Construction of an EMR Testbed

Janos Mathe, Sean Duncavage, Jan Werner, Akos Ledeczi, Bradley Malin, Janos Sztipanovits

Institute for Software Integrated Systems, Vanderbilt University

Goal

Create a tool suite capable of addressing the following challenges of Clinical Information Systems (CIS)

- Adaptation to changing requirements
- Privacy and security concerns
- Integration of legacy systems

Our approach allows for the rapid prototyping of experimental CIS by providing

- a design and execution environment and
- security and privacy analysis

Approach

Facilitating existing tools

1) SOA standards (from the OASIS standards family)
   - BPEL
   - XACML
2) Other related standards
   - WSDL
   - XSD
3) SOA execution platforms
   - OracleBPEL
   - AciveBPEL
   - Apache ODE
4) Policy execution engine
   - SunXACML
   - Prolog
5) Modeling abstractions, language creation
   - Generic Modeling Environment (GME)
6) Model translation
   - GREAT

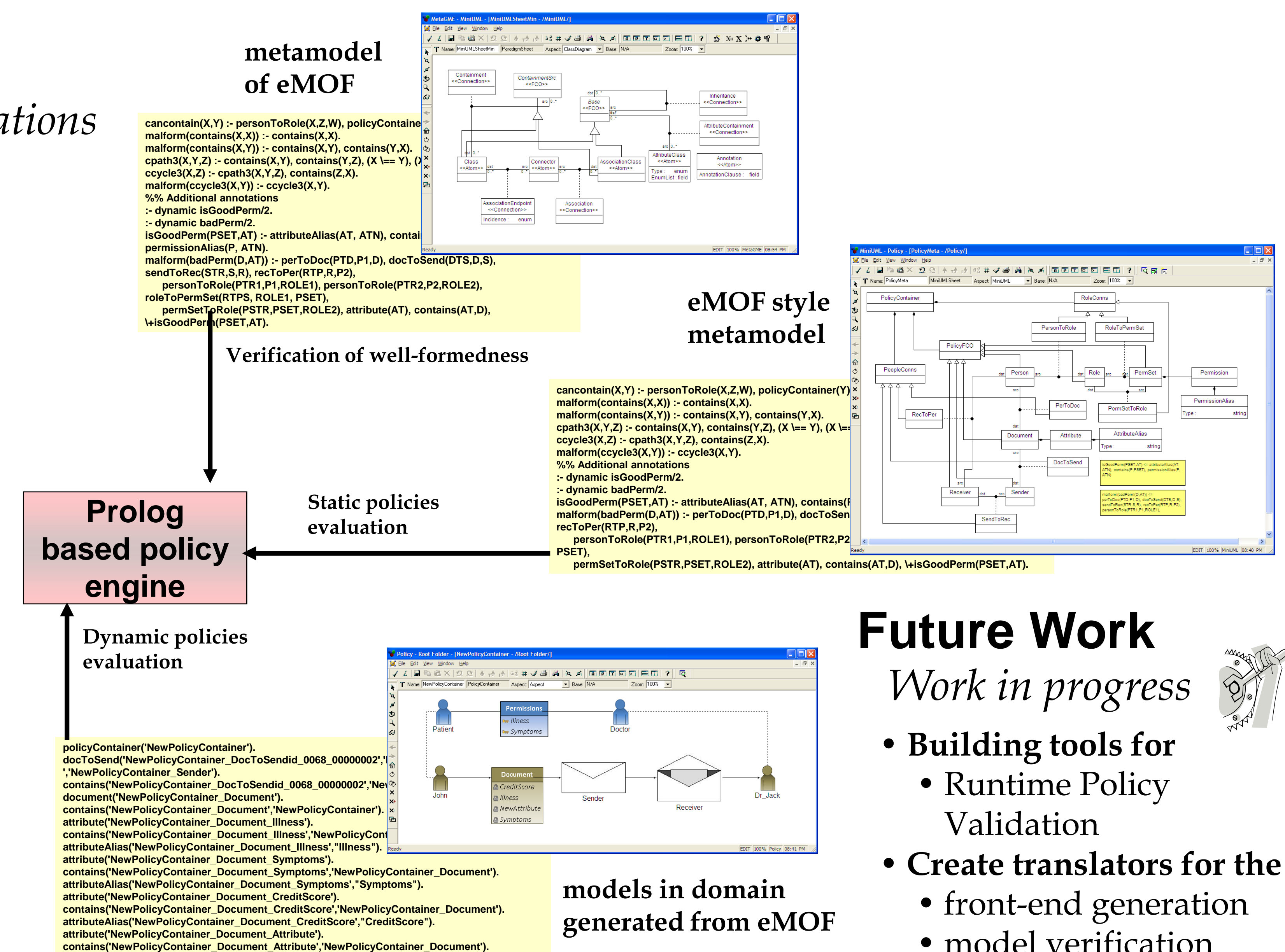*From language design to workflow execution*

1. Using Model Based Design (MBD) we express domain specific modeling abstractions as formally defined Domain Specific Modeling Languages (DSML)
2. Configuration of Generic Modeling Environment (GME) based on DSML to be able to build domain specific models
3. Built models are translated to Service Oriented Architectures (SOA) standards like BPEL, WSDL, XACML .
4. Translated models can be used to drive an execution engine / platform
5. The models can also be translated to used in a verification or simulation system.
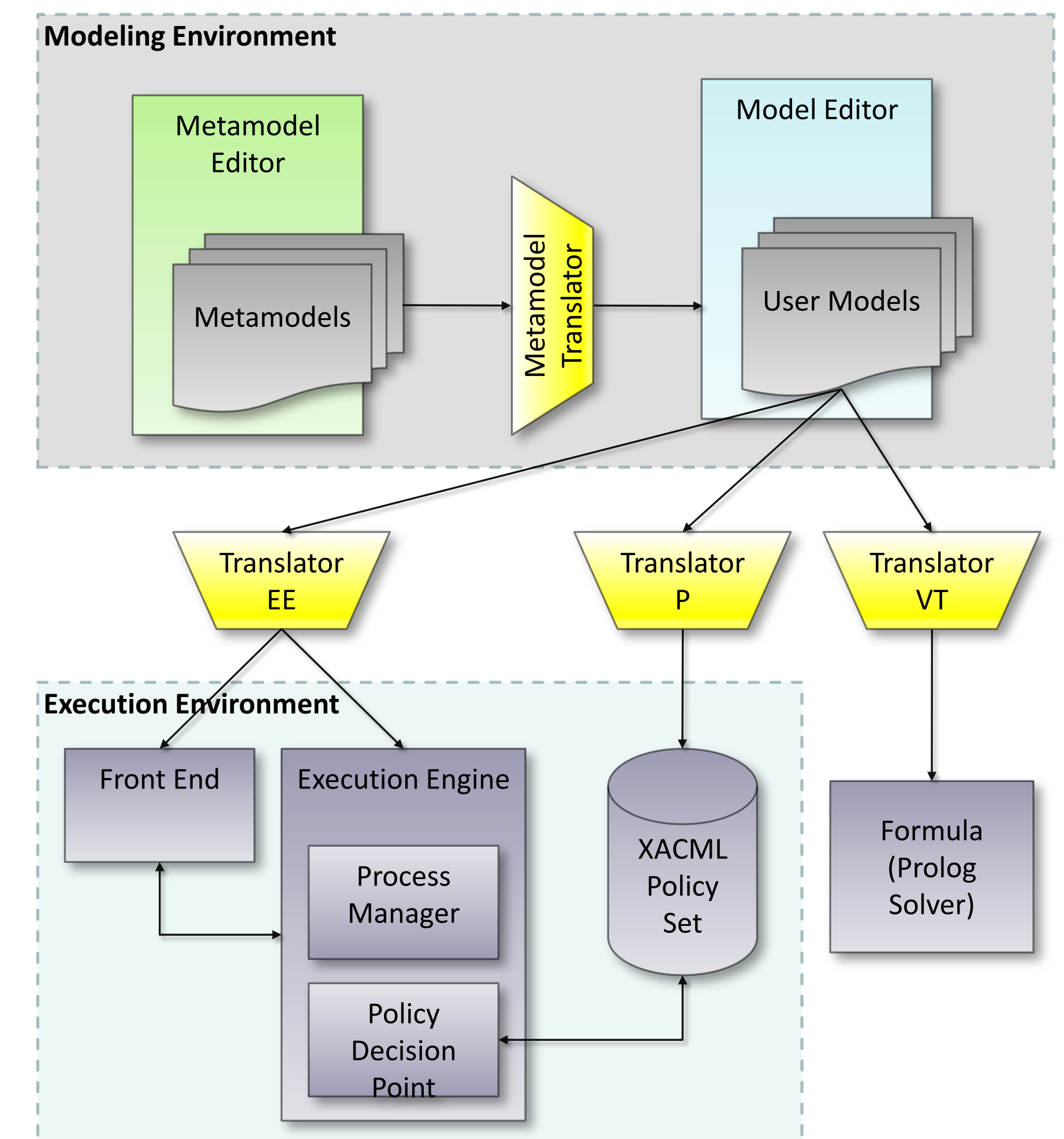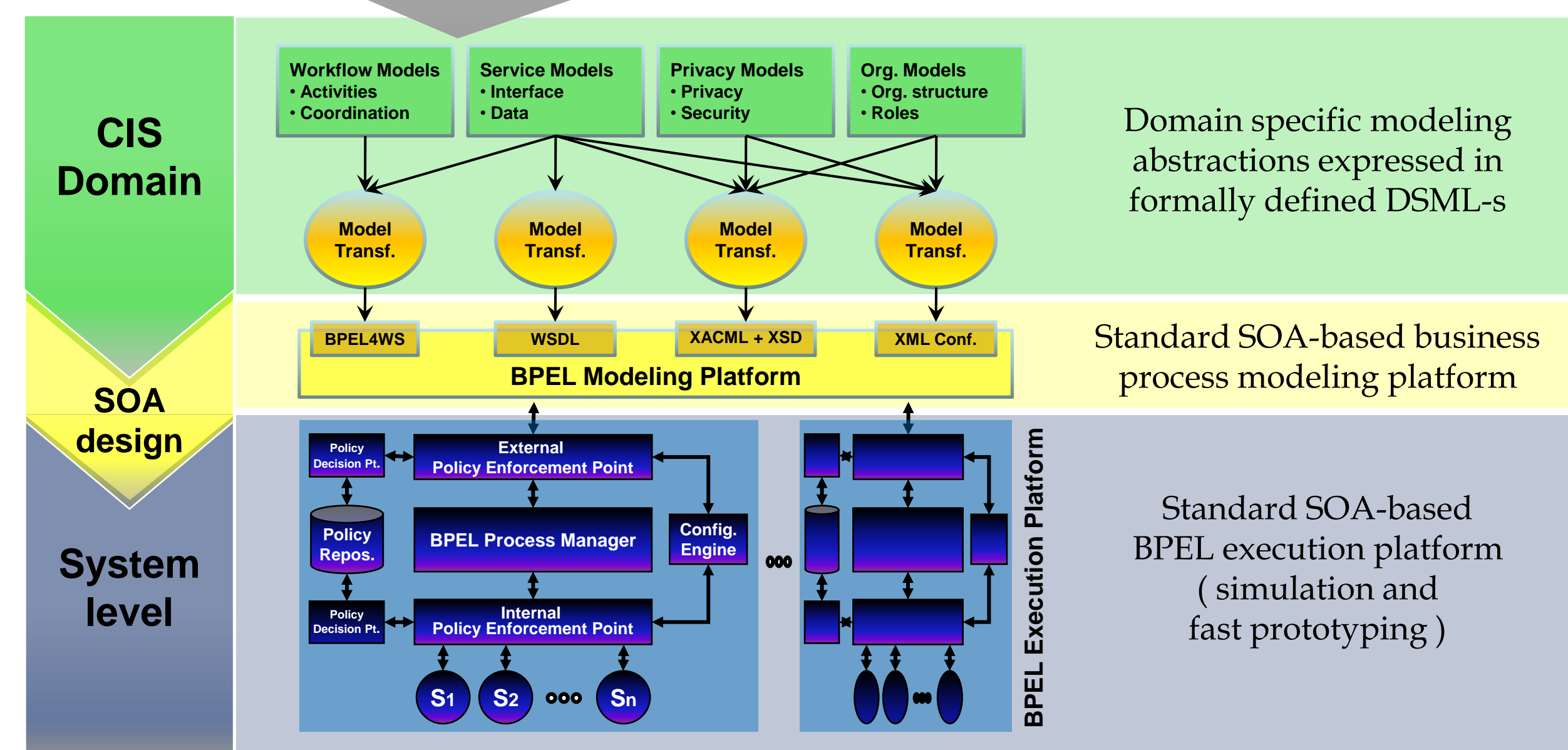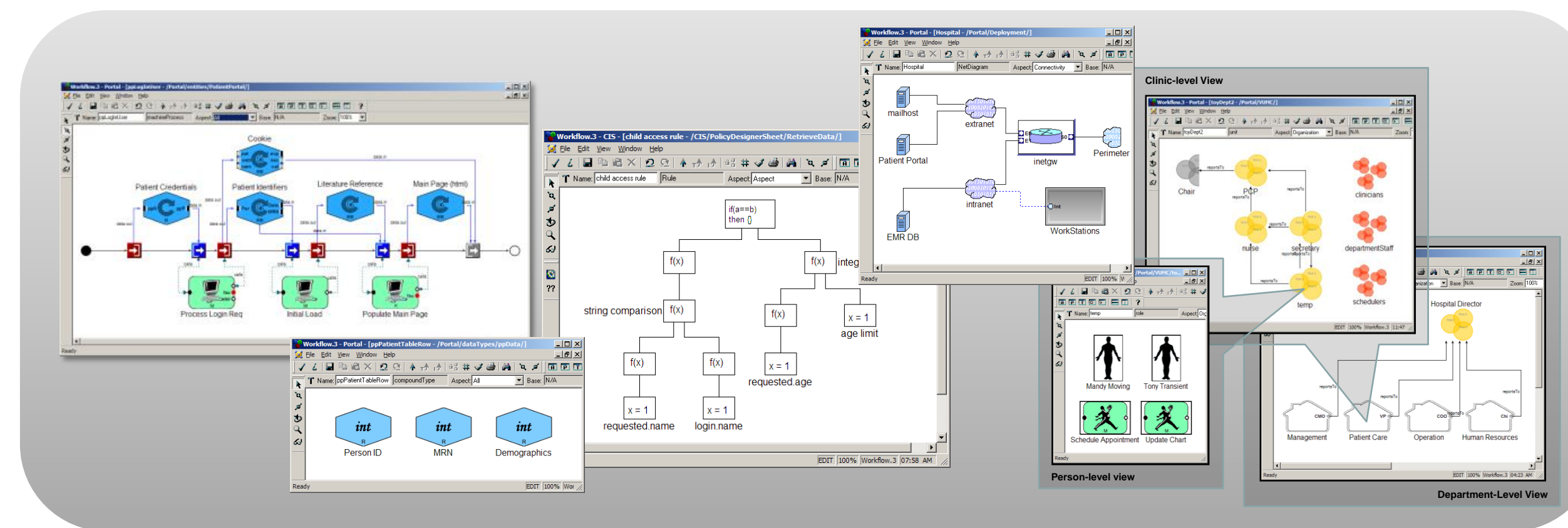
Policy modeling

Static and dynamic policy representations

1. Current approach
   MICIS supports static policies using OCL model constraints and dynamic policies using XACML. This setup allows one to build privacy policies for CIS systems.

2. Suggested approach
   Structural Semantics allows one to represent static policies as metamodel annotations and dynamic policies as model annotations that are translated to set of Prolog rules. The advantage of this approach is that the events are recorded by the decision point, and future access is based on history of access events.

metamodel of eMOF

Verification of well-formedness

eMOF style metamodel

Static policies evaluation

Prolog based policy engine

Dynamic policies evaluation

models in domain generated from eMOF

Future Work

Work in progress

- **Building tools for**
  - Runtime Policy Validation
- **Create translators for the**
  - front-end generation
  - model verification