Model-Integrated Computing

Janos Sztipanovits
ISIS, Vanderbilt University
janos.sztipanovits@vanderbilt.edu
ISIS Parameters

- Established by the School of Engineering at Vanderbilt University in 1998
- Academic/professional research organization
- Composition (housed in 3 buildings):
  - 35 Research Scientists & Staff Engineers
  - 7 Faculty
  - 5 Admin Staff
  - 50 Graduate students
- ~$11,000,000 in FY04 research awards
Research Thrusts

- **Model-Integrated Computing (MIC)**
  - Modeling, model transformation, model analysis and code generation tools for MDA. Semantic foundations for embedded & hybrid systems. Open tool integration platform for MDA.
- **Middleware for Distributed Real-time & Embedded Systems**
  - Adaptive & reflective middleware, model-based integration technology above component models, Real-time CORBA (ACE/TAO), Model Driven Architectures, secure middleware
- **Model-Based Systems Applications**
  - Fault management, distributed control, automotive, avionics
- **Secure Networked Embedded System**
  - Shooter location, security in system context
ISIS Industry Partners
ISIS Academic Partners
MIC Milestones 90's

1990
- IPCS/DuPont
  - Activity Modeling Language
  - XGEM
    - Developed: 1990-1993
    - Deployed in 1993
    - Used in Control Room
- Domain Specific Modeling Languages
  - Basic Model Translators
  - Composition platform: MGK
  - OODB

1995
- International Space Station
  - FDIR Modeling Tool
    - physical
    - functional
  - Diagnosability Analysis Tool
  - Diagnostic System
- GM-Saturn
  - Developed: 1995-1996
  - Deployed in 1996
  - Production use in 2 plants
  - Model-Based System Integration Tool
  - Data Servers
  - Bottleneck Tool
  - Process Viewer
  - Multiple aspect modeling
  - Metaprogrammable model builder: XGEM
  - Model migration

1999
- DuPont Chemicals
  - Activity Modeling Tool
    - process models
    - activity models
  - Diagnostic System
- SSPF/SATURN
  - MS Platform
  - UML-based Metamodelling
  - GME
  - Activity Modeling Language
  - Developed: 1995-1996
  - Deployed in 1996
  - Production use in 2 plants
- Domain Specific Modeling Languages
  - Basic Model Translators
  - Composition platform: MGK
  - OODB
- Adaptive Target Rec.
  - Data Flow
  - Hardware Resources
  - Behavior
  - Developed: 1997-2000
  - HW/SW Co-design
  - VHDL generation
  - Embeddable “Active” Models
  - Generative Modeling
  - GME x
  - Design space modeling
  - UML-based Metamodelling
  - GME
MIC Milestones 90's

1990
- Activity Modeling Language
  - DuPont Chemicals
  - Developed: 1990-1993
  - Deployed in 1993
  - Used in Control Room
- FDIR Modeling Tool
  - Physical
  - Functional
- Diagnosability Analysis Tool
- Diagnostic System
- SSPF/SATURN MS Platform
- UML-based Metamodeling
- GME

1995
- International Space Station
  - Developed: 1992-1994
  - Deployed in 1995
  - Used program wide
- GM-Saturn
  - Developed: 1993-1996
  - Production use in 2 plants
- Model-Based System Integration Tool
- Data Servers
- Fault Trees Tool
- Process Viewer
- Multiple aspect modeling
- Metaprogrammable model builder: XGEM
- Model migration

1999
- Design space modeling
- Embedded Models
  - Developed: 1997-2000
  - HW/SW Co-design
  - VHDL generation
- SSPF/SATURN MS Platform
- UML-based Metamodeling
- GME
- Embeddable "Active" Models
- Generative Modeling
- GME x
- Design space modeling
MIC Milestones 90's

1990
- Activity Modeling Language
  - DuPont Chemicals
  - Developed: 1990-1993
  - Deployed in 1993
  - Used in Control Room
- IPCS/DuPont
  - XGEM
  - DTool/Boeing

1995
- International Space Station
  - Developed: 1992-1994
  - Deployed in 1995
  - Used program wide
- MIC Alliance
  - SSPF/SATURN
    - MS Platform
  - UML-based Metamodeling
  - GME

1999
- Design space modeling
  - Embedded Models
- GM-Saturn
  - Model-Based System Integration Tool
    - Developed: 1995-1996
    - Deployed in 1996
    - Production use in 2 plants
- Data Servers
- Bottleneck Tool
- Process Viewer
- Simulator
- SW Generator
- VHDL Generator

- Domain Specific Modeling Languages
- Basic Model Translators
- Composition platform: MGK
- OODB
- Multiple aspect modeling
- Metaprogrammable model builder: XGEM
- Model migration
- Metamodeling
- Model-based distributed system integration
- Multiple-target generators
- Embeddable "Active" Models
- Generative Modeling
- GME x
- Design space modeling
MIC Milestones 90's

1990
- Activity Modeling Language
- Developed: 1990-1993
- Deployed in 1993
- Used in Control Room
- DuPont Chemicals

1995
- IPCS/DuPont
- XGEM
- DTool/Boeing
- MIC Alliance
- SSPF/SATURN
- MS Platform
- UML-based Metamodelling
- GME

1999
- Design space modeling
- Embedded Models
- Adaptive Target Rec.
  - Data Flow
  - Hardware Resources
  - Behavior
- Developed: 1997-2000
- HW/SW Co-design
- VHDL generation

- Domain Specific Modeling Languages
- Basic Model Translators
- Composition platform: MGK
- OODB

- Multiple aspect modeling
- Metaprogrammable model builder: XGEM
- Model migration

- Metamodelling
- Model-based distributed system integration
- Multiple-target generators

- Embeddable “Active” Models
- Generative Modeling
- GME x
- Design space modeling
MIC Milestones 00’s

2000
- GME 2000
- OTIF
- GRATIS
- DOC Group joins
- OMG MIC PSIG
- ESCHER
- GReAT
- COSMIC
- Semantic Anchoring
- FCS Boeing
- NSF TRUST
- System/Security Codesign

2005
- DARPA
- Applications
  - Time Synch Services
  - Coordination Services
  - Localization Services
  - Berkeley Motes/TINY-OS.
- NSF-ITR
- Future
  - NSF TRUST S&TC
  - DARPA, DoD
  - Industry Funding
- MIC Infrastructure Research:
  - Semantic anchoring of DSML-s (embedded systems)
  - Model transformations
  - Model synthesis
  - Active models
  - Metaprogrammable tool designs
- New MIC Applications:
  - Network-centric systems
  - Security-systems co-design in embedded systems
  - Systems biology
  - Automotive, avionics, unmanned systems applications

- $ 50M DARPA Program
  (Sztipanovits, Bay)
- Model Transformation Technology
- Open Tool Integration Framework
- Open Experimental Platforms

- $ 45M DARPA Program
  (Sztipanovits, Raghavan)
- Fine-grain distributed system
- Service composition
- Exploration of new apps.
  (shooter location)

- $ 13M Large ITR
  UC Berkeley/ISIS-VU
- Hybrid Systems
- Model-based design
- Tool Architectures
- Experimental Systems

MIC Infrastructure Research:
- Semantic anchoring of DSML-s (embedded systems)
- Model transformations
- Model synthesis
- Active models
- Metaprogrammable tool designs

New MIC Applications:
- Network-centric systems
- Security-systems co-design in embedded systems
- Systems biology
- Automotive, avionics, unmanned systems applications
MIC Goal

- Building increasingly complex networked systems from components
  - Naïve “plug-and-play” approach does not work in embedded systems (neither in larger non-embedded systems)
  - We need to model, analyze, and integrate the components using models
- Aligned with the goals of MDA
Models - Models - Models

Typical System Layers

- HCI
- Applications
- Services
- Communication and Computing
- Sensing

Operational models (OV)
  - Context Models
  - User Models
  - Operational Tasks Models
  - Application Mapping

Application Models (SV)
  - Functional models
  - System Modalities
  - Service Mapping Models

Service Models
  - Application Services
  - Integration Services
  - Communication Services
  - Resource mapping

Computing & Comm.
  - OS Abstraction
  - Hardware Architecture
  - Communication Models

Sensor Models
  - Fusion Models
  - Exploitation Models

Use of Models:
- Analysis
- Architecture Exploration
Two Dimensions of Model-Based Design

System Composition

- Heterogeneous
- Distributed
- Embedded
- Layered

Tool Composition

- Composable
- Integrated
- Correct by construction

ESCHER is non-profit Research Institute maintaining open-source, quality controlled repository of tools and software for Model-based Design.
/http:www.escherinstitute.org/
# System Composition Layers

<table>
<thead>
<tr>
<th>Component Behavior</th>
<th>Ptolemy II</th>
<th>Giotto</th>
<th>IF</th>
</tr>
</thead>
<tbody>
<tr>
<td>Java Code/Behavioral Models</td>
<td>Hierarchical Module Interconnection</td>
<td>Hierarchical Module Interconnection</td>
<td>Hierarchical Timed Automaton</td>
</tr>
<tr>
<td>Heterogeneous Models of Computation + Directors</td>
<td>-Interface Theory: -Resource Interfaces</td>
<td>-Giotto: TT Static Periodic Schedule</td>
<td>Asynchronous Interactions: - P2P - Unicast - Multicast</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Dynamic Priorities</td>
</tr>
</tbody>
</table>

- P2P  
- Unicast  
- Multicast
<table>
<thead>
<tr>
<th>Tool Composition Layers</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Domain-Specific Tools; Design Environments</strong></td>
</tr>
<tr>
<td><strong>Metaprogrammable Tools, Integration Environments</strong></td>
</tr>
<tr>
<td><strong>Semantic Foundation</strong></td>
</tr>
<tr>
<td><strong>Prototype Tool Chains (Software Factories) (work in progress):</strong></td>
</tr>
<tr>
<td>- ECSL - Automotive</td>
</tr>
<tr>
<td>- ESML - Avionics</td>
</tr>
<tr>
<td>- SPML - Signal Processing</td>
</tr>
<tr>
<td>- CAPE/eLMS</td>
</tr>
<tr>
<td>- COSMIC - ACE-TAO Components</td>
</tr>
<tr>
<td><strong>MIC Metaprogrammable Tool Suites: (mature or in maturation program)</strong></td>
</tr>
<tr>
<td>- GME (Generic Model Editor)</td>
</tr>
<tr>
<td>- GReAT (Model Transformation)</td>
</tr>
<tr>
<td>- OTIF (Tool Integration Framework)</td>
</tr>
<tr>
<td>- UDM (Universal Data Model)</td>
</tr>
<tr>
<td>- DESERT (Design Space Exploration)</td>
</tr>
<tr>
<td>- GME-MOF/Meta (Metamodeling Env-s)</td>
</tr>
<tr>
<td><strong>MIC Foundations (work in progress):</strong></td>
</tr>
<tr>
<td>- Semantic Anchoring Environment (SAE)</td>
</tr>
<tr>
<td>- Architecture Exploration Platform (AEP)</td>
</tr>
</tbody>
</table>
## Interrelations

<table>
<thead>
<tr>
<th>Component Behavior</th>
<th>Structure</th>
<th>Interaction</th>
<th>Resource Modeling (Schedule)</th>
</tr>
</thead>
<tbody>
<tr>
<td>- State Automaton</td>
<td>- Set-Valued Semantics</td>
<td>- Tagged Signal Model</td>
<td>- Transition Systems With Priority</td>
</tr>
<tr>
<td>- Timed Automaton</td>
<td></td>
<td>- State Automaton</td>
<td></td>
</tr>
<tr>
<td>- Hybrid Automaton</td>
<td></td>
<td>- Timed Automaton</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- ...</td>
<td></td>
</tr>
</tbody>
</table>

|-------------------------------------|-------------------------------------|-------------------------------------|-------------------------------------|

**TOOLS**

**COMPOSITION PLATFORMS**

**METAPROGRAMMABLE TOOLS**

**Domain-Specific Tools, Tool Chains**

**Metaprogrammable Tools, Environments**

**Semantic Foundation;**
doTransition (fsm as FSM, s as State, t as Transition) =
require s.active
step exitState (s)
step if t.outputEvent <> null then
  emitEvent (fsm, t.outputEvent)
step activateState (fsm, t.dst)

Semantic Foundation Libraries

Domain-Specific Environments

Prototype Tool Chains (Software Factories) (work in progress):
- ECSL - Automotive
- ESML - Avionics
- SPML - Signal Processing
- CAPE/eLMS
- COSMIC - ACE-TAO Components

MIC Metaprogrammable Tool Suites: (mature or in maturation program)
- GME (Generic Model Editor)
- GReAT (Model Transformation)
- OTIF (Tool Integration Framework)
- UDM (Universal Data Model)
- DESERT (Design Space Exploration)
- GME-MOF/Meta (Metamodeling Env-s)

MIC Foundations (work in progress):
- Semantic Anchoring Environment (SAE)
- Architecture Exploration Platform (AEP)
Results:

MIC Metaprogrammable Tool Suite

Meta-Model of StateFlow using uml/OCL as meta modeling language.

GME, UDM, GREAT, DESERT
Completed tool suite, available through the ESCHER Repository
Mission Control Platform (MCP)
Example - 1

Common Semantic Domain
Abstract Syntax: Meta-Models
Domain Models and Tool Interchange Formats

Mission Control Platform Tool Chain

Component Model
Component Structure
Thread Models
Component Interaction
Schedulability Analysis
System Integration
Implementing Design Flows: Modeling and Transformations

- Large influence of concrete syntax
- No clear role of semantics
- It is not clear what are we doing?
Abstract Syntax Metamodelling

- Gives structural semantics for the models
- Set-valued Semantic Domain for the metamodels and transformations
MCP Example - 2

Common Semantic Domain

Abstract Syntax and Transformations: Meta-Models

Domain Models and Tool Interchange Formats

Mission Control Platform Tool Chain

PRISM Meta-Model → PRISM → ESML → ECSL-DP Meta-Model → ESML → AIF → ESML → CFG → CFG Meta-Model
Transformational Specification of Behavioral Semantics

\[ M_{S1} = M_{S2} \circ M_{12} \]
Semantic Anchoring

Semantic Anchoring of DSML-s

- The “Semantic Units” are selected abstract semantics such as MoC-s
- DSML-s or their aspects are anchored to the MoC-s using transformations
- The “Semantic Units” are specified in a formal framework
Architecture Exploration

“Configuration”

Domain-Specific Tool Chain

“Calibration”

Tool and Translator Metaprogramming

Basic Semantic Specifications

Concrete Syntax Free Integrated Models

Component Modeling Tools

Structure Modeling Tools

Interaction Modeling Tools

Resource Modeling Tools

Models DSML-1

Models DSML-2

Models DSML-3

Models DSML-4

Models DSML-5

Models DSML-6

Models DSML-7

Models DSML-8

Models DSML-9

Behavior Analysis

Guided Exploration

Prob. Model Checking

Temp. Model Checking

Model Checking
Teaching and Training

• Industrial:
  - Escher Institute; training programs for major end-user companies (GM, Boeing, Raytheon)

• Academic:
  - MIC course (6th year at VU, 2nd at Berkeley Students design DSML-s and model-based generators)
  - Summer program for undergraduates (SIPHER at VU, SUPERB at Berkeley)
  - Migrating down to undergraduate level is examined
Industry Interactions

- MIC PSIG at OMG (www.omg.org/mic)
- Many projects with a number of large companies (Raytheon, Boeing, GM, DuPont, Lockheed, SwRI and others)
- Direct participation in large national programs with industry (FCS, Space Exploration, etc.)
MIC and MDA

• Same general objective
• Different emphasis (MIC has a stronger systems component)
• Open source metaprogrammable tool suite of MIC has long maturation and application experience
• Increasing conceptual convergence
  - Metamodeling
  - Use of DSML instead of a Single Unified Language for Everything
  - Strong role of model transformations