-CPM-Alternatives]\ (CASE_ID, PREDECESSOR, CHANGE) values (:A_change_data.CASE_ID,	"sclect", attribute OF\ A_change_data, " from ", object OF A_change_data, " where ", object OF A\ change_data, " id = :A_change_data.object_ID ") IF Sm = "t" THEN BEGIN
= 0 AND CONF(Reas\ on_for_recomm OF A_change_data) = -1) THEN BEGIN ! ! **** define the CHANGE OF A_change_data (combine the *discipline lette\	PREDECESSOR OF A_change_data := "NULL" EXEC ISodbe 1 SQL insert into [A-CPM-Alternatives]\ (CASE_ID, PREDECESSOR, CHANGE) values (:A_change_data_CASE_ID, :A_change_data_PREDECESSOR, :A_change_da\	"sclect ", attribute OF \(\) A_change_data, " from ", object OF A_change_data, " where ", object OF A\\ change_data, " id = : A_change_data.object_ID "\) IF 3m = "\) THEN BEGIN EXEC ISodbe SQL WITH
= 0 AND CONF(Reas) on_for_recomm OF A_change_data) = -1) THEN BEGIN ! ! **** define the CHANGE OF A_change_data (combine the "discipline lette) r" with "ID of A-change-data"	PREDECESSOR OF A_change_data := "NULL" EXEC ISodbe 1 SQL insert into [A-CPM-Alternatives]\ (CASE_ID, PREDECESSOR, CHANGE) values ('A_change_data_CASE_ID, 'A_change_data_PREDECESSOR, :A_change_da\ ta_CHANGE)	"sclect ", stribute OF\ A_change_data, " from ", object OF A_change_data, " where ", object OF A\ change_data, " id = :A_change_data.object_ID ") IF \$tt = "t" THEN BEGIN EXEC ISodbe SQL WITH general_statement_of_selection END SQL \
= 0 AND CONF(Reas\ on_for_recomm OF A_change_data) = -1) THEN BEGIN ! ! **** define the CHANGE OF A_change_data (combine the "discipline lette\ r" with "ID of A-change-data" EXEC 150dbc 1 SQL select MAX(ID) from	PREDECESSOR OF A_change_data := "NULL" EXEC ISodbc 1 SQL insert into [A-CPM-Alternatives]\ (CASE_ID, PREDECESSOR, CHANGE) values (:A_change_data_CASE_ID, :A_change_data_PREDECESSOR, :A_change_da\ ta.CHANGE) END SQL	"sclect", stribute OF\ A_change_data, " from ", object OF A_change_data, " where ", object OF A\ change_data, " id = :A_change_data.object_ID ") IF Sttt = "1" THEN BEGIN EXEC ISodbe SQL WITH general_statement_of_selection END SQL \ INTO domain (tt)
= 0 AND CONF(Reas\ on_for_recomm OF A_change_data) = -1) THEN BEGIN ! ! **** define the CHANGE OF A_change_data (combine the "discipline lette\ r" with "ID of A-change-data" EXEC 15odbc 1 SQL select MAX(ID) from [A-change-data]	PREDECESSOR OF A_change_data:= "NULL" EXEC ISodbc 1 SQL insert into [A-CPM-Alternatives]\ (CASE_ID, PREDECESSOR, CHANGE) values (:A_change_data_CASE_ID, :A_change_data_PREDECESSOR, :A_change_da\ ta_CHANGE) END SQL END	"select ", attribute OF \(\) A_change_data, " from ", object OF A_change_data, " where ", object OF A\\ change_data, " id = : A_change_data.object_ID "\) IF Sm = "\(\) THEN BEGIN EXEC ISodbe SQL WITH general_statement_of_selection END SQL \ INTO domain (t) old_value OF A_change_data := (t)
= 0 AND CONF(Reas) on_for_recomm OF A_change_data) = -1) THEN BEGIN ! ! **** define the CHANGE OF A_change_data (combine the "discipline lette) r" with "ID of A-change-data" EXEC 15odbc 1 SQL select MAX(ID) from [A-change-data] END SQL INTO domain (nn)	PREDECESSOR OF A_change_data := "NULL" EXEC ISodbe 1 SQL insert into [A-CPM-Alternatives]\ (CASE_ID, PREDECESSOR, CHANGE) values (:A_change_data_CASE_ID, :A_change_data_PREDECESSOR, :A_change_da\ ta_CHANGE) END SQL END	"sclect ", stribute OF\" A_change_data, " from ", object OF A_change_data, " where ", object OF A\" change_data, " id = : A_change_data.object_ID ") IF Stn = "t" THEN BEGIN EXEC ISodbc SQL WITH general_statement_of_selection END SQL \ INTO domain (tt) old_value OF A_change_data := (tt) END
= 0 AND CONF(Reas\ on_for_recomm OF A_change_data) = -1) THEN BEGIN ! ! **** define the CHANGE OF A_change_data (combine the "discipline lette\ r" with "ID of A-change-data" EXEC 15odbc 1 SQL select MAX(ID) from [A-change-data] END SQL INTO domain (nn) nn :** nn + 1	PREDECESSOR OF A_change_data := "NULL" EXEC ISodbc 1 SQL insert into [A-CPM-Alternatives]\ (CASE_ID, PREDECESSOR, CHANGE) values (:A_change_data_CASE_ID, :A_change_data_PREDECESSOR, :A_change_da\ ta_CHANGE) END SQL END [If (new case or responding = "responding")	"select ", stribute OF\ A_change_data, " from ", object OF A_change_data, " where ", object OF A\ _change_data, " id = :A_change_data.object_ID ") IF \$ttt = "t" THEN BEGIN EXEC ISodbe SQL WITH general_statement_of_selection END SQL \ INTO domain (tt) old_value OF A_change_data := (tt) END ELSE
= 0 AND CONF(Reas\ on_for_recomm OF A_change_data) = -1) THEN BEGIN ! ! **** define the CHANGE OF A_change_data (combine the "discipline lette\ r" with "ID of A-change-data" EXEC ISodbe 1 SQL select MAX(ID) from [A-change-data] END SQL INTO domain (nn) nn := nn + 1 EXEC ISodbe 1 SQL select	PREDECESSOR OF A_change_data := "NULL" EXEC ISodbc 1 SQL insert into [A-CPM-Alternatives]\ (CASE_ID, PREDECESSOR, CHANGE) values (:A_change_data_CASE_ID, :A_change_data_PREDECESSOR, :A_change_da\ ta.CHANGE) END SQL END IF (new case or responding = "responding") THEN	"select ", attribute OF \(\) A_change_data, "from ", object OF A_change_data, " where ", object OF A\\ change_data, " id = : A_change_data.object_ID "\) IF Sm = "\(\) THEN BEGIN EXEC ISodbe SQL WITH general_statement_of_selection END SQL \ INTO domain (t) old_value OF A_change_data := (\(\) END ELSE BEGIN
= 0 AND CONF(Reas\ on_for_recomm OF A_change_data) = -1) THEN BEGIN ! ! **** define the CHANGE OF A_change_data (combine the "discipline lette\ r" with "ID of A-change-data" EXEC 15odbc 1 SQL select MAX(ID) from [A-change-data] END SQL INTO domain (nn) nn :** nn + 1	PREDECESSOR OF A_change_data := "NULL" EXEC ISodbc 1 SQL insert into [A-CPM-Alternatives]\ (CASE_ID, PREDECESSOR, CHANGE) values (:A_change_data_CASE_ID, :A_change_data_PREDECESSOR, :A_change_da\ ta_CHANGE) END SQL END [If (new case or responding = "responding")	"select ", stribute OF\ A_change_data, " from ", object OF A_change_data, " where ", object OF A\ _change_data, " id = :A_change_data.object_ID ") IF \$ttt = "t" THEN BEGIN EXEC ISodbe SQL WITH general_statement_of_selection END SQL \ INTO domain (tt) old_value OF A_change_data := (tt) END ELSE
= 0 AND CONF(Reas\ on_for_recomm OF A_change_data) = -1) THEN BEGIN ! ! **** define the CHANGE OF A_change_data (combine the "discipline lette\ r" with "ID of A-change-data" EXEC ISodbe 1 SQL select MAX(ID) from [A-change-data] END SQL INTO domain (nn) nn := nn + 1 EXEC ISodbe 1 SQL select	PREDECESSOR OF A_change_data := "NULL" EXEC ISodbc 1 SQL insert into [A-CPM-Alternatives]\ (CASE_ID, PREDECESSOR, CHANGE) values (:A_change_data_CASE_ID, :A_change_data_PREDECESSOR, :A_change_da\ ta.CHANGE) END SQL END IF (new case or responding = "responding") THEN	"select ", attribute OF \(\) A_change_data, "from ", object OF A_change_data, " where ", object OF A\\ change_data, " id = : A_change_data.object_ID "\) IF Sm = "\(\) THEN BEGIN EXEC ISodbe SQL WITH general_statement_of_selection END SQL \ INTO domain (t) old_value OF A_change_data := (\(\) END ELSE BEGIN
= 0 AND CONF(Reas) on_for_recomm OF A_change_data) = -1) THEN BEGIN ! ! **** define the CHANGE OF A_change_data (combine the "discipline lette\" with "ID of A-change-data" EXEC Isodbe I SQL select MAX(ID) from [A-change-data] END SQL INTO domain (nn) nn :** nn + 1 EXEC Isodbe I SQL select designing_discipline from [AB\"	PREDECESSOR OF A_change_data := "NULL" EXEC ISodbe 1 SQL insert into [A-CPM-Alternatives]. (CASE_ID, PREDECESSOR, CHANGE) values (:A_change_data_CASE_ID, :A_change_data_PREDECESSOR, :A_change_dat ta_CHANGE) END SQL END IF (new case or responding = "responding") THEN BEGIN	"sclect ", attribute OF \(\) A_change_data, " from ", object OF A_change_data, " where ", object OF A\) change_data, " id = : A_change_data.object_ID ") IF Stt = 't" THEN BEGIN EXEC ISodbe SQL WITH general_statement_of_selection END SQL \ INTO domain (tt) old_value OF A_change_data := (tt) END ELSE BEGIN EXEC ISodbe SQL WITH
= 0 AND CONF(Reas\ on_for_recomm OF A_change_data) = -1) THEN BEGIN ! **** define the CHANGE OF A_change_data (combine the "discipline lette\ r" with "ID of A-change-data"	PREDECESSOR OF A_change_data := "NULL" EXEC ISodbe 1 SQL insert into [A-CPM-Alternatives]\ (CASE_ID, PREDECESSOR, CHANGE) values ('A_change_data_CASE_ID, 'A_change_data_PREDECESSOR, :A_change_da\ ta_CHANGE) END SQL END IF (new case or responding = "responding") THEN BEGIN PREDECESSOR OF A_change_data :=	"select ", attribute OF \(\) A_change_data, "from ", object OF A_change_data," where ", object OF A\) change_data, " id = : A_change_data.object_ID ") IF Sm = "\) "THEN BEGIN EXEC 15odbe SQL WITH general_statement_of_selection END SQL \ INTO domain (n) old_value OF A_change_data := (n) END ELSE BEGIN EXEC 15odbe SQL WITH general_statement_of_selection END SQL \ INTO domain (n)
= 0 AND CONF(Reas\ on_for_recomm OF A_change_data) = -1) THEN BEGIN ! ! **** define the CHANGE OF A_change_data (combine the "discipline lette\ r" with "ID of A-change-data" EXEC ISodbe 1 SQL select MAX(ID) from [A-change-data] END SQL INTO domain (nn) nn := nn + 1 EXEC ISodbe 1 SQL select designing_discipline from [AB\ -amribute-designer] where building_element = :A_change_data_object AND \	PREDECESSOR OF A_change_data := "NULL" EXEC ISodbc 1 SQL insert into [A-CPM-Alternatives]\ (CASE_ID, PREDECESSOR, CHANGE) values (:A_change_data_CASE_ID, :A_change_data_PREDECESSOR, :A_change_da\ ta.CHANGE) END SQL END IF (new case or responding = "responding") THEN BEGIN PREDECESSOR OF A_change_data := PREDECESSOR OF A_change_data :=	"sclect ", stribute OF\ A_change_data, " from ", object OF A_change_data, " where ", object OF A\ _change_data, " id = :A_change_data.object_ID ") IF \$tt = 't" THEN BEGIN EXEC !Sodbe ! SQL WITH general_statement_of_selection END SQL \ INTO domain (t) old_value OF A_change_data := (tt) END ELSE BEGIN EXEC !Sodbe ! SQL WITH general_statement_of_selection END SQL \
= 0 AND CONF(Reas) on_for_recomm OF A_change_data) = -1) THEN BEGIN ! ! **** define the CHANGE OF A_change_data (combine the "discipline lette\" with "ID of A-change-data" EXEC Isodbe I SQL select MAX(ID) from [A-change-data] END SQL INTO domain (nn) mn:** nn + 1 EXEC Isodbe I SQL select designing_discipline from [AB\" - arribute-designer] where building_element = A_change_data_object AND \ attribute_of_building_element =	PREDECESSOR OF A_change_data := "NULL" EXEC ISodbe 1 SQL insert into [A-CPM-Alternatives]. (CASE_ID, PREDECESSOR, CHANGE) values (:A_change_data_CASE_ID, :A_change_data_PREDECESSOR, :A_change_da\ ta_CHANGE) END SQL END IF (new case or responding = "responding") THEN BEGIN PREDECESSOR OF A_change_data := PREDECESSOR OF A_R\ esponding_Data	"select ", attribute OF \(A_\) change_data, " from ", object OF A_\) change_data, " from ", object OF A_\) change_data, " id =: A_\) change_data.object_ID ") IF Stn = "\) THEN BEGIN EXEC ISodbe SQL WITH general_statement_of_selection END SQL \ INTO domain (tt) old_value OF A_\) change_data := (tt) END ELSE BEGIN EXEC ISodbe SQL WITH general_statement_of_selection END SQL \ INTO domain (tt) old_value OF A_\) change_data := TO STRING(
= 0 AND CONF(Reas\ on_for_recomm OF A_change_data) = -1) THEN BEGIN ! """ define the CHANGE OF A_change_data (combine the "discipline lette\ r" with "ID of A-change-data"	PREDECESSOR OF A_change_data := "NULL" EXEC ISodbe 1 SQL insert into [A-CPM-Alternatives]\ (CASE_ID, PREDECESSOR, CHANGE) values ('A_change_data_CASE_ID, 'A_change_data_PREDECESSOR, :A_change_da\ ta_CHANGE) END SQL END ! If (new case or responding = "responding") THEN BEGIN PREDECESSOR OF A_change_data := PREDECESSOR OF A_R\ esponding_Data IF (yes for second alternative = "YES")	"sclect ", attribute OF\ A_change_data, " from ", object OF A_change_data, " where ", object OF A\ _change_data, " id = : A_change_data.object_ID ") IF Stt = "t" THEN BEGIN EXEC ISodbe SQL WITH general_statement_of_selection END SQL \ INTO domain (tt) old value OF A_change_data := (tt) END ELSE BEGIN EXEC ISodbe SQL WITH general_statement_of_selection END SQL \ INTO domain (m) Old_value OF A_change_data := TO STRING(nn)
= 0 AND CONF(Reas\ on_for_recomm OF A_change_data) = -1) THEN BEGIN ! ! **** define the CHANGE OF A_change_data (combine the "discipline lette\ r" with "ID of A-change-data" EXEC 15odbc 1 SQL select MAX(ID) from [A-change-data] END SQL INTO domain (nn) nn := nn + 1 EXEC 15odbc 1 SQL select designing_discipline from [AB\ -amribute-designer] where building_element = :A_change_data_object AND \ attribute_of_building_element = :A_change_data_stribute END SQL INTO domain (tt)	PREDECESSOR OF A_change_data := "NULL" EXEC ISodbc 1 SQL insert into [A-CPM-Alternatives]\ (CASE_ID, PREDECESSOR, CHANGE) values (:A_change_data_CASE_ID, :A_change_data_PREDECESSOR, :A_change_da\ ta.CHANGE) END SQL END IF (new case or responding = "responding") THEN BEGIN PREDECESSOR OF A_change_data := PREDECESSOR OF A_R\ esponding_Data IF (yes for second alternative = "YES") THEN	"select ", attribute OF \(A_\) change_data, " from ", object OF A_\) change_data, " from ", object OF A_\) change_data, " id = : A_\) change_data.object_ID ") IF Sm = "\) THEN BEGIN EXEC 15odbe SQL WITH general_statement_of_selection END SQL \ INTO domain (n) old_value OF A_\) change_data := (n) END ELSE BEGIN EXEC 15odbe SQL WITH general_statement_of_selection END SQL \ INTO domain (m) old_value OF A_\) change_data := TO STRING(nn) END
= 0 AND CONF(Ress) on_for_recomm OF A_change_data) = -1) THEN BEGIN ! ! **** define the CHANGE OF A_change_data (combine the "discipline lette\" with "ID of A-change-data" EXEC ISodbe I SQL select MAX(ID) from [A-change-data] END SQL INTO domain (nn) nn := nn + 1 EXEC ISodbe I SQL select designing_discipline from [AB\" attribute-designer] where building_element = :A_change_data_stribute END SQL INTO domain (nt) attribute_of_building_element = :A_change_data_stribute END SQL INTO domain (nt) CHANGE OF A_change_data :=	PREDECESSOR OF A_change_data := "NULL" EXEC ISodbe 1 SQL insert into [A-CPM-Alternatives]. (CASE_ID, PREDECESSOR, CHANGE) values (:A_change_data_CASE_ID, :A_change_data_PREDECESSOR, :A_change_da\ ta_CHANGE) END SQL END IF (new case or responding = "responding") THEN BEGIN PREDECESSOR OF A_change_data := PREDECESSOR OF A_R\ esponding_Data IF (yes for second alternative = "YES") THEN BEGIN	"select ", attribute OF \(\) A_change_data, " from ", object OF A_change_data, " where ", object OF A\\ change_data, " id = : A_change_data.object_ID "\) IF Sm = "\" THEN BEGIN EXEC 15odbe SQL WITH general_statement_of_selection END SQL \ INTO domain (n) old_value OF A_change_data := (n) END ELSE BEGIN EXEC 15odbe SQL WITH general_statement_of_selection END SQL \ INTO domain (m) old_value OF A_change_data := TO STRING(nn) END

! — if it has been made before retrieve the previously	%nnaNew := TO NUMERIC(new_value OF	A_Violated_SPECIFIC_rule
used data EVEC (Souther 1 SOI) select 8 from (A shapes data)	A_change_data)	FOR the LTO ment of the OF andre
EXEC ISodbc 1 SQL select * from [A-change-data] WHERE object = :A_\	FOR (nn := 1 TO number instances OF navigate	FOR (nn := 1 TO number instances OF naviga A_Violated_SPECIFI\
change_data.object AND object_ID =	A_Violated_GENERAL\	C_rule)
:A_change_data.object_ID AND attribut\	_rule)	BEGIN
e = :A_change_data.attribute END SOL INTO	BEGIN instance number OF navigate	instance number OF navigate A_Violated_SPECIFIC_rule := nn
A_change_data_retrieved	A Violated GENERAL rule := nn	A_Volume_st zen te_tune := un
1	%onn := percent_change OF	! get the affected discipline by the change
IF number instances OF navigate retrieved data > 0 THEN	A_Violated_GENERAL_rule IF (%nnn = 0) OR (%nnnOid * (1 - %nnn) >	EXEC ISodbc 1 SQL select designing_discipling from [AB-attribut]
BEGIN	%annNew) OR (%annOld \	te-designer] where building_element =
1	" (1 + %ann) < %annNew) THEN	:A_Violated_SPECIFIC_rule.building\
output OF existing design data window := existing data for a d\	BEGIN MAKE A_message_components	element and attribute_of_building_element == :A_Violated_SPECIFIC_rule.a\
esign change display	WITH changed_attribute := attribute OF	ttribute_of_building_element_END SQL INTO dom
visible OF existing design data window := TRUE	A_change_data	(Strt)
! END	WITH changed_class := object OF A_change_data	affected_discipline OF A_Violated_SPECIFIC_rule := Sttt
ELSE	WITH changed_instance id := object ID OF	1
BEGIN	A_change_data	get the value of the affected attribute
! ! ***** if there is no previous similar change then ask	WITH old_value := old_value OF A_change_data	! check the attribute type first ("n" for numeric ("t" for text)
for the change d\	WITH new_value := new_value OF	EXEC ISodbc 1 SQL select attribute_type from
ata	A_change_data	(AB-attribute-des/
! visible OF design data window := TRUE	WITH writting_date := date_time OF A_change_data	igner] where building_element = :A_Violated_SPECIFIC_rule.building_eleme\
output OF design data window := data for a	WITH discipline_to := affected_discipline	nt and attribute of building element =
design change dispN	OF A_Violated\	:A_Violated_SPECIFIC_rule.attribu\
ay END	_GENERAL_rule WITH discipline_from := \$discipline_from	te_of_building_element END SQL INTO domain (Sm)
!	WITH sent_because_change_ID :=	(July
! ***** end of method	CHANGE OF A_change_data	statement_of_selection OF
END ATTRIBUTE tt STRING	WITH sent_because_change_ID :=	A_Violated_SPECIFIC_rule := CONCAT(\ "Select", building_element OF
ATTRIBUTE new case or responding STRING	CHANGE OF A_change_data WITH Case_ID := Case_ID OF	A Violated SPECIFIC rule, ".", attribute \
ATTRIBUTE Case_ID NUMERIC	A_change_data	of building_element OF A_Violated_SPECIFIC_rul
ATTRIBUTE current_time TIME ATTRIBUTE #year NUMERIC	WITH general_specific := "GENERAL" WITH cause of the effect :=	" from ", building_elem\ ent OF A_Violated_SPECIFIC_rule, ", ", related_cla
ATTRIBUTE #month NUMERIC	cause of the effect OF A_Vi\	OF A_Violated_SPECI\
ATTRIBUTE #day NUMERIC	olated_GENERAL_rule	FIC_rule, " where (", building_element OF
ATTRIBUTE #hour NUMERIC ATTRIBUTE #minute NUMERIC	EXEC Isodbe ! SQL insert into [A_virtual_message_elements]\	A_Violated_SPECIFIC_rule, "."\ , building_element OF A_Violated_SPECIFIC_rule,
ATTRIBUTE Sut STRING	(changed_attribute,	_id = ", building_elem\
ATTRIBUTE cancel current change SIMPLE	changed_class, changed_instance_id, old_value,	ent_id OF A_Violated_SPECIFIC_rule, ") AND (",
WHEN CHANGED BEGIN	new_value, writting_d\ ate, discipline_to,	related_class OF A_Viola\ ted_SPECIFIC_rule, ".", related_class OF
visible OF design data window := FALSE	discipline_from, sent_because_change_ID,	A_Violated_SPECIFIC_rule, *_id \
FORGET output OF design data window	general specific, Case ID, \	= ", related_instance_id OF
visible OF existing design data window := FALSE FORGET output OF existing design data window	cause_of_the_effect) values (:A_message_components.changed_attribute,	A_Violated_SPECIFIC_rule, ") AND (") statement_of_selection OF
FORGET A_attributes_of_monitored_object	:A_message_components.cha\	A_Violated_SPECIFIC_rule := CONCAT(\
FORGET A_object_ID_of_monitored_object	nged_class,	statement_of_selection OF
FORGET A_change_data END	:A_message_components.changed_instance_id, :A_message_components.old\	A_Violated_SPECIFIC_rule, relationship OF A_Vi\ olated_SPECIFIC_rule, *)*)
ATTRIBUTE send message components SIMPLE	_value,	!!
WHEN CHANGED BEGIN	:A_message_components.new_value,	FORGET Saffected_attribute_value
I	:A_message_components.writting_date\ .	FORGET %affected_attribute_value
! ***** get the discipline who is responsible for the	:A_message_components.discipline_to,	IF Stit = "t" THEN
changed attribute\ (from the database)	:A_message_components.disciplin\ e_from,	BEGIN EXEC Bodbe 1 SQL WITH
EXEC ISodbc 1 SQL select designing_discipline	:A_message_components.sent_because_change_ID,	statement_of_selection OF A_Violate\
from (AB-attribute-d\	:A_message_components.\	d_SPECIFIC_rule END SQL INTO domain
esigner) where building_element = :A_change_data.object and attribute_of(general_specific, :A_message_components.Case_ID, :A_message_components.cause_of_the_effect	(\$affected_attribute_value) END
_building_element = :A_change_data.attribute)	ELSE
END SQL INTO domain (\$discipline from)	END SQL	BEGIN
(Jaiscipine_irom)	END END	EXEC ISodbc 1 SQL WITH statement_of_selection OF A_Violate\
!	1	d_SPECIFIC_rule END SQL INTO domain
 1	 1	(%affected_attribute_value)
!	!	%nnnNew := TO NUMERIC(new_value O A_change_data)
!(1)	I—(2)————	END
\.		
! ***** to check if there is any violation of General	! ***** to check if there is any violation of specific	! if there is a violation then send a message
rules and send mel	rules and send m\	IF ((CONF(\$affected_attribute_value) < 0)
ssage if there is EXEC ISodbc 1 SQL select * from [AB-general-	essage if there is ! ==== note: the changed attribute is considered the	AND (CONF(%affecte\ d_attribute_value) < 0)) THEN
rules] where building_\	related attribute at	BEGIN
element = :A_change_data.object AND	t the database table	MAKE A_message_components
attribute_of_building_element = :A_c\ hange_data.attribute_AND rule_ignored =	EXEC ISodbe 1 SQL select * from [AB-specific- rules] where related_\	WITH changed_attribute := attribute OF A_change_data
:domain.\$No END SQL INTO A_Viol\	class = :A_change_data.object AND	WITH changed_class := object OF
ated_GENERAL_rule	related instance id = :A change_data.o\	A_change_data
%nnnOld := TO NUMERIC(old_value OF	bject_ID AND related_attribute = 'A_change_data.attribute_AND rule_igno\	WITH changed_instance_id := object_ID A_change_data
A_change_data)	red = :domain.SNo END SOL INTO	WITH old value := old value OF

A_change_data	visible OF design data window := FALSE	The possible cause of the effect is ",
WITH new_value := new_value OF A_change_data	FORGET output OF design data window [**** end of method	cause_of_the_effect OF disciplin\ e_message, "
WITH affected_attribute := attribute_of_building_elemen\	END ATTRIBUTE %nnn NUMERIC	Please indicate the expected modifications in the
t OF A Violated SPECIFIC rule WITH affected class: building element	ATTRIBUTE %conOld NUMERIC ATTRIBUTE %conNew NUMERIC	building components. "\
OF A_Violated_S\	ATTRIBUTE Sdiscipline_from STRING	1
PECIFIC_rule WITH affected_instance_id :=	ATTRIBUTE Saffected_attribute_value STRING ATTRIBUTE %affected_attribute_value NUMERIC	attachment OF valuebox 15 := message builder OF required_message
building element id OF A_V\ iolated_SPECIFIC_rule	ATTRIBUTE %yyy NUMERIC ATTRIBUTE close existing data display SIMPLE	END
WITH relationship := relationship OF	WHEN CHANGED	ATTRIBUTE specific message SIMPLE
A_Violated_SPECIFI\ C_rule	BEGIN Required_man_hours OF A_change_data :=	WHEN CHANGED BEGIN
WITH writting date := date time OF A_change_data	Required man_hours OF A_che\ age_data_retrieved	message builder OF required_message :=
WITH discipline_to := affected_discipline	DURATION OF A_change_data := DURATION	CONCAT("In VIRTUAL MODE, TO he attribute ", changed_attribute OF
OF A_Violated\ _SPECIFIC_rule	OF A_change_data_retrieved Difference_in_Build_Cost OF A_change_data :=	discipline_message, " of the buil\ ding component ", changed_class OF
WITH discipline_from := Sdiscipline_from	Difference_in_Build_C\	discipline_message, " (", changed_in\
WITH sent_because_change_ID := CHANGE OF A_change_data	ost OF A_change_data_retrieved Recommendation OF A_change_data :=	stance_id OF discipline_message, ") has been changed from the value (", old_value OF
WITH general_specific := "SPECIFIC" WITH rule_writting_date := writting_date	Recommendation OF A_change_data\	discipline_message, ")\ to the value (", new_value OF discipline_message, ").
OF A_Violated_\	Reason_for_recomm OF A_change_data :=	According to a previously define SPECIFIC rule, such
SPECIFIC_rule WITH violated_rule_id :=	Reason_for_recomm OF A_chang\ e_data_retrieved	a change may affect\ the design of the attribute
AB_specific_rules_id OF A_Vio/\	visible OF design data window := TRUE	", affected_attribute OF discipline_message, " of the
ated_SPECIFIC_rule WITH Case_ID := Case_ID OF	output OF design data window := data for a design change display	building compost\ mt ", affected_class OF discipline_message, " (",
A_change_data EXEC ISodbc 1 SQL insert into	visible OF existing design data window := FALSE FORGET output OF existing design data window	affected_instance_id\ OF discipline_message, "Y.
[A_virtual_message_elements]\	END	Please indicate the expected modifications to the
(changed_attribute, changed_class, changed_instance_id, old_value,	ATTRIBUTE second alternative SIMPLE WHEN CHANGED	affected building comp\ onents.")
new_value, affected_a\ ttribute, affected_class,	BEGIN	I market of State and State of
affected_instance_id, relationship, writting_date,	IF (CONF(first_alternative_value) >= 0) THEN	attachment OF valuebox 15 := message builder OF required_message
discipline to, di\ scipline from	BEGIN put up OF message box 3 := TRUE	END
sent_because_change_ID, general_specific,	!	ATTRIBUTE effect of menu item SIMPLE
rule_writting_date, violat\ ed_rule_id, Case_ID) values	IF button selected OF message box 3 IS yes THEN	WHEN CHANGED BEGIN
(:A_message_components.changed_attribute, :A_message_components.changed_class,	BEGIN put up OF message box 3 := FALSE	object OF A_change_data := label OF menu item
:A_message_components.changed_i\	yes for second alternative := "YES"	complete as monitored object := TRUE
nstance_id, :A_message_components.old_value,	END !	1 1 *** end of method
:A_message_components.new_value, :A_message_components.affected_attribute,	IF button selected OF message box 3 IS no THEN	END ATTRIBUTE project_name STRING
:A_message_components.affe\	BEGIN	WHEN CHANGED
cted_class, :A_message_components.affected_instance_id,	put up OF message box 3 := FALSE END	BEGIN data source OF ISodbc 1 := project_name
:A_message_components.r\ elationship,	END FLSE	title OF new case window := CONCAT("New Case for Project : ", pro\
:A_message_components.writting_date,	BEGIN	ject_name)
:A_message_components.disciplin\ e_to,	text OF message box 1 := "No first alternative was given"	title OF responding window := CONCAT("Responding for Project : ",\
:A_message_components.discipline_from,	put up OF message box 1 := TRUE	project_name)
:A_message_components.sent_b\ ecause_change_ID,	END !	output OF Select a project window := select_new_or_responding_disp\
:A_message_components.general_specific, :A_message_components.rule_writting_date,	!**** end of method END	tay END
:A_message_components.vio\	ATTRIBUTE yes for second alternative STRING	ATTRIBUTE go_ahead SIMPLE
lated_rule_id, :A_message_components.Case_ID)	ATTRIBUTE first_alternative_value STRING ATTRIBUTE SYES STRING	WHEN CHANGED BEGIN
END SQL END	ATTRIBUTE \$No STRING ATTRIBUTE general message SIMPLE	! action OF ISodbe 2 GENERAL IS connect :=
END	WHEN CHANGED	TRUE
\	BEGIN	EXEC ISodbe 2 GENERAL SQL SELECT
1	IF (CONF(cause_of_the_effect OF discipline_message) < 0) THEN	Project_name from Current_project\
! **** to indicate the the original message has been	cause_of_the_effect OF discipline_message :=	END SQL INTO Current_projects
acted upon (in case\ of "responding"	"Undefined"	(Project_name)
IF (new case or responding = "responding") THEN BEGIN	message builder OF required_message :=	visible OF main window := FALSE
tt:="Yes"	CONCAT("In VIRTUAL MODE, T\ he attribute ", changed_attribute OF	visible OF Select a project window := TRUE output OF Select a project window :=
EXEC Isodbe 1 SQL update A_virtual_message_elements set been_a\	discipline_message, " of the buil\ ding component ", changed_class OF	select_project_display
cted_upon = :domain.tt	discipline_message, " (", changed_in\	END
where id = :discipline_message.id END SQL	stance_id OF discipline_message, ")' has been changed from the value (", old_value OF	INSTANCE border 1 ISA border
END .	discipline message, ") \ to the value (", new_value OF discipline_message, ").	WITH style IS group WITH perspective IS in
first_alternative_value := CHANGE OF	Such a change may affect the design of the ",	WITH border width := 1
A_change_data FORGET A_change_data	discipline_to OF discipli\ ne_message, = system.	WITH background color := 192,192,192 WITH fill color := 192,192,192
FORGET A message components		WITH highlight color := 255 255 255

WITH shadow color := 128,128,128 WITH location := 295,60,420,125

INSTANCE border 2 ISA border
WITH style IS group
WITH perspective IS in
WITH border width := 1
WITH background color := 192,192,192
WITH fill color := 192,192,192
WITH highlight color := 255,255,255
WITH shadow color := 128,128,128
WITH location := 10,60,290,125

INSTANCE border 6 ISA border
WITH myle IS shadow
WITH perspective IS in
WITH border width:= 6
WITH beckground color:= 192,192,192
WITH fill color:= 0,0,128
WITH highlight color:= 255,255,255
WITH shadow color:= 0,0,0
WITH location:= 15,15,195,140

INSTANCE border 7 ISA border
WITH style IS picture frame
WITH perspective IS in
WITH border width: = 1
WITH beckground color := 192,192,192
WITH fill color := 192,192,192
WITH shallow color := 255,255,255
WITH shallow color := 128,128,128
WITH location := 425,65,630,165

INSTANCE border 8 ISA border
WITH style IS shadow
WITH perspective IS in
WITH border width:=4
WITH background color:= 192,192,192
WITH fill color:= 192,192,192
WITH highlight color:= 255,255,255
WITH shadow color:= 128,128,128
WITH location:= 20,220,634,372

INSTANCE border 9 ISA border
WITH style IS group
WITH perspective IS in
WITH border width:= 1
WITH border width:= 1
WITH border width:= 192,192,192
WITH fill color:= 192,192,192
WITH highlight color:= 255,255,255
WITH shadow color:= 128,128,128
WITH location:= 5,175,350,400

INSTANCE border 11 ISA border
WITH style IS picture frame
WITH perspective IS in
WITH border width:=1
WITH border width:=1
WITH background color:=192,192,192
WITH fill color:=192,192,192
WITH highlight color:=255,255,255
WITH shadow color:=128,128,128
WITH location:=5,5,395,330

INSTANCE border 12 ISA border
WITH style IS picture frame
WITH perspective IS in
WITH border width:= 1
WITH border width:= 1
WITH blockground color:= 192,192,192
WITH fill color:= 192,192,192
WITH highlight color:= 255,255,255
WITH shadow color:= 128,128,128
WITH location:= 10,10,390,325

INSTANCE border 13 ISA border WITH style IS picture frame WITH perspective IS in WITH border width := 1 WITH background color := 192,192,192 WITH fill color := 192,192,192 WITH highlight color := 255,255,255 WITH shadow color := 128,128,128 WITH location := 5,5,395,375

INSTANCE border 14 ISA border
WITH style IS picture frame
WITH perspective IS in
WITH border width:= 1
WITH background color:= 192,192,192
WITH fill color:= 192,192,192
WITH highlight color:= 255,255,255
WITH shadow color:= 128,128,128

WITH location := 10,180,390,370

INSTANCE border 15 ISA border
WITH style IS group
WITH perspective IS in
WITH border width: = 1
WITH background color := 192,192,192
WITH fill color := 192,192,192
WITH highlight color := 255,255,255
WITH shadow color := 128,128,128
WITH location := 10,130,290,210

INSTANCE border 16 ISA border WITH style IS group WITH perspective IS in WITH border width: = I WITH border width: = I WITH background color := 192,192,192 WITH fill color := 192,192,192 WITH highlight color := 255,255,255 WITH shadow color := 128,128,128 WITH location := 295,130,570,210

INSTANCE border 17 ISA border
WITH style IS picture frame
WITH perspective IS in
WITH border width := [
WITH blockground color := 192,192,192
WITH fill color := 192,192,192
WITH highlight color := 255,255,255
WITH shadow color := 128,128
WITH location := 10,40,390,175

INSTANCE border 18 ISA border
WITH style IS picture frame
WITH perspective IS in
WITH border width := 1
WITH background color := 192,192,192
WITH fill color := 192,192,192
WITH highlight color := 255,255,255
WITH shadow color := 128,128,128
WITH location := 490,180,580,255

INSTANCE border 19 ISA border
WITH style IS edit control
WITH perspective IS is
WITH border width:= 1
WITH border width:= 1
WITH background color:= 192,192,192
WITH fill color:= 192,192,192
WITH highlight color:= 255,255,255
WITH shadow color:= 128,128,128
WITH location:= 5.5,205,215

INSTANCE border 20 ISA border
WITH style IS picture frame
WITH perspective IS in
WITH border width := 1
WITH bockground color := 192,192,192
WITH fill color := 192,192,192
WITH highlight color := 255,255,255
WITH shadow color := 128,128,128
WITH location := 10,10,240,195

INSTANCE border 21 ISA border
WITH style IS shadow
WITH perspective IS in
WITH border width := 6
WITH background color := 192,192,192
WITH fill color := 0,0,128
WITH highlight color := 25,25,255,255
WITH shadow color := 0,0,0
WITH location := 25,20,225,135

INSTANCE border 22 ISA border
WITH style IS picture frame
WITH perspective IS in
WITH border width:= 1
WITH background color:= 192,192,192
WITH fill color:= 192,192,192
WITH highlight color:= 255,255,255
WITH shadow color:= 128,128,128
WITH location:= 10,10,240,195

INSTANCE border 23 ISA border
WITH style IS shadow
WITH perspective IS in
WITH border width: = 6
WITH border width: = 6
WITH fill color: = 192,192,192
WITH fill color: = 0,0,128
WITH highlight color: = 255,255,255
WITH shadow color: = 0,0,0
WITH location: = 2,252,320,130

INSTANCE combobox 1 ISA combobox WITH enabled := TRUE WITH sort := FALSE WITH update with current instance := TRUE WITH location := 155,245,295,265

INSTANCE the domain ISA domain WITH \$Yes := "Yes" WITH \$No := "No"

INSTANCE ISodbe t ISA ISodbe
WITH data source: "build"
WITH connection prompting: "FALSE
WITH auto commit: "TRUE
WITH append: "FALSE
WITH default error handling: "TRUE

INSTANCE ISodbe 2 GENERAL ISA ISodbe WITH data source := "general"
WITH connection prompting := FALSE WITH auto commit := TRUE
WITH append := FALSE
WITH default error handling := TRUE

INSTANCE message box I ISA message box WITH title := "INPUT ERROR" WITH text: = "You need to input query data" WITH modal style IS app modal WITH icon IS stop sign WITH button IS ok WITH default button IS one

INSTANCE message box 2 ISA message box WITH title := "INPUT ERROR" WITH modal style IS app modal WITH icon IS stop sign WITH button IS ok WITH default button IS one

INSTANCE message box 3 ISA message box
WITH text: "You are creating a second alternative
change
Would you like to do so?"
WITH modal style IS app modal
WITH to IS information

WITH modal style IS app modal WITH icon IS information WITH button IS yes no WITH default button IS one

INSTANCE navigate 1 ISA navigate
WITH class attachment := discipline_message

INSTANCE navigate retrieved data ISA navigate
WITH class attachment := A_change_data_retrieved

INSTANCE navigate A_Violated_GENERAL_rule ISA navigate WITH class attachment := A_Violated_GENERAL_rule

INSTANCE navigate A_Violated_SPECIFIC_rule ISA navigate
WITH class attachment :=
A_Violated_SPECIFIC_rule

INSTANCE navigate temp_current_menu_items ISA navigate
WITH class attachment := temp_current_menu_items

INSTANCE navigate menu ISA navigate WITH class attachment := menu

INSTANCE the application ISA application WITH unknowns fail: = TRUE WITH threshold: = 50 WITH title display: = Starting display WITH ignore breakpoints: = FALSE WITH reasoning on: = FALSE WITH numeric precision: = 8 WITH demon strategy IS fire first

INSTANCE Starting display ISA display WITH wait: = TRUE
WITH delay changes: = FALSE
WITH items [1]: = border 19
WITH items [2]: = border 6
WITH items [3]: = textbox 1
WITH items [4]: = UNDETERMINED
WITH items [5]: = UNDETERMINED
WITH items [6]: = UNDETERMINED
WITH items [6]: = UNDETERMINED
WITH items [7]: = pushbutton 25

```
WITH fill color := 192,192,192
WITH menus [1] := UNDETERMINED
                                                                                                                                        INSTANCE data for a design change display ISA
                                                                                                                                                                                                                                                                                WITH items [2]:= border 21
                                                                                                                                                                                                                                                                               WITH items [3]:= textbox 62
WITH items [4]:= listbox 5
WITH fill color:= 192,192,192
                                                                                                                                        display
WITH wait := TRUE
 INSTANCE New Case display ISA display
                                                                                                                                           WITH delay changes := FALSE
     WITH wait := TRUE
                                                                                                                                          WITH items [1] := border 11
WITH items [2] := border 12
                                                                                                                                                                                                                                                                                WITH menus [1] := UNDETERMINED
   WITH wait: "TRUE
WITH delay changes: "FALSE
WITH items [1]: "border 16
WITH items [2]: "border 15
WITH items [3]: "UNDETERMINED
WITH items [4]: "border 2
WITH items [5]: "textbox 2
WITH items [6]: "border 1
WITH items [7]: In weather 4
                                                                                                                                          WITH items [3] := textbox 40
WITH items [4] := textbox 41
WITH items [5] := textbox 42
WITH items [6] := textbox 43
                                                                                                                                                                                                                                                                             INSTANCE select_new_or responding display ISA
                                                                                                                                                                                                                                                                            display
WITH wait := FALSE
                                                                                                                                                                                                                                                                             WITH wait := FALSE
WITH delay changes := FALSE
WITH items [1] := border 22
WITH items [2] := border 23
WITH items [2] := tenthos 63
WITH items [4] := pushbutton 6
WITH items [5] := pushbutton 12
WITH fill color := 192,192,192
WITH man [1] := IUDESTENDER
                                                                                                                                         WITH items [7] := textbox 44
WITH items [8] := textbox 45
WITH items [8] := promptbox 3
WITH items [10] := promptbox 4
WITH items [11] := promptbox 6
WITH items [12] := UNDETERMINED
WITH items [13] := UNDETERMINED
    WITH items [7]:= textbox 4
   WITH items [8] := textbox 5
WITH items [9] := textbox 6
    WITH items [10] := valuebox 3
 WITH items [10] := valuebox 3
WITH items [11] := textbox 7
WITH items [12] := pushbutton 3
WITH items [13] := textbox 28
WITH items [14] := UNDETERMINED
WITH items [15] := textbox 35
WITH items [16] := listbox 1
WITH items [17] := listbox 2
WITH items [18] := textbox 36
WITH items [18] := valuebox 17
                                                                                                                                         WITH items [13] = promptbox 8
WITH items [14] = pushbutton 18
WITH items [15] = UNDETERMINED
WITH items [16] = UNDETERMINED
WITH items [17] = UNDETERMINED
                                                                                                                                                                                                                                                                               WITH menus [1 ] := UNDETERMINED
                                                                                                                                                                                                                                                                            INSTANCE listbox I ISA listbox
                                                                                                                                                                                                                                                                              WITH show current instance := TRUE
WITH destination := object ID OF A change data
                                                                                                                                         WITH items [18] := UNDETERMINED WITH items [19] := UNDETERMINED WITH items [20] := UNDETERMINED
                                                                                                                                                                                                                                                                               WITH location := 156,141,280,165
                                                                                                                                                                                                                                                                            INSTANCE listbox 2 ISA listbox
                                                                                                                                        WITH items [20] = UNDETERMINED
WITH items [21] = UNDETERMINED
WITH items [22] = UNDETERMINED
WITH items [23] = UNDETERMINED
WITH items [24] = UNDETERMINED
WITH items [25] = UNDETERMINED
WITH items [26] = UNDETERMINED
WITH items [27] = pusibution 17
WITH items [27] = pusibution 17
WITH items [28] = WITH items [28]
   WITH items [19] := valuebox 17
WITH items [20] := valuebox 18
WITH items [21] := UNDETERMINED
                                                                                                                                                                                                                                                                              WITH show current instance := TRUE
WITH destination := attribute OF A_change_data
WITH location := 415,140,560,165
   WITH items [22] := valuebox 26
WITH fill color := 192,192,192
                                                                                                                                                                                                                                                                            INSTANCE listbox 3 ISA listbox
   WITH meaus [1] := UNDETERMINED
                                                                                                                                                                                                                                                                              WITH show current instance: = TRUE
WITH destination: = object_ID OF A_change_data
WITH location: = 170,256,339,280
INSTANCE responding display ISA display WITH wait := TRUE
                                                                                                                                        WITH items [28] := tenthox 53
WITH items [29] := tenthox 54
WITH items [30] := prompthox 14
WITH items [31] := valuebox 12
WITH items [31] := prompthox 15
WITH items [32] := prompthox 15
WITH fill color := 192,192,192
 WITH wait: = TRUE
WITH delay changes: = FALSE
WITH items [1]: = border 18
WITH items [2]: = UNDETERMINED
WITH items [2]: = UNDETERMINED
WITH items [4]: = UNDETERMINED
WITH items [5]: = UNDETERMINED
WITH items [6]: = UNDETERMINED
WITH items [7]: = UNDETERMINED
WITH items [8]: = UNDETERMINED
WITH items [9]: = UNDETERMINED
WITH items [10]: = UNDETERMINED
WITH items [11]: = UNDETERMINED
WITH items [12]: = UNDETERMINED
WITH items [13]: = UNDETERMINED
WITH items [14]: = UNDETERMINED
                                                                                                                                                                                                                                                                            INSTANCE listbox 4 ISA listbox
                                                                                                                                                                                                                                                                             WITH show current instance := TRUE
WITH destination := attribute OF A change_data
                                                                                                                                                                                                                                                                               WITH location := 170,331,337,355
                                                                                                                                         WITH menus [1.1 := UNDETERMINED
                                                                                                                                                                                                                                                                            INSTANCE listbox 5 ISA listbox
                                                                                                                                      INSTANCE existing data for a design change display
                                                                                                                                                                                                                                                                              WITH show current instance := TRUE
                                                                                                                                      INSTRUCTOR STATE STATES A SISPLEY

WITH delay changes: FALSE

WITH delay changes: FALSE

WITH items [1] := border 13

WITH items [2] := border 14

WITH items [3] := border 17

WITH items [5] := promptbox: 10

WITH items [6] := promptbox: 10

WITH items [6] := promptbox: 12

WITH items [7] := promptbox: 12

WITH items [8] := promptbox: 12

WITH items [10] := UNDETERMINED

WITH items [10] := UNDETERMINED

WITH items [11] := textbox: 48

WITH items [11] := textbox: 49

WITH items [12] := textbox: 50

WITH items [13] := textbox: 50

WITH items [14] := textbox: 51

WITH items [16] := pushbutton: 19

WITH items [16] := pushbutton: 19

WITH items [17] := pushbutton: 20

WITH items [18] := textbox: 56

WITH items [19] := textbox: 56

WITH items [20] := textbox: 57

WITH items [21] := UNDETERMINED

WITH items [22] := UNDETERMINED

WITH items [23] := UNDETERMINED

WITH items [24] := UNDETERMINED

WITH items [25] := textbox: 59

WITH items [26] := textbox: 60

WITH items [27] := UNDETERMINED

WITH items [28] := textbox: 60

WITH items [29] := textbox: 20

WITH items [31] := valuebox: 21

WITH items [31] := valuebox: 23

WITH items: [35] := valuebox: 23

WITH items: [36] := valuebox: 23

WITH items: [36] := valuebox: 23
                                                                                                                                     ISA display
WITH wait := TRUE
                                                                                                                                                                                                                                                                             WITH source := Project_name OF Current_projects
WITH destination := project_name
                                                                                                                                                                                                                                                                              WITH location := 25,150,225,185
                                                                                                                                                                                                                                                                           INSTANCE promptbox 3 ISA promptbox
WITH fost := "System"
WITH justify IS left
WITH frame := TRUE
   WITH items [15] := UNDETERMINED WITH items [16] := UNDETERMINED
                                                                                                                                                                                                                                                                             WITH show current := TRUE WITH formst := "#"
   WITH items [17] = UNDETERMINED WITH items [18] := UNDETERMINED WITH items [19] := UNDETERMINED
                                                                                                                                                                                                                                                                             WITH attachment := Recommendation OF
                                                                                                                                                                                                                                                                             \_change_data
WITH location := 335,100,375,130
   WITH items [20] := UNDETERMINED WITH items [21] := UNDETERMINED WITH items [22] := UNDETERMINED
                                                                                                                                                                                                                                                                           INSTANCE promptbox 4 ISA promptbox
                                                                                                                                                                                                                                                                             WITH font := "System"
  WITH items [23] := UNDETERMINED WITH items [24] := valuebox 15 WITH items [25] := valuebox 13
                                                                                                                                                                                                                                                                             WITH justify IS left
                                                                                                                                                                                                                                                                             WITH frame := TRUE
WITH scroll := TRUE
   WITH items [26] := valuebox 14
                                                                                                                                                                                                                                                                             WITH show current := TRUE
  WITH items [27] := UNDETERMINED WITH items [28] := textbox 26
                                                                                                                                                                                                                                                                             WITH attachment := Reason_for_recomm OF
                                                                                                                                                                                                                                                                             L_change_data
WITH location := 20,155,375,210
   WITH items [29]:= pushbutton 7
  WITH items [30] := pushbutton 8
WITH items [31] := pushbutton 9
                                                                                                                                                                                                                                                                          INSTANCE promptbox 6 ISA promptbox WITH font := "System"
  WITH items [32] := pushbutton 10
WITH items [33] := textbox 27
                                                                                                                                                                                                                                                                             WITH justify IS left
 WITH items [34] := pushbutton 13
WITH items [35] := radiobutton group 2
                                                                                                                                                                                                                                                                             WITH frame := TRUE
                                                                                                                                                                                                                                                                             WITH show current := TRUE
  WITH items [36] := valuebox 9
WITH items [37] := listbox 3
                                                                                                                                                                                                                                                                             WITH attachment := Difference_in_Build_Cost OF
 WITH items [38] := valuebox 10
WITH items [39] := listbox 4
                                                                                                                                                                                                                                                                             WITH location := 290,215,375,245
   WITH items [40] := valuebox [1]
                                                                                                                                                                                                                                                                          INSTANCE promptbox 8 ISA promptbox WITH font := "System"
   WITH items [41]:= textbox 20
 WITH items [42] := textbox 29
WITH items [43] := textbox 30
WITH items [44] := textbox 31
WITH items [45] := textbox 33
                                                                                                                                                                                                                                                                             WITH justify IS left
                                                                                                                                                                                                                                                                             WITH frame :- TRUE
                                                                                                                                                                                                                                                                             WITH show current := TRUE
WITH strachment := DURATION OF A_change_data
WITH location := 230,251,280,281
                                                                                                                                        WITH items [35] := valuebox 23
                                                                                                                                        WITH items [36]:= valuebox 24
WITH items [37]:= valuebox 25
WITH items [38]:= pushbutton 21
 WITH items [46] := textbox 34
WITH items [47] := UNDETERMINED
                                                                                                                                       WITH items [39]:= pushbutton 22
WITH items [40]:= pushbutton 23
WITH items [41]:= pushbutton 24
WITH fill color:= 192,192,192
 WITH items [48] := textbox 37
WITH items [49] := pushbutton 15
                                                                                                                                                                                                                                                                           INSTANCE promptbox 9 ISA promptbox
                                                                                                                                                                                                                                                                             WITH font := "System"
 WITH items [50]:= UNDETERMINED
WITH items [51]:= textbox 38
WITH items [52]:= UNDETERMINED
WITH items [53]:= valuebox 27
WITH items [53]:= valuebox 27
                                                                                                                                                                                                                                                                             WITH justify IS left
                                                                                                                                                                                                                                                                             WITH frame := TRUE
                                                                                                                                         WITH menus [1] := UNDETERMINED
                                                                                                                                                                                                                                                                             WITH show current := TRUE
                                                                                                                                                                                                                                                                             WITH attachment := Recomm
  WITH items [54] := textbox 61
WITH fill color = 192,192,192
                                                                                                                                     INSTANCE select_project_display ISA display WITH wait := FALSE
                                                                                                                                                                                                                                                                             A_change_data_retrieved
WITH location := 335,185,375,215
  WITH menus [1]: = UNDETERMINED
                                                                                                                                        WITH delay changes := FALSE
WITH items [1] := border 20
                                                                                                                                                                                                                                                                          INSTANCE promptbox 10 ISA promptbox
```

WITH font := "System" WITH justify IS left INSTANCE pushbutton 12 ISA pushbutton WITH foot size := 20 WITH label := "Responding" WITH frame := TRUE WITH frame := FALSE
WITH text := "VIRTUAL CHANGE WITH show current := TRUE WITH attribute attachment := responding to change WITH attachment := Difference in Build Cost OF WITH display attachment := responding display MODULE A_change_data_retneved WITH location := 290,270,375,300 WITH location := 130,145,225,175 WITH location = 35 25 165 120 INSTANCE pushbutton 13 ISA pushbutton WITH label := "END" INSTANCE promptbox 11 ISA promptbox WITH font := "System" INSTANCE textbox 2 ISA textbox WITH attribute attachment := exit OF the application WITH pen color := 255,255,255 WITH fill color := 128,0,0 WITH justify IS left WITH frame := TRUE WITH location := 490,360,580,395 WITH justify IS center WITH show current := TRUE
WITH stackment := DURATION OF INSTANCE pushbutton 15 ISA pushbutton WITH font := "Times New Roman"
WITH font style IS bold, italic CF FALSE, underline A_change_data_retrieved WITH location := 240,305,280,335 WITH attribute attach CF FALSE, strikeoul WITH location := 490 265 580 350 CF FALSE WITH font size := 18 INSTANCE pushbutton 18 ISA pushbutton WITH label := "OK" WITH attribute attachment := close change data INSTANCE promptbox 12 ISA promptbox WITH font := "System" WITH frame := TRUE
WITH text := "DEFINE WHAT YOU WANT TO WITH justify IS left WITH frame := TRUE display WITH location := 290,250,375,280 WITH location := 10,10,570,50 WITH show current := TRUE WITH attachment := Required_man_hours OF INSTANCE textbox 4 ISA textbox A_change_data_retneveu WITH location := 240,335,280,365 INSTANCE pushbutton 17 ISA pushbutton WITH label := "CANCEL." WITH pen color := 0,0,0 WITH fill color := 192,192,192 WITH attribute attachment := cance WITH location := 290,285,375,315 WITH justify IS left ncel current change INSTANCE promptbox 13 ISA promptbox WITH font := "System"
WITH text := "Building Comp.:" WITH font := "System"
WITH justify IS left INSTANCE pushbutton 19 ISA pushbutton WITH label := "OK" WITH location := 20,65,160,85 WITH frame := TRUE WITH scroll := TRUE WITH attribute attachment := close existing data INSTANCE textbox 5 ISA textbox WITH show current := TRUE fisplay WITH location := 290,305,375,335 WITH pen color := 0,0,0 WITH attachment := Reason for recomm OF WITH fill color := 192,192,192 nee data retrieved WITH justify IS left WITH Justiny 15 sets
WITH font: "System"
WITH text: "Suiding Comp. ID:"
WITH location: = 20,140,155,160 WITH location := 20,230,375,265 INSTANCE pushbutton 20 ISA pushbutton WITH label := "CANCEL" INSTANCE promptbox 14 ISA promptbox WITH attribute attachment = cancel current change WITH font := "System" WITH location := 290,335,375,365 WITH justify IS center INSTANCE textbox 6 ISA textb WITH frame := TRUE WITH show current := TRUE INSTANCE pushbutton 21 ISA pushbutton WITH pen color := 0,0,255 WITH fill color := 192,192,192 WITH label := "First" WITH attachment := new_value OF A_change_data WITH location := 170,45,375,75 nt := action OF navigate WITH attribute attach WITH justify IS left retrieved data IS first WITH font := "System"
WITH font style IS bold, italic, underline CF FALSE, INSTANCE promptbox 15 ISA promptbox WITH location := 200.85.255.110 strikeout CF FALS\ WITH foot := "System"
WITH justify !S left
WITH frame := TRUE WITH foot size := 5 INSTANCE pushbutton 22 ISA pushbutton WITH label := "N" WITH text := "From the upper Menu" WITH location := 135,65,285,85 WITH show current := TRUE WITH attribute attache ent := action OF navigate WITH attachment := Required_man_hours OF ved data IS next A_change_data WITH location := 230,284,280,314 WITH location := 260.85,285.110 INSTANCE textbox 7 ISA textbox WITH pen color := 0,0,0 WITH fill color := 192,192,192 INSTANCE pushbutton 23 ISA pushbutton INSTANCE pushbutton 3 ISA pushbutton WITH label := "END" WITH label := "P"
WITH attribute attachment := action OF navigate WITH justify IS left WITH font := "System"
WITH text := "Comp. Attribute:"
WITH location := 305,140,415,165 WITH attribute attachment := exit OF application retrieved data IS prev\ WITH location := 425 60 570 125 WITH location := 285,85,310,110 INSTANCE pushbutton 6 ISA pushbutton WITH label : ""New Case" INSTANCE textbox 21 ISA textbox INSTANCE pushbutton 24 ISA pushbutton WITH label := "Last" WITH pen color := 255,255,255 WITH fill color := 255,0,0 WITH attribute attachment := start new case WITH justify IS center
WITH foot: ""Times New Roman"
WITH font style IS bold, italic CF FALSE, underline WITH location := 30,145,125,175 WITH attribute attachs nt := action OF navigate retrieved data IS last INSTANCE pushbutton 7 ISA pushbutton WITH location := 315,85,381,110 CF FALSE, strikeou\ WITH label := "First" WITH attribute attachment := action OF navigate 1 IS INSTANCE pushbutton 25 ISA pushbutton WITH label := "Go Ahead" CE FAT SE first WITH location := 345,140,430,165 WITH font size := 18 WITH frame: TRUE
WITH frame: "TRUE
WITH text: = "RELATED MESSAGES"
WITH location: 10,10,630,50 WITH attribute attachment := go_al WITH location := 35,160,180,195 **INSTANCE** pushbutton 8 ISA pushbuttor WITH label := "Next" INSTANCE radiobutton group 2 ISA radiobutton WITH attribute attachment := action OF navigate 1 IS group WITH pen color := 0,0,0 INSTANCE textbox 23 ISA textbox WITH justify IS left WITH location := 430,140,480,165 WITH fill color := 192,192,192 WITH frame := TRUE WITH font := "System" WITH frame := TRUE **INSTANCE** pushbutton 9 ISA pushbutton WITH group label := "select discipline" WITH show current := TRUE WITH text := "Read and Acted Upon Message (Black WITH label := "Prev." WITH attribute attachment := action OF navigate 1 IS WITH attachment := select_discipline OF WITH location := 20,95,410,120 A_Responding_Data WITH location := 480 140 530 165 WITH location := 350,165,490,405 INSTANCE textbox 24 ISA textbox WITH pen color := 0,128,0 WITH fill color := 255,255,255 INSTANCE text style I ISA text style INSTANCE pushbutton 10 ISA pushbutton WITH label := "Laut" WITH font := "System" WITH justify IS left WITH attribute attachment := action OF navigate 1 IS WITH font := "System"
WITH frame := TRUE last WITH location := 530,140,580,165 INSTANCE textbox 1 ISA textbox WITH pen color := 255,255,255 WITH fill color := 0.0,128 WITH text := "Un Read and Un Acted Upon Message INSTANCE pushbutton 14 ISA pushbutton WITH location := 20,125,410,150 WITH justify IS center WITH label := "END" WITH font := "Times New Roman"
WITH font style IS bold, italic, underline CF FALSE, WITH attribute attachment := exit OF the application INSTANCE textbox 25 ISA textbox

strikeout CF FALS\

WITH location := 215,40,365,78

WITH pen color := 255,0,0

WITH fill color := 255,255,255 WITH justify IS left WITH justify IS left WITH justify IS left
WITH font: "System"
WITH frame: "TRUE
WITH text: "Read but Un Acted Upon Message WITH form: = "System"
WITH text: = "Expected impact on building cost:"
WITH location:= 20,220,255,245 WITH font := "System WITH font style IS bold, italic, underline CF FALSE, strikeout CF FALS\ WITH font size := 5 INSTANCE textbox 43 ISA textbox WITH location := 20,155,410,180 WITH text := "Define from the WITH pen color := 0,0,0 WITH fill color := 192,192,192 upper Menu*
WITH location := 10,215,135,255 INSTANCE textbox 26 ISA textbox WITH justify IS left WITH pen color := 0,0,255 WITH fill color := 192,192,192 WITH foot := "System"
WITH text := "Required design time [hours]:"
WITH location := 20,255,220,275 INSTANCE textbox 30 ISA textbox WITH justify IS left WITH pen color := 0,0,0 WITH fill color := 192,192,192 WITH font := "System"
WITH text := "# of Messay INSTANCE textbox 44 ISA textbox WITH justify IS left WITH location := 5,140,105,165 WITH font := "System"
WITH text := "Building Component ID:" WITH per color := 0,0,0 WITH fill color := 192,192,192 INSTANCE textbox 27 ISA textbox WITH location := 15,255,171,280 WITH justify IS left NSTANCE textbox 27 ISA textbox
WITH pen color := 0,0,255
WITH fill color := 192,192,192
WITH justify IS left
WITH font := "System"
WITH text := "# of Current Message"
WITH location := 155,140,305,165 WITH font := "System"
WITH text := "Required man hours
WITH location := 20,285,165,305 **INSTANCE** textbox 31 ISA textbox WITH pen color := 0,0,255 WITH fill color := 192,192,192 WITH justify IS left **INSTANCE** textbox 45 ISA textbox WITH font := "System"
WITH font style IS bold, italic, underline CF FALSE, WITH pen color := 0,0,128 WITH fill color := 192,192,192 INSTANCE textbox 28 ISA textbox strikeout CF FALS\ WITH justify IS left WITH font: = "System"
WITH font style IS italic
WITH text: = "(0 = Can not be done)
(4 = Highly recommended)" WITH pen color := 0,0,0 WITH fill color := 192,192,192 WITH foot size '= 5 WITH justify IS left WITH text := "Select by double click" WITH font := "System"
WITH text := "Case ID:" WITH location := 10,275,165,305 WITH location := 20,100,245,135 WITH location := 325,65,385,85 INSTANCE textbox 33 ISA textbox WITH pen color := 0,0,0 WITH fill color := 192,192,192 INSTANCE textbox 46 ISA textbox INSTANCE textbox 32 ISA textbox WITH pen color := 0,0,0 WITH fill color := 192,192,192 WITH justify IS left WITH pen color := 255,255,255 WITH fill color := 255,0,0 WITH justify IS center WITH justify IS left WITH foot := "System"
WITH text := "Component Attribute:" WITH foot := "System"
WITH text := "Level of Recon WITH font := "Times New Roman"
WITH font style IS bold, italic CF FALSE, underline WITH location := 15,324,162,350 endation: (0 to 4)* WITH location := 20,185,275,205 CF FALSE, strikeoul INSTANCE textbox 34 ISA textbox WITH pen color := 0,0,255
WITH fill color := 192,192,192
WITH justify IS left t CF FALSE INSTANCE textbox 48 ISA textbox WITH font size := 36 WITH pen color := 0,0,0 WITH fill color := 192,192,192 WITH frame := TRUE
WITH text := "END OF MONITORING MODULE"
WITH location := 77,87,548,307 WITH font := "System"
WITH font style IS bold, italic, underline CF FALSE, strikeout CF FALS\ WITH justify IS left WITH font := "System"
WITH text := "Reasons for the Recommendation INSTANCE textbox 35 ISA textbox WITH pen color := 0,0,255 WITH fill color := 192,192,192 WITH font size := 5
WITH text := "Select by double click" WITH location := 20,210,330,230 WITH justify IS left WITH location := 10,345,165,370 INSTANCE textbox 49 ISA textbox WITH font := "System"
WITH font style IS bold, italic, underline CF FALSE, WITH pen color := 0,0,0 WITH fill color := 192,192,192 INSTANCE textbox 37 ISA textbox WITH pen color := 0,0,255 WITH fill color := 192,192,192 WITH justify IS left strikeout CF FALS\ WITH justify IS left WITH foot := "System"
WITH text := "Expected impact on building cost:"
WITH location := 20,275,255,295 WITH font size := 5 WITH font := "System"
WITH text := "Case ID"
WITH location := 505, 190, 560, 210 WITH text := "Select by double clicking" WITH location := 20,165,175,205 INSTANCE textbox 50 ISA textbox NSTANCE textbox 50 ISA textbox
WITH pen color := 0,0,0
WITH fill color := 192,192,192
WITH justify IS left
WITH foat := "System"
WITH foat := "Required design time [hours]:"
WITH location := 20,310,220,330 INSTANCE textbox 39 ISA textbox INSTANCE textbox 38 ISA textbox WITH pen color := 255,255,255 WITH fill color := 128,0,0 WITH pen color := 0.0,255 WITH fill color := 192,192,192 WITH justify IS left WITH justify IS center
WITH font := "Times New Roman" WITH font := "System"
WITH text := "Discipline which received the WITH font style IS bold, italic CF FALSE, underline INSTANCE textbox 51 ISA textbox WITH pen color := 0,0,0 WITH fill color := 192,192,192 CF FALSE, strikeou\ WITH location := 20,65,345,90 t CF FALSE WITH font size := 18 INSTANCE textbox 36 ISA textbox WITH frame := TRUE
WITH text := "Read the Messages and Respond" WITH justify IS left WITH font := "System WITH pen color := 0,0,255 WITH fill color := 192,192,192 WITH text := "Required man hours:" WITH location := 5,5,580,45 WITH justify IS left WITH location := 20,340,200,360 WITH font := "System"
WITH font style IS bold, italic, underline CF FALSE, INSTANCE textbox 40 ISA textbox WITH pen color := 0,0,0 WITH fill color := 192,192,192 INSTANCE textbox 52 ISA textbox strikeout CF FALS WITH pen color := 255,255,255 WITH fill color := 128,0,0 WITH justify IS left WITH Sont size := 5 WITH font := "System"
WITH text := "Level of Recommendation: (0 to 4)" WITH justify IS center WITH text := "Select by WITH font := "Times New Roman"
WITH font style IS bold, italic CF FALSE, underline double clicking" WITH location := 20,80,290,100 WITH location = 305 165 415 205 INSTANCE textbox 41 ISA textbox WITH pen color := 0,0,0 WITH fill color := 192,192,192 t CF FALSE INSTANCE textbox 20 ISA textbox WITH foot size := 14 WITH pen color := 0,0,0 WITH fill color := 192,192,192 WITH frame := TRUE
WITH text := "Such a Change has been done before"
WITH location := 10,10,390,35 WITH justify IS left
WITH font := "System"
WITH text := "Reasons for the Recommendation WITH justify IS left
WITH font := "System"
WITH text := "Building Component:"
WITH location := 15,195,155,215 INSTANCE textbox 53 ISA textbox WITH pen color := 0,0,0 WITH location := 20,135,380,155 WITH fill color := 192,192,192 INSTANCE textbox 29 ISA textbox INSTANCE textbox 42 ISA textbox WITH justify IS left WITH font := "System" WITH pen color := 0,0,255 WITH fill color := 192,192,192 WITH pen color := 0,0,0 WITH fill color := 192,192,192 WITH text := "Current Value:"

INSTANCE tenthox 54 ISA tenthox WITH pen color: = 0,0,0 WITH fill color: = 192,192,192 WITH justify IS left WITH font: = "System" WITH tent: = "Proposed value:" WITH location: = 20,50,147,70

INSTANCE textbox 55 ISA textbox

WITH location := 20,20,120,40

WITH pen color := 0,0,0
WITH fill color := 192,192,192
WITH justify IS left
WITH font := "System"
WITH tent := "Number of times the change occured:"
WITH location := 20,50,270,70

INSTANCE tentbox 56 ISA tentbox WITH pen color := 0,0,0 WITH fill color := 192,192,192 WITH justify IS left WITH foot := "System" WITH tent := "Case ID:" WITH location := 20,120,80,140

INSTANCE textbox 57 ISA textbox WITH pen color := 0,0,0 WITH fill color := 192,192,192 WITH justify IS left WITH foot := "System" WITH text := "Date:" WITH GESTON := 20.85.65.105

INSTANCE textbox 58 ISA textbox WITH pen color := 0,0,0 WITH fill color := 192,192,192 WITH justify IS left WITH font := "System" WITH text := "Current #:" WITH location := 290.50.357,70

INSTANCE textbox 59 ISA textbox WITH pan color: • 0,0,0 WITH fill color: • 192,192,192 WITH justify IS left WITH font: • "System" WITH text: • "Change Code:" WITH location: • 220,120,300,140

INSTANCE textbox 47 ISA textbox WITH pen color := 0,0,0 WITH fill color := 192,192,192 WITH justify IS left WITH font := "System" WITH text := "Original Value:" WITH location := 20,150,125,170

INSTANCE textbox 60 ISA textbox WITH pen color := 0,0,0 WITH fill color := 192,192,192 WITH justify IS left WITH font := "System" WITH text := "Modefied Value" WITH location := 200,150,310,170

INSTANCE textbox 61 ISA textbox
WITH pen color := 0,0,0
WITH fill color := 192,192,192
WITH justify IS center
WITH font := "System"
WITH text := "Second
Alternative
Change"
WITH location := \$00,280,575,340

INSTANCE textbox 62 ISA textbox
WITH pen color := 255,255,255
WITH fill color := 0,0,128
WITH justify IS center
WITH font := "Times New Roman"
WITH font style IS bold, italic CF FALSE, underline
CF FALSE, strikeou\
t CF FALSE
WITH font size := 18
WITH font size := 18
WITH font size := 18
WITH font size := 35,40,210,100

INSTANCE textbox 63 ISA textbox WITH pen color := 255,255,255 WITH fill color := 0,0,128 WITH justify IS center
WITH font: "Times New Roman"
WITH font style IS bold, italic CF FALSE, underline
CF FALSE, strikeou\
1 CF FALSE
WITH font size: "24
WITH text: "Virtual Mode"
WITH location: "65,35,185,110
INSTANCE valuebox 2 ISA valuebox

INSTANCE valuebox 2 ISA valuebo
WITH justify IS left
WITH font := "System"
WITH frame := TRUE
WITH cloped := TRUE
WITH location := 160,170,290,203

INSTANCE valuebox 3 ISA valuebox
WITH pen color := 0,0,128
WITH fill color := 192,192,192
WITH justify IS center
WITH font := "System"
WITH frame := TRUE
WITH clipped := TRUE
WITH attachment := object OF A_change_data
WITH location := 20,90,280,115

INSTANCE valuebox 13 ISA valuebox

WITH pen color := 255,0,0
WITH fill color := 192,192,192
WITH justify IS center
WITH font := "System"
WITH frame := TRUE
WITH clipped := TRUE
WITH stackment := number instances OF navigate 1
WITH location := 110,140,140,165

INSTANCE valuebox 14 ISA valuebox WITH pen color := 255,0,0 WITH fill color := 192,192,192 WITH justify IS center WITH foot := "System" WITH frame := TRUE

WITH clipped := TRUE WITH attachment := instance number OF navigate 1 WITH location := 305,140,335,165

INSTANCE valuebox 15 ISA valuebox WITH pen color: = 0,0,0 WITH fill color: = 192,192,192 WITH fill color: = 192,192,192 WITH fill color: = 192,192,192 WITH foot: = "System" WITH frame: = TRUE WITH clipped: = TRUE WITH scroll: = TRUE WITH borizontal scroll: = TRUE WITH borizontal scroll: = TRUE WITH location: = 5,50,580,135

INSTANCE valuebox 16 ISA valuebox WITH justify IS center WITH font: = "System" WITH frame: = TRUE WITH clipped: = TRUE WITH attachment: = discipline_to OF discipline_message WITH location: = 340,65,410,90

INSTANCE valuebox 17 ISA valuebox
WITH pen color := 0,0,128
WITH fill color := 192,192,192
WITH justify IS center
WITH font := "System"
WITH frame := TRUE
WITH clipped := TRUE
WITH strachment := object_ID OF A_change_data
WITH location := 155,170,280,200

WITH location := 155,170,280,200

INSTANCE valuebox 18 ISA valuebox
WITH pen color := 0,0,128
WITH fill color := 192,192,192

WITH justify IS center
WITH font: "System"
WITH frame: "TRUE
WITH clipped: "TRUE
WITH attachment: "attribute OF A_change_data
WITH location: 415,170,560,200

INSTANCE valuebox 9 ISA valuebox WITH pen color := 0,0,128 WITH fill color := 192,192,192 WITH justify IS center WITH font := "System" WITH frame := TRUE WITH clipped := TRUE WITH attachment := object OF A_change_data WITH location := 170,190,340,220

INSTANCE valuebox 10 ISA valuebox
WITH pen color := 0,0,128
WITH fill color := 192,192,192
WITH justify IS center
WITH font := "System"
WITH frame := TRUE
WITH clipped := TRUE
WITH attachment := object_ID OF A_change_data
WITH location := 170,285,340,315

INSTANCE valuebox 11 ISA valuebox
WITH pen color: = 0,0,128
WITH fill color: = 192,192,192
WITH justify IS center
WITH fors: = "System"
WITH frame: = TRUE
WITH clipped: = TRUE
WITH stachment: = attribute OF A_change_data
WITH location: = 170,360,340,390

INSTANCE valuebox 12 ISA valuebox
WITH pen color := 255,0,0
WITH fill color := 192,192,192
WITH justify IS center
WITH foat := "System"
WITH frame := TRUE
WITH clipped := TRUE
WITH dipped := TRUE
WITH attachment := old_value OF A_change_data
WITH location := 170,15,375,40

INSTANCE valuebox 19 ISA valuebox
WITH pen color := 128,0,0
WITH fill color := 192,192,192
WITH justify IS center
WITH foot := "System"
WITH frame := TRUE
WITH dipped := TRUE
WITH attachment := instance number OF navigate
retrieved data
WITH location := 360,45,380,70

INSTANCE valuebox 20 ISA valuebox
WITH pen color := 0,0,128
WITH fill color := 192,192,192
WITH justify IS center
WITH font := "System"
WITH frame := TRUE
WITH elipped := TRUE
WITH attachment := number instances OF navigate
retrieved data
WITH jostion := 265 45 285 70

INSTANCE valuebox 21 ISA valuebox WITH pen color: = 0,0,255 WITH fill color: = 192,192,192 WITH justify IS center WITH finst := TSUE WITH frame: = TRUE WITH dipped: = TRUE WITH strachment: = date_time OF A_change_data_retrieved WITH location: = 60,85,195,110

INSTANCE valuebox 22 ISA valuebox WITH pen color: = 0,0,128 WITH fill color: = 192,192,192 WITH justify IS center WITH font: = "System" WITH frame: = TRUE WITH clipped: = TRUE WITH stackment: = CASE_ID OF A_change data_retrieved WITH location: = 85,115,195,140

INSTANCE valuebox 23 ISA valuebox WITH pen color: = 0,0,128 WITH fill color: = 192,192,192 WITH justify IS center WITH fill color: = 185ystem* WITH firame: = TRUE WITH clipped: = TRUE WITH attachment: = CHANGE OF A_change_data_retrieved WITH location: = 305,115,380,140

INSTANCE valuebox 24 ISA valuebox

WITH pen color := 0,0,128
WITH fill color := 192,192,192
WITH justify IS center
WITH foot := "System"
WITH frame := TRUE
WITH dipped := TRUE
WITH attachment := old_value OF
A_change_data_retrieved
WITH location := 120,145,195,170

INSTANCE valuebox 25 ISA valuebox
WITH pen color:= 0,0,128
WITH fill color:= 192,192,192
WITH justify IS center
WITH font:= "System"
WITH frame:= "RUJE
WITH clipped := TRUJE
WITH statchment:= new_value OF
A_change_data_retrieved
WITH location:= 305,145,380,170

INSTANCE valuebox 26 ISA valuebox
WITH pen color := 255,0,0
WITH fill color := 192,192,192
WITH justify IS center
WITH font := "System"
WITH frame := TRUE
WITH disped := TRUE
WITH attachment := Case_ID
WITH location := 315,90,400,115

INSTANCE valuebox 27 ISA valuebox WITH pea color := 255,0,0 WITH fill color := 192,192,192 WITH justify IS center WITH foat := "System" WITH frame := TRUE WITH clipped := TRUE WITH attachment := Case_ID OF discipline_message WITH location := 505,215,565,245

INSTANCE main window ISA window WITH location: 212,134,432,384 WITH menus [1]: UNDETERMINED WITH style IS movesble, sizeable, closeable WITH title: "Change-Monitoring" WITH visible: TRUE

INSTANCE expand window ISA window WITH location: 39,72,793,584 WITH menus [1]: UNDETERMINED WITH style IS moveable, sizeable, closeable WITH title: "Change-Monitoring Module" WITH visible: FALSE

INSTANCE design data window ISA window WITH location: 122,80,510,442 WITH menus [1]: "UNDETERMINED WITH style IS moveable, sizeable, closeable CF FALSE WITH title: ""Data for A Design Change" WITH visible: "FALSE

INSTANCE existing design data window ISA window WITH location:= 114,44,522,450 WITH means [1]:= UNDETERMINED WITH style IS moveable, sizeable, closeable WITH title::= "Similar Change Data" WITH visible:= FALSE

INSTANCE new case window ISA window WITH location := 30,124,615,384 WITH menus [1] := UNDETERMINED WITH style IS moveable, sizeable, closeable

WITH title := "New Case"
WITH visible := FALSE
WITH output := New Case display

INSTANCE responding window ISA window WITH location: 20,5,615,455 WITH menus [1]: "UNDETERMINED WITH style IS moveable, sizeable, closeable CF FALSE WITH title: "Responding to Design Change" WITH visible: "FALSE WITH output: re-responding display

INSTANCE Select a project window ISA window WITH location := 151,149,407,383 WITH means [1]:= UNDETERMINED WITH style IS movesble, sizeable, closeable WITH title := **
WITH visible := FALSE

! DEMON GROUP: action OF navigate

DEMON I
IF action OF navigate 1
THEN CASE_ID OF A_Responding_Data := Case_ID
OF discipline_message
AND PREDECESSOR OF A_Responding_Data :=
sent_because_change_ID OF discip\
line_message

DEMON GROUP: selected OF menu item

DEMON 2 IF selected OF menu item THEN effect of menu item := TRUE

END

C.6. PLANNING AND SCHEDULING MODULE

Part A

WITH title STRING

SVERSIONIS WITH text STRING END SLOCATIONS ARE PIXELS ATTRIBUTE II NUMERIC ATTRIBUTE 59 NUMERIC ATTRIBUTE 51 STRING WITH variable text STRING WITH modal style COMPOUND CLASS available change case id WITH CASE_ID NUMERIC app modal, sys modal ATTRIBUTE delete A_CPMdata SIMPLE
WHEN CHANGED not modal CLASS border INHERITS add on, display item WITH style COMPOUND INIT app modal WITH icon COMPOUND exclamation. picture frame. edit control. information. END SOL group, shadow question stop sien INIT picture fram INIT exclamation WITH perspective COMPOUND WITH button COMPOUND abort retry ignore. INIT is ok caacel WITH border width NUMERIC retry cancel. WITH background color COLOR ves no cancel INIT 192,192,192 WITH fill color COLOR INIT 192, 192, 192 WITH default button COMPOUND one. WITH highlight color COLOR INIT 255,255,255 WITH shadow color COLOR two. INIT one INIT 128,128,128 WITH button selected COMPOUND CLASS current cancel WITH name STRING ignore, 12) CLASS Current_projects WITH Project_name STRING REGIN ok. retry. yes WITH out up SIMPLE CLASS data_elements WITH predecessor STRING WITH CHANGE STRING
WITH alternative_CHANGE STRING CLASS navigate INHERITS add on WITH class attachment CLASS REFERENCE WITH current_cpm_alternative# STRING WITH action COMPOUND END WITH current_data_alternative_in_use STRING first, previous WHEN CHANGED **CLASS** finished next, BEGIN WITH name STRING Inet WITH instance number NUMERIC CLASS ISodbe INHERITS add on WITH number instances NUMERIC alternative# := 1 WITH data source STRING WITH userid STRING CLASS other alternatives WITH password STRING WITH predecessor STRING WITH connection string STRING ARRAY SIZE I WITH connection prompting SIMPLE WITH alternative change STRING FIND data elements INIT FALSE ARRAY SIZE I WITH auto commit SIMPLE NULL. INIT TRUE CLASS temporary current WITH name STRING WHEN FOUND WITH action COMPOUND SHARED ATTRIBUTE CASE_ID NUMERIC disconnec WITH records NUMERIC WHEN CHANGED WITH transaction COMPOUND REGIN alternative#)) delete A_CPMdata := TRUE MAKE current commit rollback END WITH name := WITH append SIMPLE INIT FALSE SHARED ATTRIBUTE project_name STRING WHEN CHANGED WITH status STRING BEGIN CASE_ID) values WITH show error SIMPLE data source OF iSodbc 1 := project_name action OF iSodbc 1 IS connect := TRUE WITH default error handling SIMPLE INIT TRUE EXEC ISodbc | SQL select DISTINCT(CASE_ID) WITH error message STRING WITH native error NUMERIC from [A-CPM-Alternatives] END SQL INTO available change case id :domain.CASE ID) WITH trace file STRING title OF main window := CONCAT("Project : ", FIND END project_name) output OF main window := select_case_display CLASS last alternative data WITH name STRING END ATTRIBUTE I NUMERIC CLASS LastHighLavel ATTRIBUTE start calculations SIMPLE WHEN CHANGED previously used ones (called current) CLASS message box INHERITS add on BEGIN !**** the assumption here is that any particular

delete A_CPMdata := TRUE

```
EXEC ISodbc 1 SQL delete * FROM [A-CPMdata] where CASE ID = :domain.CASE ID
     fill data elements := TRUE
 ATTRIBUTE fill data elements SIMPLE WHEN CHANGED
 food this is to create data elements that have each
 predecessor and all
1000 it possible alternative (each predecessor and one
of its alternatives

!*** is considered as a separate element (instance)
     EXEC ISodbe | SQL select PREDECESSOR
 CHANGE, alternative CHANGE from [A-CPM-Alternatives] where CASE_ID =: CASE_ID END SQL INTO data_elements
 (predecessor, CHANGE, alternative_CHANGE)
 !*** make current_data_alternative_in_use =
 alternative#!(CHANGE)
FOR (i := 1 TO number instances OF navigate
       instance number OF navigate 12 := i
       current_data_alternative_in_use OF
data_elements := CHANGE OF data_elements
END
     make first alternative := TRUE
 ATTRIBUTE make first alternative SIMPLE
 !**** Define the field ALTERNATIVE in A-
CPMdata table in the database
|***** select the starting data elements
|***** also put alternative #1 as the
current_data_alternative_in_use OF data_elements
      WHERE predecessor OF data_elements =
       current_data_alternative_in_use OF
data_elements := CHANGE OF data_elements
CONCAT( "ALTERNATIVE#", TO STRING(
current_data_alternative_in_use OF data_elements
EXEC ISodbc 1 SQL insert into [A-CPMdata] (
PREDECESSOR, CHANGE, ALTERNATIVE,
(:data_elements.predecessor , :data_elements.current_data_alternative_in_use ,
:data_elements.current_cpm_alternative# .
                                  END SQL
100000 Repeat to fill al the A-CPMdata table in the
data base [***** select all the elements that come after the
```

change will always result in the same changes

· ·	WHEN FOUND	I WHEN FOUND
WHILE (name OF current <> "STOP")	iii := 1	current_cpm_alternative# OF data_element
BEGIN	iiiii := instance number OF navigate 11 + 1	:- CONCAT("ALTERNATIVE", TO STRING(
FOR (i := 1 TO number instances OF navigate	FIND END	alternatives))
I) BEGIN	IF (iii = 1) THEN	MAKE temporary current WITH name:
instance number OF navigate 1 := i	BEGIN	current data_alternative_in_use OF data elements
FIND data_elements	current_data_alternative_in_use OF	EXEC ISodbe I SQL insert into [A-
WHERE predecessor OF data_elements =	data_elements := alternative_CHANGE OF	CPMdata] (PREDECESSOR, CHANGE,
WHEN FOUND	data_elements yyy :=	ALTERNATIVE, CASE_ID) values (:data_elements.predecessor,
current_data_alternative_in_use OF	alternative# := alternative# + 1	:data_elements.current_data_alternative_in_use_,
data_elements := CHANGE OF data_elements	į t	:data_elements.current_cpm_alternative#
current_cpm_alternative# OF data_elements	I*** end of IF	:domain.CASE_ID)
:= CONCAT(*ALTÉRNATIVE#*, TO STRING(alternative#))	END	END SQL FIND END
MAKE temporary current	!*** end of DO	!**** end of FOR
WITH name := CHANGE OF	END	END
data_elements	UNTIL (yyy = 1 OR i > number instances OF	10000 make the temporary current as current after
EXEC ISodbe 1 SQL insert into [A- CPMdata] (PREDECESSOR, CHANGE,	navigate [1]	making sure that it has not been used before FORGET current
ALTERNATIVE, CASE_ID) values	li	FOR (ii := 1 TO number instances OF navigate
(:data_elements.predecessor,	1*** to terminate the program if no more alternatives	6)
:data_elements.current_data_alternative_in_use, :data_elements.current_cpm_alternative#,	are generate !*** which occures when yyy reach here with the value	BEGIN instance number OF nevigate 6 := ii
:domain.CASE_ID)	of "0"	yy := 0
END SQL	IF yyy = 0 THEN	FIND finished
FIND END	CHAIN "CPMLOGIC.app"	WHERE name OF finished = name OF
end of FOR END	! ;	temporary carrent WHEN FOUND
!*** make the temporary current as current after	fees to reset the elements that its two alternatives has	yy := 0
making sure that it has not been used before	been used and has been passed	WHEN NONE FOUND
FORGET current	!**** by a higher level element	yy := 1 FIND END
FOR (ii := 1 TO number instances OF navigate 6)	FOR (i := iiiii TO number instances OF navigate	1
BEGIN	BEGIN	IF (yy = 1) THEN
instance number OF navigate 6 := ii	instance number OF navigate 11 := i	BEGIN
yy := 0 FIND finished	FIND data_elements WHERE predocessor OF data_elements =	MAKE current _ WITH name := name OF temporary
WHERE name OF finished = name OF	name OF last alternative data	current
temporary current	AND alternative_CHANGE OF	MAKE finished
WHEN FOUND	data_elements <> **	WITH name := name OF temporary
yy := 0 WHEN NONE FOUND	WHEN FOUND current_data_alternative_in_use OF	current 1999 end if
yy := 1	data_elements := CHANGE OF data_elements	END
FIND END	FIND END	!*** end for
! IF (yy = 1) THEN	END	END FORGET temporary current
BEGIN		!**** end WHILE
MAKE current	[****** Start building the alternative *********	END
WITH name := name OF temporary		FORGET finished
CUITERE MAKE finished	! !***** select the starting data elements	FORGET current FORGET temporary current
WITH name := name OF temporary	!**** also put the current_dats_alternative_in_use	!
CUITTEN	OF data_elements	!*** to continue the process
!**** end if END	FIND data_elements WHERE predecessor OF data_elements =	read the last alternative := TRUE
I*** end for	"NULL"	!*** end of method
END	WHEN FOUND	END
FORGET temporary current	current_cpm_alternative# OF data_elements :=	ATTRIBUTE alternative NUMERIC
end WHILE	CONCAT("ALTERNATIVES", TO STRING(alternatives))	ATTRIBUTE read the last alternative SIMPLE WHEN CHANGED
FORGET finished	MAKE CUITERS	BEGIN
FORGET current	WITH name :=	t
FORGET temporary current	current data_alternative_in_use OF data_elements	!*** define last alternative current cpm_alternative# := CONCAT(
1	EXEC ISodbe 1 SQL insert into [A-CPMdata] (PREDECESSOR, CHANGE, ALTERNATIVE,	"ALTERNATIVES", TO STRING(alternatives))
read the last alternative := TRUE	CASE_ID) values	1
!**** end the method	(:data_elements.predecessor,	1000 delete the existing last alternative data FORGET last alternative data
ATTRIBUTE make other alternatives SIMPLE	:data_elements.current_data_alternative_in_use , :data_elements.current_cpm_alternative# ,	FURGET last attemptive data
WHEN CHANGED	:domain.CASE_ID)	1000 read the data
BEGIN	END SQL	EXEC ISodbc I SQL select CHANGE from (A-
100 find first instance "backward" in last alternative data	FIND END	CPMdata] WHERE ALTERNATIVE =
i:= t	1***** Repeat to fill al the A-CPMdata table in the	:current_cpm_alternative#
iii := 0	data base	END SQL INTO last alternative data (name)
yyy := 0	!**** select all the elements that come after the	1988 so consissos she manages
' _В	previously used ones (called current) !***** the assumption here is that any particular	!*** to continue the process make other alternatives := TRUE
BEGIN	change will always result in the same changes	!
instance number OF navigate 11 := number	1	!** end of method
instances OF navigate 11 + 1 - i	WHILE (name OF current ◆ "STOP") REGIN	END ATTRIBUTE was NUMERIC
i := i + FIND data_elements	BEGIN FOR (i := 1 TO number instances OF navigate	ATTRIBUTE yyy NUMERIC ATTRIBUTE current_cpm_siternative# STRING
WHERE predecessor OF data_elements =	i)	ATTRIBUTE III NUMERIČ
name OF last alternative data	BEGIN	ATTRIBUTE IIII NUMERIC
AND alternative_CHANGE OF data_elements <> ***	instance number OF navigate 1 := i FIND data_elements	ATTRIBUTE go_ahead SIMPLE WHEN CHANGED
AND current_data_alternative_in_use OF	WHERE predecessor OF data_elements =	BEGIN
data elements = CHANGE OF data elements	name OF current	1

WITH items [7]:= UNDETERMINED
WITH items [8]:= UNDETERMINED
WITH items [9]:= UNDETERMINED
WITH items [10]:= UNDETERMINED
WITH items [11]:= UNDETERMINED
WITH items [12]:= pushbutton 8
WITH items [13]:= pushbutton 10
WITH items [14]:= pushbutton 10
WITH items [15]:= pushbutton 11
WITH items [16]:= pushbutton 12
WITH items [17]:= pushbutton 12
WITH items [18]:= UNDETERMINED
WITH items [18]:= UNDETERMINED action OF ISodbe 2 GENERAL IS connect := WITH location := 166,57,302,83 TRUE INSTANCE pushbutton 9 ISA pushbutton EXEC ISodbc 2 GENERAL SOL SELECT WITH label := "connect" Project_name from Current projects WITH attribute attachment := action OF ISodbe IS END SQL INTO Current_projects (Project_name) WITH location := 167,30,309,54 INSTANCE pushbutton 10 ISA pushbutton WITH label := "fill data elements" WITH attribute attachment := fill data elements output OF main window := select_project_display WITH location := 170,87,307,114 INSTANCE border I ISA border WITH style IS shadow WITH items [19] := UNDETERMINED WITH items [19] := UNDETERMINED WITH items [20] := pushbutton 7
WITH items [21] := UNDETERMINED WITH items [22] := UNDETERMINED WITH items [23] := UNDETERMINED WITH menus [1] := UNDETERMINED INSTANCE pushbutton II ISA pushbutton WITH label := "make first alternative" WITH perspective IS in WITH border width := 4 WITH attribute attachment := make first alternative WITH background color := 192,192,192 WITH location := 170.118.322.152 WITH fill color := 0,0,12\$ WITH highlight color := 255,255,255 WITH shadow color := 0,0,0 INSTANCE pushbutton 12 ISA pushbutton WITH label := "next alternatives" INSTANCE starting display ISA display WITH attribute attachment := make other alternatives WITH location := 174,208,324,242 WITH location := 10,10,200,155 WITH wait := FALSE
WITH delay changes := FALSE
WITH delay changes := FALSE
WITH items [1] := border 2
WITH items [2] := border 1
WITH items [3] := textbox 2
WITH items [4] := UNDETERMINED
WITH items [5] := UNDETERMINED
WITH items [6] := UNDETERMINED
WITH items [7] := UNDETERMINED
WITH items [8] := UNDETERMINED
WITH items [9] := UNDETERMINED
WITH items [9] := UNDETERMINED
WITH items [10] := UNDETERMINED
WITH items [10] := UNDETERMINED WITH wait := FALSE INSTANCE border 2 ISA border WITH style IS edit control INSTANCE pushbutton 6 ISA pushbutton WITH perspective IS in WITH border width := 1 WITH label := "read last alternative in order"
WITH attribute attachment := read the last alternative WITH background color := 192,192,192 WITH location := 156,179,354,209 WITH fill color := 192,192,192 WITH highlight color := 255,255,255 WITH shadow color := 128,128,128 INSTANCE pushbutton 7 ISA pushbutton WITH label:= "Start Calculations" WITH attribute attachment:= start calculations WITH location := 5,5,205,220 WITH location := 368,55,502,90 WITH items [11] := pushbutton 13
WITH fill color := 192,192,192
WITH menus [1] := UNDETERMINED INSTANCE the domain ISA domain INSTANCE pushbutton 13 ISA pushbutton WITH label := "Go Ahead" INSTANCE iSodbe I ISA iSodbe WITH connection prompting := FALSE WITH auto commit := TRUE WITH attribute attachment := go_ahead WITH location := 30,170,180,205 INSTANCE select_project_display ISA display WITH wait := FALSE WITH append := FALSE
WITH default error handling := TRUE WITH delay changes := FALSE INSTANCE textbox 2 ISA text WITH delay changes := FALSE
WITH items [1] := border 2
WITH items [2] := border 1
WITH items [3] := UNDETERMINED
WITH items [3] := instoox 2
WITH items [5] := textbox 4
WITH fill color := 192,192,192
WITH menus [1] := UNDETERMINED WITH pen color := 255,255,255 WITH fill color := 0,0,128 INSTANCE ISodbe 2 GENERAL ISA ISodbe WITH data source := "general*
WITH connection prompting := FALSE
WITH auto commit := TRUE WITH justify IS center WITH font := "Times New Roman"
WITH font style IS bold, italic, underline CF FALSE, WITH action IS connect strikeout CF FALSE WITH action is connect
WITH append := FALSE
WITH default error handling := TRUE WITH font size := 16 WITH text := "PLANNING AND SCHEDULING INSTANCE select_case_display ISA display WITH wait := FALSE WITH delay changes := FALSE MODULES INSTANCE navigate 1 ISA navigate WITH class attachment := current WITH location := 16.42, 190, 123 WITH items [1]: border 2 INSTANCE textbox 3 ISA textbox WITH items [2] := border 1 WITH items [3] := listbox 1 WITH pen color := 255,255,255 WITH fill color := 0,0,128 INSTANCE navigate 4 ISA navigate WITH class attachment := finished WITH justify IS center
WITH font := "Times New Roman"
WITH font style IS bold, italic CF FALSE, underline WITH items [4] := textbox 3 INSTANCE nevigate 6 ISA nevigate WITH items [5] := UNDETERMINED WITH fill color := 192,192,192 WITH class attachment := temporary current WITH menus [1] := UNDETERMINED CF FALSE, strikeout CF FALSE INSTANCE navigate 11 ISA navigate WITH font size := 18 INSTANCE dataFileOut ISA file WITH class attachment := last alternative data WITH text := "Select WITH filename := "CPMin.dat" WITH shared := TRUE Change Case ID* INSTANCE navigate 12 ISA navigate WITH class attachment := data_elements WITH location := 20.35.185.120 WITH default error handling := TRUE INSTANCE textbox 4 ISA textbox INSTANCE navigate 13 ISA navigate
WITH class attachment := LastHighLavel INSTANCE events data in ISA file WITH pen color := 255,255,255 WITH fill color := 0,0,128 WITH filename := "cpmout.dat"
WITH shared := TRUE WITH justify IS center
WITH font := "Times New Roman"
WITH font style IS bold, italic CF FALSE, underline WITH default error handling := TRUE INSTANCE the application ISA application WITH unknowns fail := TRUE WITH threshold := 50 INSTANCE listbox I ISA listbox CF FALSE, strikeout CF FALSE WITH font size := 18
WITH text := "Select a Project from the List" WITH title display := starting display WITH show current instance := TRUE WITH ignore breakpoints := FALSE WITH source := CASE_ID OF available change case WITH reasoning on := FALSE WITH location := 15,45,195,105 WITH numeric precision := 8 WITH destination := CASE ID WITH demon strategy IS fire first WITH location := 20,175,190,205 INSTANCE main window ISA window WITH location := 206,121,423,373 WITH menus [1] := UNDETERMINED INSTANCE display I ISA display **INSTANCE listbox 2 ISA listbox** WITH wait := TRUE WITH show current instance := TRUE WITH style IS moveable, sizeable, closeable WITH delay changes: FALSE
WITH delay changes: FALSE
WITH items [1]: UNDETERMINED
WITH items [2]: UNDETERMINED
WITH items [3]: UNDETERMINED
WITH items [4]: UNDETERMINED
WITH items [5]: UNDETERMINED
WITH items [6]: UNDETERMINED WITH source := Project_name OF Current_projects WITH title := ** WITH destination := project_name WITH location := 15,170,195,205 WITH visible := TRUE FND INSTANCE pushbutton 8 ISA pushbutton WITH label := "delete A-CPMdata"

WITH attribute attachment := delete A_CPMdata

Part B

SVERSIONSS WITH icon COMPOUND WHEN CHANGED SLOCATIONS ARE PIXELS exclamation, information !*** this is to create data elements that have each CLASS available change case id WITH CASE_ID NUMERIC predecessor and all lane it possible alternative (each predecessor and one stop sign INIT exclamation of its alternatives CLASS border INHERITS add on, display item WITH style COMPOUND WITH button COMPOUND 1 *** is considered as a separate element (instance)
EXEC 15odbe 1 SQL select PREDECESSOR, abort retry ignore, CHANGE, alternative CHANGE from [A-CPM-Alternatives] where CASE_ID =: CASE_ID END SQL INTO data_elements edit control. ok cancel group, shadow retry cancel, yes no. (predecessor, CHANGE, alternative_CHANGE) INIT picture frame WITH perspective COMPOUND !*** make current_data_alternative_in_use = alternative#1(CHANGE)

FOR (i = 1 TO number instances OF navigate INIT ok WITH default button COMPOUND INIT in two. 12) WITH border width NUMERIC REGIN INTL instance number OF navigate 12 := i current_data_alternative_in_use OF WITH background color COLOR WITH button selected COMPOUND INIT 192,192,192 abort data_elements := CHANGE OF data_elements WITH fill color COLOR cancel. END INIT 192,192,192 ignore, WITH highlight color COLOR INIT 255,255,255 WITH shedow color COLOR BO. make first alternative :- TRUE ok, END retry, ATTRIBUTE make first alternative SIMPLE INIT 128,128,128 WHEN CHANGED yes WITH put up SIMPLE BEGIN **CLASS** current 100000 Define the field ALTERNATIVE in A-WITH name STRING CLASS navigate INHERITS add on CPMdata table in the database WITH class attachment CLASS REFERENCE WITH action COMPOUND alternative# := 1 CLASS Current_projects WITH Project_name STRING |***** select the starting data elements first, CLASS data_elements DEXT. current_data_alternative_in_use OF data_elements WITH predecessor STRING WITH CHANGE STRING FIND data_elements WITH instance number NUMERIC WHERE predecessor OF data_elements = WITH alternative_CHANGE STRING WITH number instances NUMERIC THE LE WITH current_cpm_alternatives STRING WITH current_data_alternative_in_use STRING WHEN FOUND CLASS other alternativ current_data_alternative_in_use OF WITH predecessor STRING ARRAY SIZE I data_elements := CHANGE OF data_elements
current_cpm_alternative# OF data_elements :=
CONCAT(*ALTERNATIVE#*, TO STRING(CLASS finished WITH name STRING WITH alternative change STRING ARRAY SIZE I alternative#))
MAKE current CLASS ISodbc INHERITS add on WITH data source STRING CLASS temporary current WITH name := WITH userid STRING at data_atternative in use OF data_elements
EXEC ISodbe | SQL insert into [A-CPMdata](WITH name STRING WITH password STRING WITH connection string STRING SHARED ATTRIBUTE CASE_ID NUMERIC WHEN CHANGED PREDECESSOR, CHANGE, ALTERNATIVE, WITH connection prompting SIMPLE INIT FALSE CASE_ID) values BEGIN (:data_elements.predecessor, :data_elements.current_data_alternative_in_use, WITH suto commit SIMPLE delete A_CPMdata := TRUE INIT TRUE END :data_elements.current_cpm_alternative# .:domain.CASE ID) WITH action COMPOUND SHARED ATTRIBUTE project_name STRING WHEN CHANGED connect. END SOL BEGIN FIND END data source OF ISodbc 1 := project_name action OF ISodbc 1 IS connect := TRUE EXEC ISodbc 1 SQL select DISTINCT(CASE_ID) WITH records NUMERIC WITH transaction COMPOUND !**** Repeat to fill al the A-CPMdata table in the data base !!**** select all the elements that come after the milhack from [A-CPM-Alternatives] WITH append SIMPLE INIT FALSE previously used ones (called current)

100000000 the assumption here is that any particular END SQL INTO available change case id title OF main window := CONCAT("Project : ", WITH STATUS STRING project_name) change will always result in the same changes WITH show error SIMPLE output OF main window := select_case_display WITH default error handling SIMPLE END WHILE (name OF current <> "STOP") INIT TRUE ATTRIBUTE I NUMERIC BEGIN WITH error message STRING ATTRIBUTE start calculations SIMPLE FOR (i := 1 TO number instances OF navigate WITH native error NUMERIC WITH trace file STRING WHEN CHANGED BEGIN BEGIN delete A_CPMdata := TRUE instance number OF navigate 1 := i CLASS last alternative data WITH name STRING END FIND data elements ATTRIBUTE II NUMERIC ATTRIBUTE Y NUMERIC ATTRIBUTE SI STRING WHERE predecessor OF data_elements = name OF current CLASS LastHighLavel WHEN FOUND ATTRIBUTE delete A_CPMdata SIMPLE WHEN CHANGED current_data_alternative_in_use OF ASS message box INHERITS add on data_elements := CHANGE OF data_elements WITH title STRING WITH text STRING current_cpm_alternatives OF data_elements := CONCAT(*ALTERNATIVES*, TO STRING(EXEC ISodbe | SQL delete * FROM [A-CPMdata]
ere CASE_ID = :domain.CASE_ID WITH variable text STRING alternative#)) MAKE temporary current
WITH name := CHANGE OF WITH modal style COMPOUND END SQL labom ocal fill data elements := TRUE not modal EXEC ISodbe I SQL insert into [A-INIT app modal ATTRIBUTE fill data elements SIMPLE

CPMdata] (PREDECESSOR, CHANGE,

ALTERNATIVE, CASE_ID) values		FOR (ii := 1 TO number instances OF navign
(:data_elements.predecessor, :data_elements.current_data_alternative_in_use,	!*** to terminate the program if no more alternatives	6)
:data_elements.current_cpm_alternative#,	are generate !*** which occures when yyy reach here with the value	BEGIN instance number OF navigate 6 := ii
:domain.CASE_ID)	of "O"	yy := 0
END SQL FIND END	IF yyy = 0 THEN	FIND finished
I**** and of FOR	CHAIN "CPMLOGIC.app"	WHERE name OF finished = name OF
END	li	temporary current WHEN FOUND
loose make the temporary current as current after	!**** to reset the elements that its two alternatives has	yy := 0
making sure that it has not been used before FORGET current	been used and has been passed	WHEN NONE FOUND
FOR (ii := 1 TO number instances OF navigate	10000 by a higher level element FOR (i := iiiii TO number instances OF navigate	yy:**1 FIND END
6)	(1)	ו ו
BEGIN	BEGIN	IF (yy = 1) THEN
instance number OF navigate 6 := ii yy := 0	instance number OF navigate 11 := i FIND data elements	BEGIN
FIND finished	WHERE predecessor OF data_elements =	MAKE current WITH name := name OF temporary
WHERE name OF finished = name OF	name OF last alternative data	current
temporary current WHEN FOUND	AND alternative_CHANGE OF	MAKE finished
yy := 0	data_elements <> "" WHEN FOUND	WITH name := name OF temporary
WHEN NONE FOUND	current_data_alternative_in_use OF	!*** end if
yy := t	data_elements := CHANGE OF data_elements	END
FIND END	FIND END	!*** end for
IF (yy = 1) THEN	END	END FORCET ASSESSMENT OF THE PROPERTY OF THE P
BEGIN	[======================================	FORGET temporary current
MAKE current	[****** Start building the alternative *********	END
WITH name :** name OF temporary current		FORGET finished
MAKE finished		FORGET current
WITH name := name OF temporary		FORGET temporary current
Current	OF data_elements	!*** to continue the process
!**** end if END	FIND data_elements	read the last alternative := TRUE
!*** end for	WHERE predecessor OF data_elements = "NULL"	! !*** end of method
END	WHEN FOUND	END
FORGET temporary current	current_cpm_alternativef OF data_elements :=	ATTRIBUTE alternatives NUMERIC
!**** end WHILE END	CONCAT("ALTERNATIVES", TO STRING(ATTRIBUTE read the last alternative SIMPLE
FORGET finished	alternativef()) MAKE current	WHEN CHANGED BEGIN
FORGET current	WITH name :=	1 200
FORGET temporary current	current_data_alternative_in_use OF data_elements	1*** define last alternative
1	EXEC ISodbc 1 SQL insert into [A-CPMdata] (current_cpm_alternative# := CONCAT(
read the last alternative := TRUE	PREDECESSOR, CHANGE, ALTERNATIVE, CASE_ID) values	"ALTERNATIVE#", TO STRING(alternative#))
!**** end the method	(:data_elements.predecessor,	!*** delete the existing last alternative data
END	:data_elements.current_data_akernative_in_use ,	FORGET last alternative data
ATTRIBUTE make other alternatives SIMPLE WHEN CHANGED	:data_elements.current_cpm_alternatives,	1000
BEGIN	:domain.CASE_ID) END SQL	1 *** read the data EXEC 15odbc 1 SQL select CHANGE from [A-
1 *** find first instance "backward" in last alternative	FIND END	CPMdata]
data i:=1		WHERE ALTERNATIVE =
iii := 0	1***** Repeat to fill al the A-CPMdata table in the	:current_cpm_alternative# END SQL INTO last alternative data (name)
yyy := 0	isses select all the elements that come after the	i
	previously used ones (called current)	!*** to continue the process
DO BEGIN	(***** the assumption here is that any particular	make other alternatives := TRUE
instance number OF navigate 11 := number	change will always result in the same changes	!
instances OF navigate 11 + 1 - i	WHILE (name OF current <> "STOP")	END
i:=i+1	BEGIN	ATTRIBUTE yyy NUMERIC
FIND data_clements WHERE predecessor OF data_clements =	FOR (i := 1 TO number instances OF navigate	ATTRIBUTE current_cpm_alternative# STRING
name OF last alternative data	I) BEGIN	ATTRIBUTE III NUMERIC ATTRIBUTE IIII NUMERIC
AND alternative_CHANGE OF	instance number OF navigate 1 := i	ATTRIBUTE so shead SIMPLE
data_elements 🗢 **	FIND data_elements	WHEN CHANGED
AND current_data_alternative_in_use OF data_elements = CHANGE OF data_elements	WHERE predecessor OF data_elements = name OF current	BEGIN
WHEN FOUND	WHEN FOUND	! action OF ISodbe 2 GENERAL IS connect :=
iii := 1	current_cptn_alternative# OF data_elements	TRUE
iiiii := instance number OF navigate 1 [+]	- CONCAT("ALTERNATIVE#", TO STRING(· •
FIND END	alternative(f)) MAKE temporary current	EXEC ISodbe 2 GENERAL SQL SELECT Project name from Current projects
IF (iii = 1) THEN	WITH name :=	END SQL INTO Current_projects
BEGIN	current_data_alternative_in_use OF data_elements	(Froject_name)
CUITERT_data_alternative_in_use OF data_elements := alternative_CHANGE OF	EXEC ISodbe 1 SQL insert into [A-	
data_elements	CPMdata] (PREDECESSOR, CHANGE, ALTERNATIVE, CASE_ID) values	output OF main window := select_project_display
yyy:= l	(:data_elements.predecessor	END
alternative# := alternative# + 1	:data_elements.current_data_alternative_in_use ,	
! !*** end of IF	:data_elements.current_cpm_alternative#,	INSTANCE border 1 ISA border
END	:domain.CASE_ID) END SQL	WITH style IS shadow WITH perspective IS in
!	FIND END	WITH perspective is in WITH border width := 4
!*** end of DO	!**** end of FOR	WITH background color := 192,192,192
END UNTIL (yyy = 1 OR i > number instances OF	END	WITH fill color := 0,0,128
navigate (1)	!**** make the temporary current as current after making sure that it has not been used before	WITH highlight color := 255,255,255 WITH shadow color := 0,0,0
• -	FORGET current	WITH location := 10,10,200,155
	•	

INSTANCE border 2 ISA border
WITH style IS edit control
WITH perspective IS in
WITH border width:= 1
WITH background color:= 192,192,192
WITH fill color:= 192,192,192
WITH highlight color:= 255,255,255
WITH shadow color:= 128,128,128
WITH location:= 5,5,205,220

INSTANCE the domain ISA domain

INSTANCE ISodbe 1 ISA ISodbe WITH connection prompting: FALSE WITH auto commit: TRUE WITH append: FALSE WITH default error handling: TRUE

INSTANCE ISodoc 2 GENERAL ISA ISodoc WITH data source: ""general" WITH connection prompting: "FALSE WITH auto commit: "TRUE WITH action IS connect WITH append: "FALSE WITH default error handling: "TRUE

INSTANCE navigate 1 ISA navigate WITH class attachment := current

INSTANCE navigate 4 ISA navigate WITH class attachment := finished

INSTANCE navigate 6 ISA navigate
WITH class attachment := temporary current

INSTANCE navigate 11 ISA navigate
WITH class attachment := last alternative data

INSTANCE navigate 12 ISA navigate
WITH class attachment := data_elements

INSTANCE navigate 13 ISA navigate
WITH class attachment := LastHighLavel

INSTANCE the application ISA application WITH unknowns fail: = TRUE WITH threshold: = 50 WITH tale display: = starting display WITH ignore breakpoints: = FALSE WITH reasoning on: = FALSE WITH numeric precision: = 8 WITH demon strategy IS fire first

INSTANCE display 1 ISA display
WITH wait: = TRUE
WITH delay changes: = FALSE
WITH items [1] := UNDETERMINED
WITH items [2] := UNDETERMINED
WITH items [3] := UNDETERMINED
WITH items [4] := UNDETERMINED
WITH items [5] := UNDETERMINED
WITH items [6] := UNDETERMINED
WITH items [6] := UNDETERMINED
WITH items [7] := UNDETERMINED
WITH items [9] := UNDETERMINED
WITH items [9] := UNDETERMINED
WITH items [10] := UNDETERMINED
WITH items [11] := UNDETERMINED
WITH items [12] := pushbutton 9
WITH items [13] := pushbutton 10
WITH items [14] := pushbutton 11
WITH items [15] := pushbutton 12
WITH items [16] := pushbutton 12
WITH items [17] := pushbutton 12
WITH items [18] := UNDETERMINED
WITH items [19] := UNDETERMINED
WITH items [19] := UNDETERMINED

WITH items [21]: UNDETERMINED WITH items [22]: UNDETERMINED WITH items [23]: UNDETERMINED WITH menus [1]: UNDETERMINED

INSTANCE starting display ISA display WITH wait: FALSE WITH delay changes: FALSE WITH items [1]: border 2 WITH items [2]: border 1 WITH items [3]: textbox 2 WITH items [4]: UNDETERMINED WITH items [5]: UNDETERMINED WITH items [6]: UNDETERMINED WITH items [7]: UNDETERMINED WITH items [8]: UNDETERMINED WITH items [8]: UNDETERMINED WITH items [9]: UNDETERMINED WITH items [9]: UNDETERMINED WITH items [10]: UNDETERMINED WITH items [11]: pushbutton 13 WITH fill color: 192,192,192

WITH menus [1]: UNDETERMINED

INSTANCE select_project_display ISA display WITH wait := FALSE
WITH delay changes := FALSE
WITH items [1] := border 2
WITH items [2] := border 1
WITH items [2] := worder 1
WITH items [3] := UNDETERMINED
WITH items [4] := isstox 2
WITH items [4] := isstox 2
WITH items [6] := extbox 4
WITH fill color := 192,192,192
WITH menus [1] := UNDETERMINED

INSTANCE select_case_display ISA display
WITH wait := FALSE
WITH delay changes := FALSE
WITH items [1] := border 2
WITH items [2] := border 1
WITH items [3] := listbox 1
WITH items [4] := textbox 3
WITH items [4] := UNDETERMINED
WITH fill color := 192,192,192
WITH menus [1] := UNDETERMINED

INSTANCE dataFileOut ISA file WITH filename := "CPMin.dat" WITH shared := TRUE WITH default error handling := TRUE

INSTANCE events data in ISA file WITH filename:= "cpmout.dat" WITH shared:= TRUE WITH default error handling:= TRUE

INSTANCE listbox 1 ISA listbox
WITH show current instance := TRUE
WITH source := CASE_ID OF available change case
id

WITH destination := CASE_ID WITH location := 20,175,190,205

INSTANCE listbox 2 ISA listbox
WITH show current instance: "TRUE
WITH source: "Project_name OF Current_projects
WITH destination: "project_name
WITH location: 15,170,195,205

INSTANCE pushbutton 8 ISA pushbutton WITH label := "delete A-CPMdata" WITH stiribute strachment := delete A_CPMdata WITH location := 166,57,302,83

INSTANCE pushbutton 9 ISA pushbutton WITH label := "connect" WITH attribute attachment := action OF ISodbc IS connect WITH location := 167,30,309,54 INSTANCE pushbutton 10 ISA pushbutton WITH label: "fill data elements" WITH attribute attackment: "fill data elements WITH location: 170,87,307,114

INSTANCE pushbutton 11 ISA pushbutton WITH label: = "make first alternative" WITH stribute strachment: = make first alternative WITH location := 170, 118.322, 152

INSTANCE pushbutton 12 ISA pushbutton WITH label: ""rent alternatives" WITH attribute attachment: "" make other alternatives WITH location: "174,208,324,442

INSTANCE pushbutton 6 ISA pushbutton WITH label := "read last alternative in order" WITH attribute attachment := read the last alternative WITH location := 156,179,354,209

INSTANCE pushbutton 7 ISA pushbutton WITH label: ""Start Calsulations" WITH attribute attackment: "start calculations WITH location: "368,55,502,90

INSTANCE pushbutton 13 ISA pushbutton WITH label: ""Go Ahead" WITH attribute attachment: "go_ahead WITH location: "30,170,180,205

INSTANCE textbox 2 ISA textbox
WITH pen color: = 255,255,255
WITH fill color: = 0,0,128
WITH justify IS center
WITH font: = "Times New Roman"
WITH font style IS bold, italic, underline CF FALSE, strikeout CF FALSE
WITH font size: = 16
WITH text: = "PLANNING AND SCHEDULING MODULE"
WITH location: = 16,42,190,123

INSTANCE textbox 3 ISA textbox
WITH pen color := 255,255,255
WITH fill color := 0,0,128
WITH justify IS center
WITH font := "Times New Roman"
WITH font style IS bold, italia CF FALSE, underline
CF FALSE, strikeout CF FALSE
WITH font size := 18
WITH text := "Select
Change Case ID"
WITH location := 20,35,185,120

INSTANCE textbox 4 ISA textbox
WITH pen color: = 255,255,255
WITH fill color: = 0.0,128
WITH justify IS center
WITH fost: = "Times New Roman"
WITH font style IS bold, italic CF FALSE, underline
CF FALSE, strikeout CF FALSE
WITH fost size: = 18
WITH text: = "Select a Project from the List"
WITH location: = 15,45,195,105

INSTANCE main window ISA window
WITH location := 206,121,423,373
WITH menus [1] := UNDETERMINED
WITH style IS moveable, sizeable, closeable
WITH title := ***
WITH visible := TRUE

END