

CONFSYS: The CINDI Conference Support System

Bipin C. Desai

Concordia University, Montreal, Canada

August 1999

Abstract

This paper describes an online conference management system to be used by three groups of users, authors, reviewers and program chair(s). Authors can submit their papers on-line; the reviewers (program committee members) can use the system to select, in an on-line auction, the papers in their domain of expertise for review and later download and review papers assigned to them; and finally the program chair(s) can use the system to automatically perform an initial assignment of papers. This assignment is based on the reviewers' preference from the auction process as well as certain guidelines used in avoiding conflict of interest. The system also facilitates the program committee in debating the review and converging to a decision about the papers. The authors can directly get the results of the review and the blind comments from the reviewers, online. This system is implemented using Java servlet communicating with a MySQL database server.

1 Introduction

In managing an academic conference, the program chair (PC) is required to deal with many repetitive administrative tasks such as: interaction with authors and program committee members (reviewers), paper collection, paper allocation, distributing the paper to the reviewers, collating, sorting and tabulating the evaluations, orchestrating the debate of controversial evaluation of some of the papers, making the final tabulation and preparing the notification and comments to the reviewers and authors.

A number of Web-based systems for automating the above tasks have been reviewed in [1]. Here, we consider the features of the following recent systems: CyberChair[2], START[3] and WitanWeb[4]. As in many of the systems reviewed in [1], they are implemented using script languages: CyberChair uses Python, while START and WitanWeb use Perl. All three systems have features such as electronic paper submission, assignment of papers to reviewers, submissions of reviews by the program committee members, mailings to PC members and authors, etc.

CyberChair system implements the champion identification scheme presented in [5]. The system supports four groups of users: the maintainer, the authors, the PC and the reviewers. It has following major features: author registration, upload of abstract, paper

(either in PS or PDF format), and camera-ready version of paper via web: paper allocation based on reviewers' preferences and expertise, conflict of evaluations detection. The reviewers are asked to fill in the classification of papers on the review form. The possible score assigned to a paper is one of the following: A (representing that the reviewer is willing to advocate and/or accept); B (accept, but could reject); C (reject, but could accept) and D (detractor/ strong reject). CyberChair only consider the lowest and highest classification given to a paper, and shows the evaluation conflicts by a coloring scheme. For example, Red means serious evaluations conflict – this occurs when a paper get scores both of types A and D; green means no conflict – for C and D type of scores.

START [3] uses Perl as its scripting language; it appears that it is no longer actively supported and our attempts to interact with one of its authors were futile. It doesn't provide author registration, a feature that has been added independently by some users. It has the following features: paper submission and paper format conversion from Postscript to PDF; assignment, using a round robin scheme, papers to reviewers taking into account their preferences; provides the reviewers facility to examine details about the papers on-line, including view abstract and download papers, and submit scores; generate basic reports; notify authors if their papers are accepted or rejected.

WitanWeb[4] has one more special user named (external)referees, who are chosen by reviewers to evaluate some papers assigned to them. In WitanWeb, the reviewer has the same permission as the PC. WitanWeb includes the following major features: support paper submission in multiple formats; authors create and update their personal information as well as paper information. They can update and delete their paper files; assigns paper to reviewers; allow them to record the scores; notify authors if their papers are accepted or rejected.

Some of the systems reviewed in [1] are no longer supported. Most of these systems use a script programming languages and CGI technology. However, Java Servlet technology has better performance. Most of this software requires some efforts to handle different situations. So, system administrators are required for set-up; a service offered by the commercialization of these systems. The user interface is non-existent in many systems especially for administration and requires running scripts.

The Conference Management System (ConfSys) presented in this paper is an entirely a Web-based system which provides facility for the Program chair(s) to set up the details for a meeting, allows authors to register and submit papers to the system on-line; records the topic of expertise of the members of the program committee members (reviewers); helps the PC by performing an automatic allocation of the submitted papers to the reviewers. The reviewers have a facility to bid in an auction for papers to review and later to download and review the assigned paper via the Internet. The ConfSys uses the DBLP database [6] in the automatic assignment of papers to avoid any conflict of interest and thus helps in the allocation of papers to the reviewers for a fair and impartial review of each paper.

In section 2, we give the architecture of the system and the flow of events; details of implementing the major features are presented in section 3. Section 4 gives the major functions of ConfSys.

2. Architecture and Sequence of Events

The major part of this system (Figure 1) is a web-based application, which uses a 3-tier framework. The system supports three groups of users: authors, reviewers and program (co) chair(s)(PC). We assume that the reviewers may use external referees, however the reviewers are responsible for the actual evaluations and would participate in their debate. For clear and easy notation, we assume that authors and PC are female and the reviewers are male.

On the client side, the web browser is the only tool to allow users of the system, including paper authors, reviewers and PC, to communicate with server. The server side stores the business logics for the system. By reviewing the client's request, the server will produce an appropriate response. Finally the database supports the server and keeps all information sent by clients.

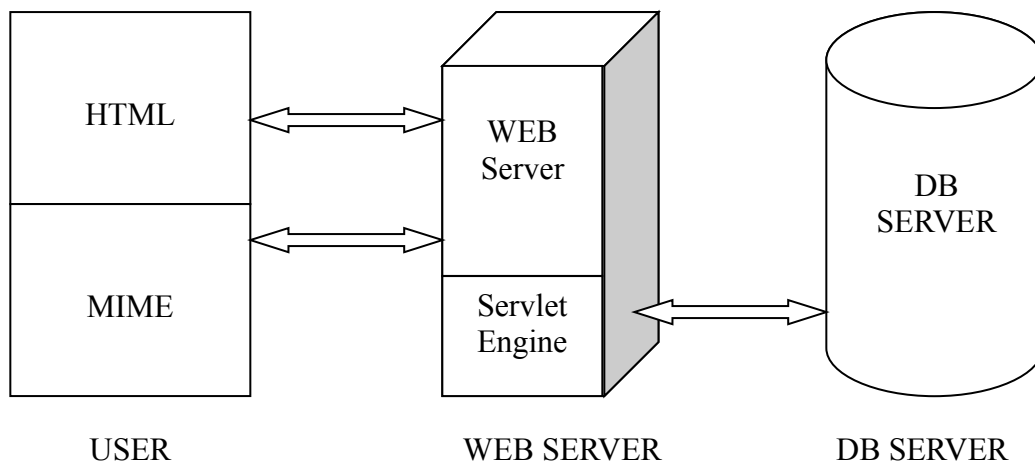


Figure 1. Architecture of ConfSys

In Figure 2, below, we outline the flow of events, the actors involved and the actions that are orchestrated by ConfSys; it thus indicates the interaction of each group and shows the manual and automatic actions supported by it.

The first group of users is the authors who, once the call for papers (CFP) is announced, register with the ConfSys as authors and receive, via email, a user name (UID) and password (PW). Using the UID and PW they can submit their papers and later get the results of the reviews directly through ConfSys. The second group of users is the reviewers who are emailed their user ID and password by the system once the PC enters

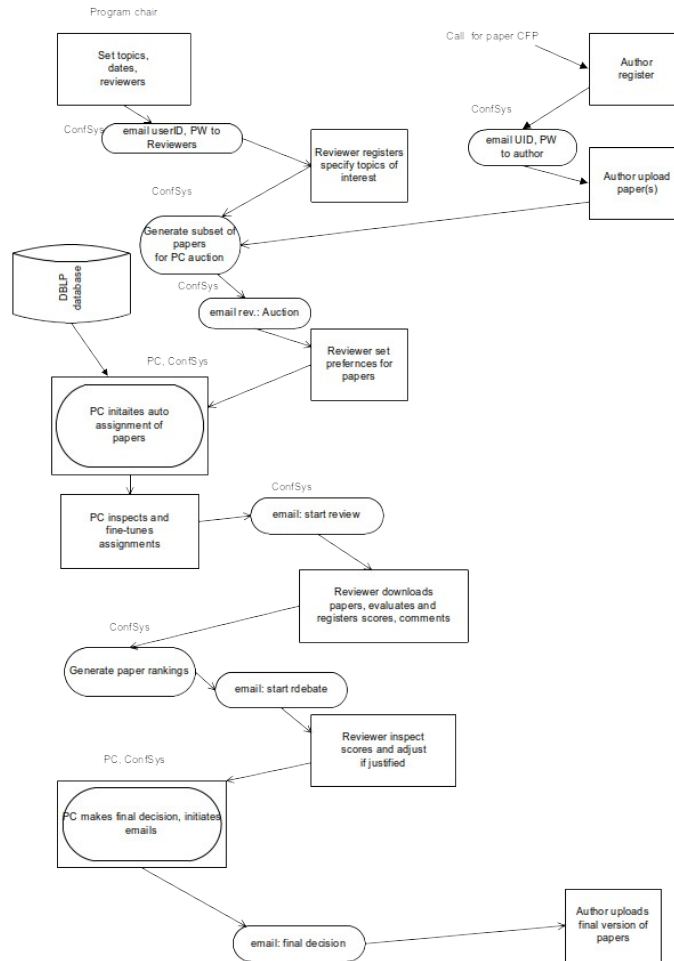


Figure 2. ConfSys Sequence of Events

this information during the setup of the system for an academic meeting; using these, they register their topic of interests that would be used in selecting a subset of papers for the auction and automatic paper assignment processes. The reviewers can download papers, evaluate them and submit the result to ConfSys. The third group is the program chair(s) (PC) who is responsible for setting up the system for a convocation of a meeting, by inserting the reviewers' name and emails, the topics of the meeting and the important dates; the PC initiates the automatic allocation of the papers to the reviewers and do the fine-tuning, initiate the debate and make final decision on papers and instructs ConfSys to send the notification of the conclusion of the evaluation to the authors by email. The first user group is the largest, usually several hundred people for an average conference; second group involves tens of users, usually one hundred or less reviewers. There are only one or two users in last group.

For the functions of the first and second user group, we need to consider concurrency, synchronization and database locks. Functions for last user group involve complex logic for example, in the algorithm for automatic papers allocation to reviewers; however, because of the limited number of users, they require minimal multi-user consideration.

3. Implementation

Since this is a web-based application, it makes sense that users can use browsers (IE or Netscape) to communicate with the server. Other function to be added are the ability to upload papers by authors, and the ability of the reviewers to download the papers they are to evaluate and submit their score and comments on them. They also need to debate the reviews and if needed modify some scores in case of contradictions in the evaluation of a given paper. The application allows authors to upload file from client side to server side, not only in HTML format but also MIME format. In order to handle non-HTML package sent by clients using browser, there is a need of an additional tool on the server side. Normally this tool is called a multi request handler. For ConfSys system we use two java classes called MultiRequest and MultiRequestServlet [7].

Since this application uses Java Servlet as middleware, a servlet engine needs to be deployed on the server side. We have implemented the current version using Tomcat [11] as the servlet engine. Though Tomcat includes function of a web server, it is not advisable to use it as both web server and servlet engine due its poorer performance as a web server. ConfSys hence uses Apache [12] as the HTML server and Tomcat as the servlet engine extension of Apache to support Java servlet classes.

The back end is the database server. MySQL[13] is used as the database server in this application; it is suitable for middle size application and it interfaces easily with Java or C/C++. In the case users try to query the database, the request finally will be passed to the servlet engine and a Java class will do the job for the query.

3.1 File Uploads

To upload a file from client to server via Internet, we must use the HTTP (0.9, 1.0, 1.1) protocol. HTTP [8] is designed to handle text transmission; however, files to be uploaded can be in many format, for example PDF or word processor file format. Since HTTP 1.0, includes MIME (Multipurpose Internet Mail Extensions [9]), a client can send data uploaded in multipart for one request. If there are more than one data types in a HTTP stream, the keywords “Multipart” and “Boundary” will appear. Multipart indicates the information for the embedded part and Boundary quotes the beginning and the end of the embedded part. Correspondingly, in the server side there must be a way to receive the embedded part for one HTTP request. In ConfSys, we use a Java class called MultitRequest [7] to extract the embedded part on the server side.

We allow the author the flexibility in uploading a file; this includes canceling an uploaded file or replace it by another. However, since the embedded part in a multipart request can only be from a web client to a web server, there is no way for a web server to send back the embedded part to a web browser. This causes a problem to propagate the name of the uploaded file from one web page to another web page needed for the above feature. To solve this problem, we use a child window for the uploading page instead of

using an upload button directly in the main window. The child window is used to select the file to be uploaded; this file is stored in a temporary folder on the server and the file name is written to the main window on the client screen. File name can be carried from page to page and get to be verified. If the client makes the final confirmation for the uploaded file and submitted related information, then the file will be moved from a temporary folder to the regular folder for uploaded files. If the author decides to delete the uploaded file during processing, the file in the temporary folder would be deleted. If the author aborts the submission process, the uploaded file will remain in the temporary folder. If an author returns to resubmit a file with the same name as an aborted upload of a file in the temporary folder, it will be overwritten.

File upload function is a multi-user function and requires concurrency control to avoid file with the same name submitted by more than one user overwriting each other¹. To avoid the file being overwritten, one of the users will be requested to change the file name. Once an author click to confirms the upload, the server saves the file into regular folder with the user specified name that is guaranteed to be unique.

3.2 DBLP System and its conversion to MySql table

Digital Bibliography & Library Project (DBLP) [6], contains information on conference papers, journals papers, books and thesis etc. Each DBLP file, in XML, represents one item; there being thousands of such files in tree-structured directories. Since searching XML files in such a tree-structured directories is time consuming with Java, we have pre-processed the information from these files into a relational database; this pre-processing step is needed only once; after which it can be used efficiently. For conflict avoidance, we only need to verify the co-author relationship; hence we create a table with three columns. One is key, which is an incremental number. Another is the path of the file, which we need to verify during the automatic assignment process. The last is a string, which contains the authors of each item in the DBLP database. If the reviewer's name and of the author's name appear in the same tuple of this table, then it is considered a conflict.

Pre-processing the DBLP files is straightforward since all the authors are tagged. We used a C program for this process and insert the result directly in the MySql database. This approach has excellent performance since querying a table in the database is much faster than searching XML files in a tree structure directories using a Java program.

3.3 Paper allocation algorithm

Conf Sys has three paper allocation schemes. These are:

- (a) Manually by the PC,
- (b) Automatic followed by fine-tuning by the PC,
- (c) PC auction followed by an automatic allocation followed by fine-tuning.

¹ Many authors tend to use name for files such as paper03.pdf or ideas03.pdf

For the PC auction, ConfSys uses a scheme where reviewers, indicate via web forms, their areas of expertise from the list of topics for the conference. The authors are required to indicate the topics for their papers as well. Once the call for paper (CFP) deadline is over, the system generates a list of papers, which match their expertise and/or a random subset of papers for each reviewer and the reviewer is requested to indicate their priority (high, average, low) for reviewing the papers.

In the case GC member uses the automatic allocation, several constrains will be applied to this function. There is no limitation for GC's manual allocation or for the fine-tuning stage.

Following are the constrains for the automatic allocation:

- i. The same paper should not be allocated to same reviewer more than once.
- ii. A paper cannot be allocated to a reviewer if either the limit of papers for the reviewer or the number of reviewers assigned to the paper is reached.
- iii. A paper can be allocated to a reviewer if the intersection of the set of the paper's topics and reviewer's interests is not empty.
- iv. A paper can't be allocated to a reviewer if one of the authors and the reviewer are from the same organization.
- v. A paper can't be allocated to a reviewer if one of the authors and the reviewer were co-authors in the past.

For the constrains i to iv, we can use the information from the system database; for constrain v, we use the information generated from the DBLP database for the decision.

For the constrain i, the system can check the allocation table in the database. For the constrain ii, it uses the parameters set by PC to indicates the number of reviewers for a paper and the maximum number of papers for each reviewer. For the constrain iii, the system uses the paper's topics, a required field when an author submits a paper, with the reviewer's area of interests. If there is an overlap, then the paper can be allocated the reviewer. For the constrain iv, we can use the information provided by paper author and reviewers.

The problem of expressing the same organization in different ways is resolved using a number of heuristics such as matching the initial letters of a multiword organization or using different abbreviations etc. For example, University of Quebec at Montreal in French is written as Université du Québec à Montréal, UQAM, or Univ. du Que. à Montréal. For authors we need to use similar heuristics for names; for example, Hong Feng Lee can be written as H. F. Lee or as Hong F. Lee.

The automatic allocation algorithm also considers the preference of the PC as well as the above-mentioned constraints. CyberChair and START also implemented their algorithms to allocate papers to reviewers based on expertise and preferences. However, they simplify the problem by assigning any number of papers to reviewers.

An optimal allocation algorithm based on preferences could be very complex and time consuming to execute. The algorithm used in ConfSys is sub-optimal but much simpler to implement and the allocation constraints, in most cases, are satisfied without additional fine-tuning by the PC. The algorithm attempts to assign most reviewers their preferences unless there is an apparent conflict of interest.

The algorithm gives higher priority to reviewer's first preference before considering a match of topics. The automatic algorithm assigns a paper to a reviewer who has indicated a high priority for a paper, even though the author specified topics for the paper and the reviewers' interests are not overlapping. During the setup of ConfSys, the PC assigns the number of reviews for each paper (called reviewer limit, R_i); and the maximum number of paper to assign to each reviewer (called paper limit, P_i)

The automatic allocation algorithm has the following features:

- In the auction process, the reviewer can assign high, average, and low preference or no interest, for each paper. The automatic allocation algorithm, categorizes all submitted papers into four groups: paper with at least one or more high, average, low priorities and the same topic.
- Once the auction process is over, the automatic allocation algorithm sorts the paper in each category based on their count. For example, papers assigned high preferences are sorted on paper id and the total number of high priority for the paper. The algorithm will attempt to allocate papers to reviewers who have assigned high priority. Thus a paper having the largest number of high priority will be allocated earlier in the process.
- If the number of priorities assigned to a paper is $\geq R_i$, then the paper can be assigned based only on the priorities.
- If the number of high priority assigned to a paper is $\leq R_i$, then the algorithm can assign paper directly to those reviewers with high priority followed by lower priority until the limit R_i is reached.
- If the number of reviewers assigned to a paper based on the priority is $< R_i$ then matching of topics of the paper with the reviewers' interests is used.
- For each assignment, the algorithm checks if one of the authors and the reviewer is working in the same organization, or if one of the authors and the reviewer were co-author of paper using DBLP.
- For each assignment, the algorithm checks both R_i and P_i
- In choosing to assign a paper to a reviewer, the algorithm chooses a reviewer with the minimum assignment number; each time a reviewer is assigned a paper, his assignment number is incremented.
- The remaining papers, which are not in any of the four categories, would not be assigned by the algorithm; PC is required to allocate them manually during the fine-tuning stage.

3.4 Database connection strategy

To connect to the MySQL database using Java, we must use a JDBC driver for MySQL. Since there is a lot of common code for initiating a JDBC connection, a better way to implement the DB connection is to use a centralized Java class to do all database operations. Thus setting database connection in each Java class where we need to communicate with database is avoided. The common Java class is called ConferenceSql. When we need a database connection in a class we need to first “instance” the ConferenceSql class then simply use its functions. This not only centralizes the connection functions in one class but also makes these functions reusable. One requirement is that all ConferenceSql's functions that are to be accessed from other classes must be made "static". This is because the JDBC driver is "static" class and we can't wrap a static class with a non-static class in Java.

4. Major functions of ConfSys

The PC sets up the parameter for a conference by interacting with the ConfSys via the PC menu shown in Figure 3. Details of some of these are given below.

4.1 Topic control

Since every conference has its own topics, the PC must have a way to insert the topics, the reviewers and other parameters for the event. These functions give the PC a way to add, delete and modify the topics. The interface is shown in Figure 4.

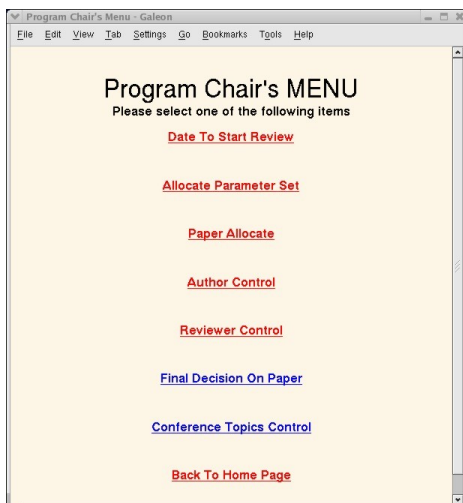


Figure 3. Program Chair Menu



Figure 4 Conference Topic Control

4.2 Reviewer control

The program committee members (reviewers) are entered in the ConfSys as shown in Figure 5. Through this interface, the program chair can add a new reviewer, delete an existing reviewer who could not participate or modify the details for an existing reviewer. The details entered are the reviewer's name and email address. The system generates a random user ID and a password. These are sent by email to reviewers who can interact with ConfSys through the reviewer menu shown in Figure 6; they are required to register their topics of interests as shown in Figure 7.

The screenshot shows a web browser window titled "PC Member Control - Galeon". The page displays a table with the following columns: PCID, PASSWORD, LASTNAME, FIRSTNAME, ORG, EMAIL, MODIFY, and DELETE. The table contains 20 rows of reviewer information.

PCID	PASSWORD	LASTNAME	FIRSTNAME	ORG	EMAIL	MODIFY	DELETE
xjin	xjintest02	xjin	xjin	Concordia University	xjin@cs.concordia.ca	MODIFY	DELETE
201	neamec10	Traina	Agma	null	null	MODIFY	DELETE
107	niridr41	Dogac	Asuman	null	null	MODIFY	DELETE
143	naexam69	Iyer	Bala	USA	balayer@us.ibm.com	MODIFY	DELETE
104	maixas50	Ooi	Beng-Chin	National University of Singapore, Singapore	ooibc@comp.nus.edu.sg	MODIFY	DELETE
112	nugumt35	Kemme	Bettina	null	null	MODIFY	DELETE
133	nixoe15	Kerherv	Brigitte	null	null	MODIFY	DELETE
144	magats11	Wai-lok Cheung	David	Hong Kong, China	dcheung@csis.hku.hk	MODIFY	DELETE
415	nirzm89	Lee	Dik Lun	Hong Kong University of Science and Technology, Hong Kong SAR, China	dlee@cs.ust.hk	MODIFY	DELETE
316	nienig96	Papadias	Dimitris	Dept. of Computer Science, HKUST	dimitris@cs.ust.hk	MODIFY	DELETE
324	nuuzud47	Sacca	Domenico	null	null	MODIFY	DELETE
125	namanz12	Bertino	Elisa	University of Milano	bertino@ dico.unimi.it	MODIFY	DELETE
126	miamia45	Paciti	Esther	null	null	MODIFY	DELETE
327	nisind80	Casati	Fabio	HP, USA	fabio.casati@hp.com	MODIFY	DELETE
128	mecets08	Ling	Feng	null	null	MODIFY	DELETE
160	muuwuk30	Lochovsky	Frederick	HKUST, Hong Kong	fred@cs.ust.hk	MODIFY	DELETE
261	merem159	Navarro	Gonzalo	null	null	MODIFY	DELETE

Figure 5 Inserting reviewers

The screenshot shows a web browser window titled "Reviewer's Menu - Galeon". The page displays a menu with the following items:

- Reviewer's MENU**
- Please select one of the following items
- [Reviewer Registration](#)
- [Reviewer Auction](#)
- [Information About Score & Confidence](#)
- [PC Paper File Download](#)
- [Reviewer Review](#)
- [Final Decision On Paper](#)
- [Back To Home Page](#)

Figure 6 Reviewer Menu

The screenshot shows a web browser window titled "Reviewer's Registration OR Modification - Galeon". The page displays a registration form with the following fields:

- Reviewer's Registration OR Modification**
- Please fill in the following information(all fields are required)
- Last Name: Doe
- First Name: Jane
- ORG,Country: OkeefooneeKee Univ
- Address: OUniversity
- Email Address: Johnedoe@cs.ou.zz
- Salutation: Dr.
- Please choose from the following, the topics of your interest. Use the 'Ctrl' key when you select multiple items(required)
- Access Methods and Data Structures
- Active Databases
- Authorization and DB Security
- Bioinformatics
- Concurrency Control and Recovery
- Constraint and Rule Management
- Content Based Searching
-

Figure 7. Reviewer registration

4.3 Setting up parameters for automatic allocation

For the automatic allocation function, the PC needs two parameters to limit the number of papers a reviewer can review and the number of reviewers for each paper P_1 and R_1 . The PC tries to avoid the situation where some reviewers get too many papers while others have very few. She also makes sure that all papers are reviewed by at least R_1 reviewers.

4.4 Paper uploading function

Once an author registers with the ConfSys, she is sent a password to the email address specified in the registration. She can use the password and login to the system. She is presented with a menu as shown in Figure 8. Selecting the paper upload function she is presented with a form (Figure 9), which she uses to fill in her information and those of co-author(s) and to upload the file for the paper. Once she clicks the "continue" button, she is presented with a confirmation page; it displays every required field that was filled correctly with a warning message if a required field is missing. She can click the "confirm" button if everything is in order. If she needs to modify something, she can click "modify" button to return to the page shown in Figure 9. All the information she filled will be re-displayed including the file name using the scheme discussed in section 3.1. If and only if when she confirms, the file for the paper and all the information for it would be saved in the server.

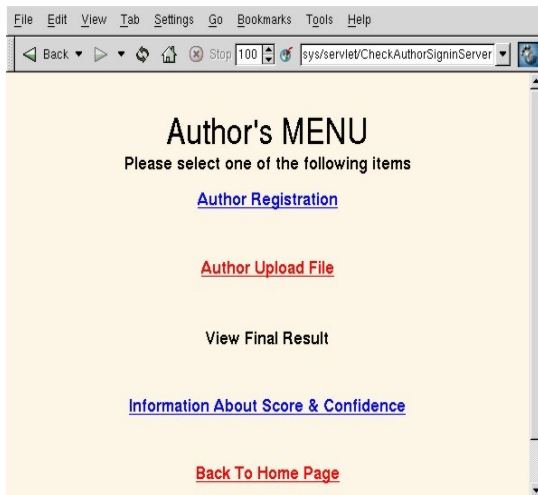


Figure 8. Author menu

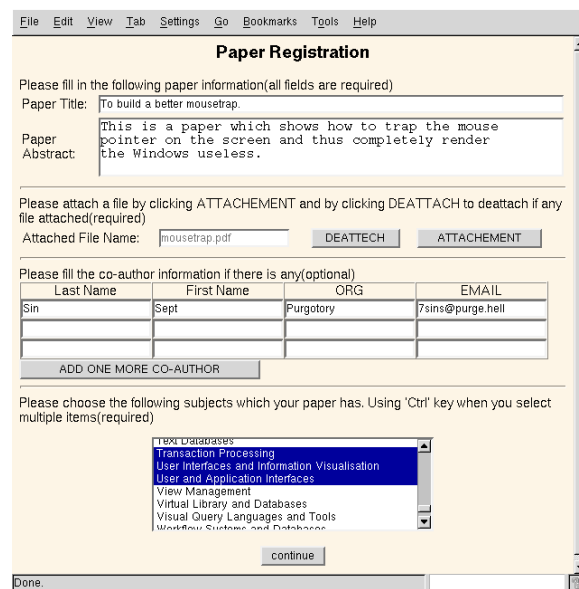


Figure 9. Paper Uploading

4.5 Paper allocation, review and debate

Once the paper submission deadline is over, the PC can use the auto allocation function, discussed earlier. At the start of the process, the number of reviewers for each paper is 0.

Figure 10 shows a partial allocation made manually. The auto-allocation function would attempt to assign reviewers for most of the papers and the most of reviewers would be assigned some of the papers. If the PC clicks the detail button for each paper she will see the names of reviewers for the paper. Notice that it is possible that some papers would have very few or no reviewers assigned. This is because these papers do not match the interests of reviewers or those who can review these papers have already been allocated other papers by the allocation algorithm or there may be conflicts. In this case, the PC must fine-tune the allocation manually and assign these papers to reviewers regardless of the allocation rules used for automatic allocation. Also the PC may not be satisfied with the result of automatic allocation. In which case, she can use the manual allocation function to do some adjustment. Once the PC feels comfortable with the allocation, she can set the start review date to let reviewers review the papers.

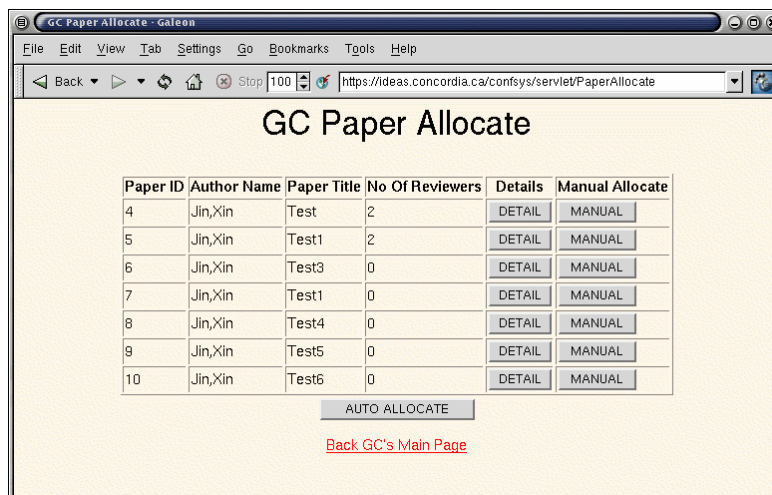


Figure 10

Once the PC has allocated the papers to a reviewer, she can set the start review date to enable the reviewer to download the files for papers allocated to them. There is a security concern for downloading the files for papers. If the files are put in a folder that is accessible by the web server it is possible for any one, who knows the name of the directory and file name, to download it. To avoid this drawback, we don't put the files in a web server accessible folder but let the servlet read the files from a non-web directory and return the stream for downloading by the client. Since the download servlet can only be accessed by the reviewers, only they can download the assigned files. This is an application level security.

A reviewer can download the papers allocated to him and reviews them; he can use the Review link from the reviewer menu (Figure 6) to register the evaluation information including score, and comments both for the author and the program chair(s). The result will be sent to the server and the values will be refreshed in the main evaluation window. The comments and scores are modifiable by the reviewer (during the debate stage of the

evaluation process) until the PC makes a final decision on a paper. Once a final decision on a paper is made, that paper could not be evaluated by any reviewers who had not submitted the review for the paper nor can an already submitted review be modified. A message will be displayed to indicate that a final decision on this paper has been made.

4.5 Final decision and communication to authors

Once the majority of the reviews for each paper have been made, the program chair needs to make a decision on each paper. Based on the reviewers' scores, weights and comments the program chair can either accept a paper as full paper or short paper or simply reject the paper. In the web page given in Figure 11, the program chair can click the detail button to see the complete evaluation from each reviewers for the paper in a pop-up window and can make final decision for the paper in that window. Once a decision has been made, the result will appear in the decision column.

PAPER ID	TITLE	DOWNLOAD	AUTHOR NAME	AUTHOR ORG	DETAIL	AVG_SCORE	AVG_CONF	WEIGHT	DECISION	MAIL SENT	EMAIL
4	Test	Test.pdf	Jin,Xin	Concordia University	DETAIL	7	2.5	7	Accepted	Yes	MAIL
5	Test1	Test1.zip	Jin,Xin	Concordia University	DETAIL	5.3	2.8	4.5	Rejected	No	MAIL
6	Test3	Test3.pdf	Jin,Xin	Concordia University	DETAIL	0	0	0		No	MAIL
7	Test2	Test2.doc	Jin,Xin	Concordia University	DETAIL	0	0	0		No	MAIL
8	Test4	Test4.txt	Jin,Xin	Concordia University	DETAIL	0	0	0		No	MAIL
9	Test5	Test5.pdf	Jin,Xin	Concordia University	DETAIL	0	0	0		No	MAIL
10	Test6	Test6.zip	Jin,Xin	Concordia University	DETAIL	0	0	0		No	MAIL

Figure 11

Once a final decision is made, the program chair(s) initiate ConfSys to send emails to inform authors and reviewers to check the final results of papers. The comments made by the reviewers meant only for the program chair(s) will not be displayed to the authors.

The author can only view the reviews her paper(s). The author can see all reviewers' IDs, comments, score, confidence, and the weighted score as well as the final decision for the paper. The final result is only available after program chair(s) makes final decision and sends e-mail to them. If the paper is accepted, author can make refinements on the paper and resubmit the paper.

Reviewers can view final results of papers that they reviewed. Other reviewers' comments, scores, weight, etc. will be also be shown in the window without the names of any reviewers. The final result is only available after the program chair(s) makes final decision and the systems sends emails the authors and reviewers.

5. Conclusions

Confsys can coordinate the authors, reviewers and PC to process conference paper via Internet. It also reduces the manual operation normally to be done by the PC and streamline the paper allocation operation as well as the collecting of reviews and tabulating the final results. All the software packages used in this application are Open Source. These software packages are very stable and proved by the Open Source community to be reliable.

ConfSys has been used in a test environment and successfully in production environment. It is a component of the Cindi Digital library [15]; our goal is to provide not only a conference management system but also to add the papers submitted to the digital library for better dissemination. Conf Sys is in the process of being released to support academic conferences and symposia.

Reference

- [1] R. Snodgrass. Summary of Conference Management Software. <http://www.acm.org/sigs/sgb/summary.html>
- [2] CyberChair's homepage. <http://cyberchair.cs.utwente.nl>
- [3] START's homepage, <http://www.cs.umd.edu/~rich/START/README.html>
- [4] WitanWeb's homepage, <http://witanweb.iit.nrc.ca/>
- [5] O. Nierstrasz. Identify the Champion, An Organizational Pattern Language for Programme Committees. In *Pattern Languages of Program Design 4*, vol. 4, Addison Wesley, 2000, pp. 539-556, <http://iamwww.unibe.ch/~oscar/Champion/>
- [6] DBLP Reference Site, <http://dblp.uni-trier.de/db/>
- [7] MultiRequest and MultiRequestServlet, <http://www.geocities.com/jasonpell>
- [8] RFC 1945, <http://www.w3.org/Protocols/rfc1945/rfc1945>
- [9] RFC 1952, <http://www.cis.ohio-state.edu/cgi-bin/rfc/rfc1952.html>
- [10] JDK 1.3 Reference Site, <http://java.sun.com/j2se/1.3/>
- [11] Tomcat Reference Manual, <http://jakarta.apache.org/tomcat/tomcat-3.3-doc/index.html>
- [12] Apache, <http://www.apache.org/>
- [13] MySQL reference, <http://www.mysql.com/documentation/mysql>
- [14] Java Mail Site, <http://java.sun.com/products/javamail/>
- [15] B.Desai, CINDI – a digital Library, Reprt, Dept. of Comp Sci, 1998