



# TOWARDS A SELF-FORENSICS PROPERTY IN THE ASSL TOOLSET

Serguei A. Mokhov, Emil Vassev, Joey Paquet, and Mourad Debbabi

Computer Science and Software Engineering, Concordia Institute for Information Systems Engineering, Concordia University, Montreal, Canada  
 School of Computer Science and Informatics, University College Dublin, Dublin, Ireland

## BACKGROUND

- Autonomic Computing (AC)
  - applies the principles of self-regulation and complexity hiding to software and hardware;
  - emphasizes the reduction of the workload needed to maintain complex systems by transforming them into self-managing autonomic systems.
- ASSL Toolset
  - A collection of tools to compile ASSL specifications into autonomic system skeletons in Java
- Forensic Lucid
  - Forensic case modeling and specification with evidence encoded as well as a crime scene described
- Self-Forensics
  - Combines self-diagnostics, reporting, analysis, and reaction to of the incidents within a software/hardware system
- JOOIP
  - hybrid OO intensional programming with Java objects and Lucid
- General Intensional Programming System (GIPSY)
  - To compile and evaluate JOOIP and Forensic Lucid programs (among other Lucid and hybrid dialects)

## AUTONOMIC SYSTEM SPECIFICATION LANGUAGE (ASSL)

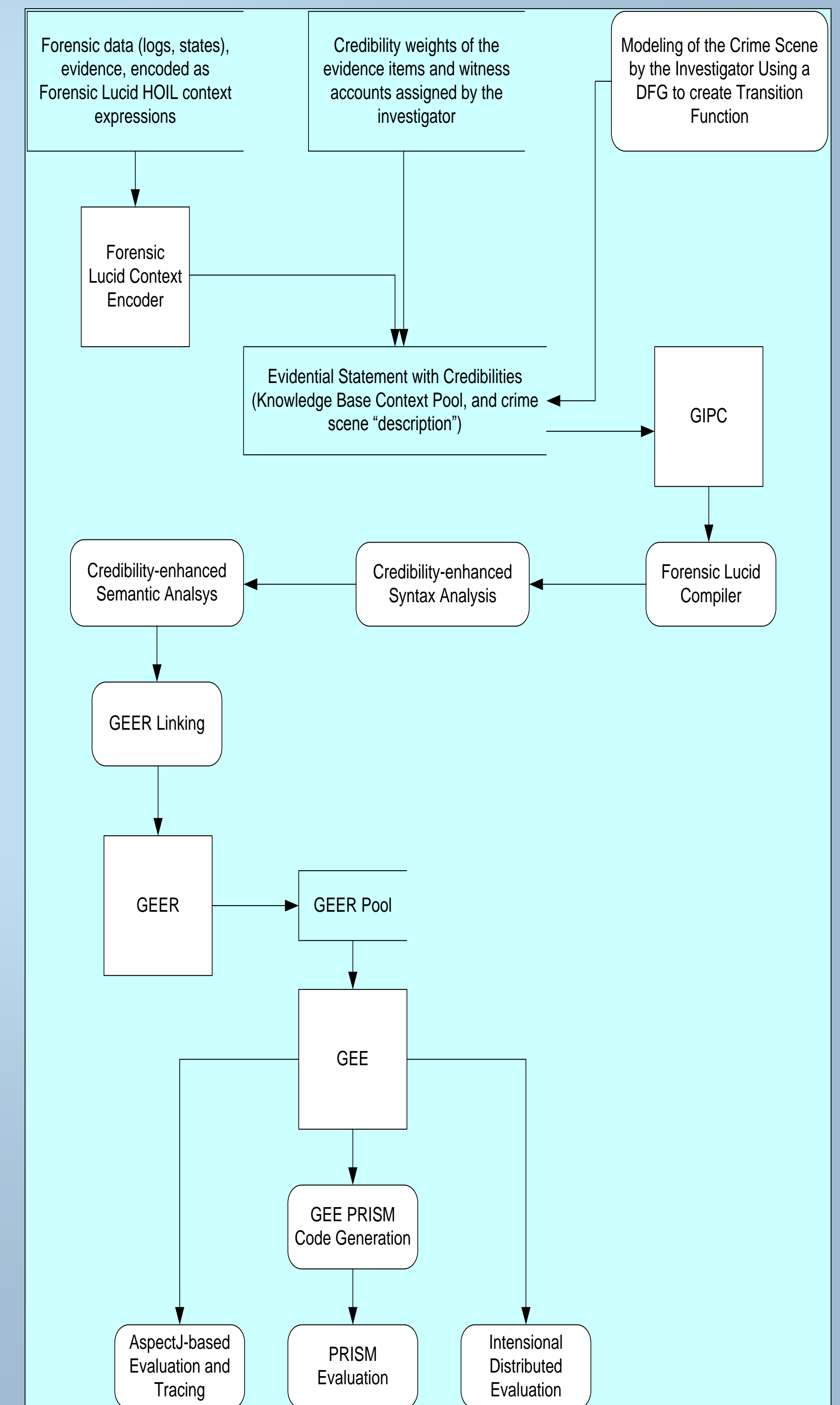
- framework for formal specification and code generation of autonomic systems (ASs);
- comprises a special formal notation and a toolset including tools that allow specifications to be edited and validated;
- considers ASs as composed of autonomic elements (AEs) communicating over interaction protocols;
- defined through the formalization of tiers.

AS	AS service-level objectives	
	AS self-management policies	
	AS architecture	
	AS actions	
	AS events	
ASIP	AS messages	
	AS channels	
	AS functions	
	AS metrics	
AE	AE service-level objectives	
	AE self-management policies	
	AE friends	
	AEIP	AE messages
		AE channels
		AE functions
		AE managed elements
	AE recovery protocols	
	AE behavior models	
	AE outcomes	
AE actions		
AE events		
AE metrics		

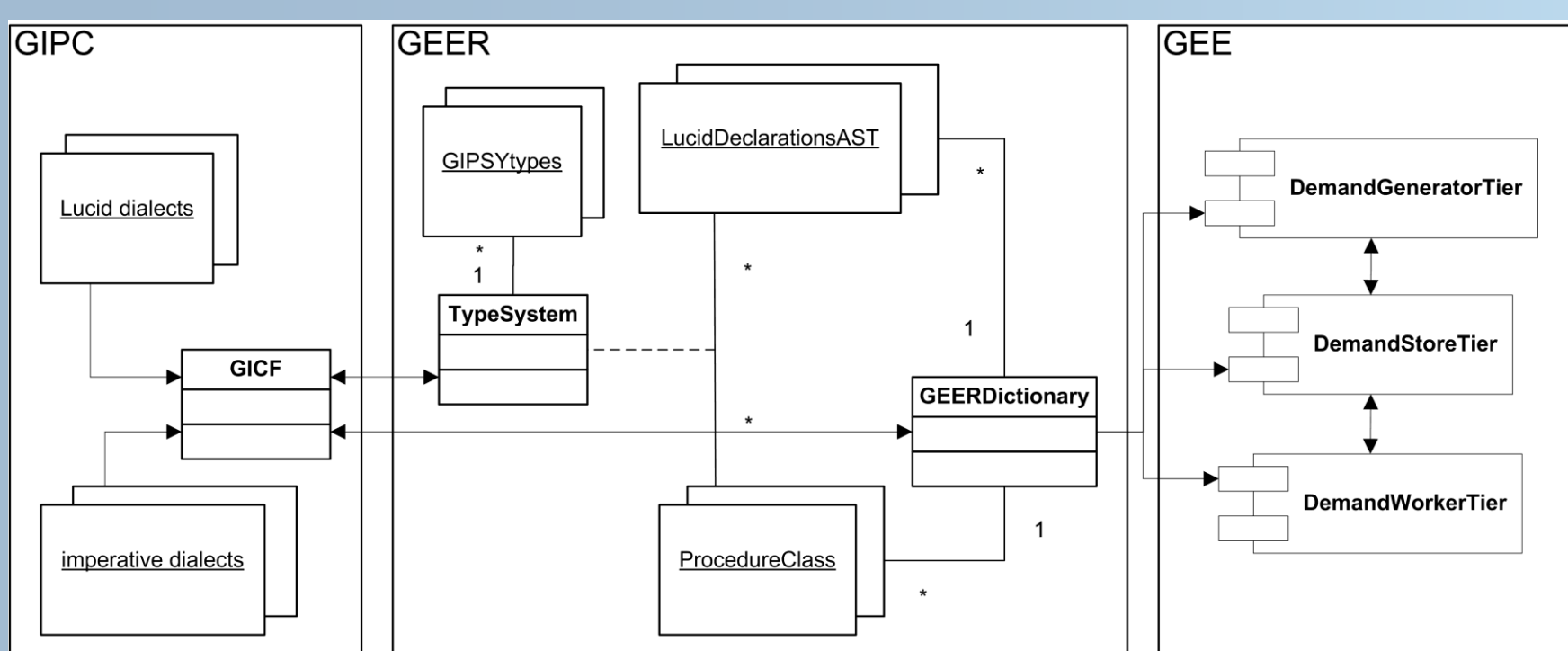
## ASSL TIERS

- AS tier - forms a general and global AS perspective, where we define the general system rules in terms of service-level objectives (SLO) and self-management policies, architecture topology, and global actions, events, and metrics applied in these rules.
- AS Interaction Protocol (ASIP) tier – forms a communication protocol perspective, where we define the means of communication between AEs.
- AE tier - forms a unit-level perspective, where we define interacting sets of individual AEs with their own behavior.

## Forensic Lucid+JOOIP Compile and Run



## GIPSY



## RESEARCH PROBLEM

Implement the self-forensics autonomic property for autonomic software systems for self-management and evidence gathering, analysis, reaction, and event reconstruction of incident handling.

## ASSL SELF-FORENSICS EXAMPLE MODEL

```

AS ADMARF {
  TYPES { MonitoredElement }
  ASSELF_MANAGEMENT {
    SELF_FORENSICS {
      FLUENT inIntensiveForensicLogging {
        INITIATED_BY { EVENTS.anomalyDetected }
        TERMINATED_BY {
          EVENTS.anomalyResolved,
          EVENTS.anomalyFailedToResolve
        }
      }
      MAPPING {
        CONDITIONS { inIntensiveForensicLogging }
        DO_ACTIONS { ACTIONS.startForensicLogging }
      }
    }
    ACTIONS {
      ACTION startForensicLogging {
        GUARDS { ASSELF_MANAGEMENT.SELF_FORENSICS.inIntensiveForensicLogging }
        VARS { Boolean ... }
        DOES {
          FOREACH member in AES {
            ONERR_DOES {
              // if error then log it too
            }
          }
        }
      }
    }
    EVENTS { // these events are used in the fluents specification
      EVENT anomalyDetected {
        ACTIVATION { SENT { ASIP.MESSAGES... } }
      }
    }
    METRICS {
      METRIC thereIsInsecurePublicMessage {
        METRIC_TYPE { CREDIBILITY }
        DESCRIPTION { "sets event's trustworthiness/credibility AE" }
        VALUE { ... }
      }
    }
  }
}
// ...
MANAGED_ELEMENTS {
  MANAGED_ELEMENT STAGE_ME {
    INTERFACE_FUNCTION logForensicEvent {
      PARAMETERS { ForensicLucidEvent pcEvent }
      RETURNS { Boolean }
    }
  }
}

```

Figure 4: The Prototype Syntactical Specification of the SELF\_FORENSICS in ASSL for ADMARF