MAST Modeling and Analysis Suite for Real-Time Applications

Grupo de Computadores y Tiempo Real. Universidad de Cantabria

http://www.ctr.unican.es

MAST MODELING

The real time model is built up using primitive components defined in the MAST suite. At present, it defines the root abstract classes and a number of concrete ones derived from them. MAST is open to new concretes classes required by new analysis methods and tools.

Main abstract modeling components defined in MAST:
- **Real-Time Situation**: Execution mode of the system that provides the context for the analysis.
- **Transaction**: Sequences of activities that may be executed in response to a pattern of events.
- **Activity**: It models the execution of a piece of code or the transmission of a message.
- **External event**: It models the stimulus from external devices or timers.
- **Timing requirement**: It models a requirement imposed on the instant of finalization of an activity.
- **Processing Resource**: It models an active resource (processor or network).
- **Scheduling Server**: It models a schedulable entity hosted in a processing resource. It has a scheduling policy assigned.
- **Operation**: It models the amount of processing capacity that is required for executing a piece of code or transmitting a message.
- **Shared Resource**: It models a passive-protected resource.

PROFILES

They offer higher levels of modeling abstraction for some specific software methodologies or environments. Each profile defines a lot of new modeling components that implement concepts and artifacts introduced by the semantics of the associated software methodology.

At present three profiles have been defined:
- **UML_MAST**: Profile for modeling real-time systems that are designed using object-oriented representation techniques and UML-CASE tools.
- **ADA_MAST**: Profile for modeling real-time distributed applications, built with Ada 95 platform that satisfy both, Annex D and Annex E of the Ada 95 standard.
- **CBSE_MAST**: Profile for modeling real-time systems based on software components (CBSE).

MAST _FILE

The real-time MAST model is described through a formatted ASCII description that serves as the input to the analysis tools.

In order to facilitate the integration of new tools, the MAST suite offers an interface with:
- A parser that converts the ASCII description of the system into a data structure that is accessible from the tools.
- An open repository and its interface.

ANALYSIS AND DESIGN TOOLS

The MAST toolset is open and extensible. Its object-oriented design and the available managing interface facilitate that other design teams enhance it with new tools.

The present version is implemented for fixed priority systems. Today, MAST handles single-processor, multiprocessor and distributed systems based on different scheduling strategies and it is able to model most real-time operating systems and languages (like POSIX and Ada).

### Analysis tools versus kind of system supported

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Classic Rate</td>
<td>X</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Monotonic</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Varying Priorities</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Holistic</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Offset Based</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Unoptimized</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Offset Based</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Optimized</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Non-Linear</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

AUTOMATED ENVIRONMENTS

**ToolDriver**: is a GUI based application that may be executed as an autonomous tool. It works directly with the ASCII files defined in MAST.

**UML MAST Framework**: is a "pseudo-Add-in" that enhances the use of a UML graphical tool by including a modeling framework with the MAST_RT_View.

This framework has been implemented on Rational Rose UML CASE tool.