Agenda

- Part 1
  - Introduction to MDD for RT/E systems & MARTE in a nutshell
- Part 2
  - Non-functional properties modeling
  - Outline of the Value Specification Language (VSL)
- Part 3
  - The timing model
- Part 4
  - A component model for RT/E
- Part 5
  - Platform modeling
- Part 6
  - Repetitive structure modeling
- Part 7
  - **Model-based analysis for RT/E**
- Part 8
  - MARTE and AADL
- Part 9
  - Conclusions

Goals in Non-Functional (or Quantitative) Analysis

It offers a mathematically-sound way to calculate NFPs of interest based on other available NFPs and the system behavior

- **Different Goals for Evaluate & Verify System Architectures**
  - Point evaluation of the output NFPs for a given operating point defined by input NFPs
  - Search over the parameter space for feasible or optimal solutions
  - Sensitivity analysis of some output results to some input parameters
  - Scalability analysis: how the system performs when the problem size or the system size grow.

MARTE Features for Quantitative Analysis

- **Improvements w.r.t. SPT**
  - Extend implementation and scheduling models
    - e.g. distributed systems, hierarchical scheduling
  - Extend the set of analysis techniques supported
    - e.g. offset-based techniques
  - Extend timing annotations expressiveness
    - Overheads (e.g. messages passing)
    - Response times (e.g. BCET & ACET)
    - Timing requirements (e.g. miss ratios and max. jitters)
- **New features w.r.t. SPT**
  - Support for sensitivity analysis
  - Improve modeling reuse and component-based design.
  - Support of the “Y-chart” approach: application vs. platform models

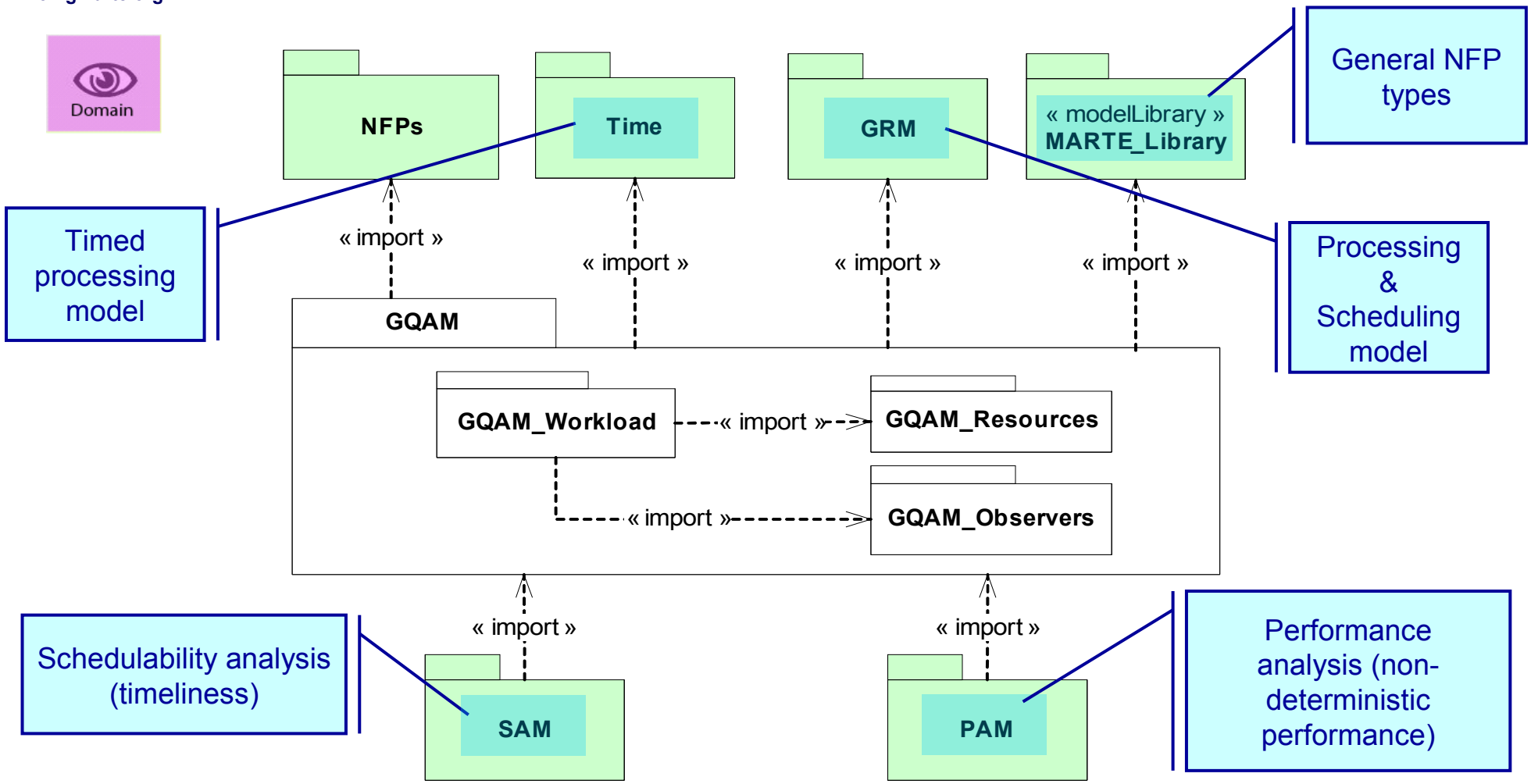
UML-Based Analysis Foundations

- **GQAM Profile factorizes common constructs and NFPs**
  - Stereotypes define "analysis" abstractions
    - workload events, scenarios,…
    - schedulable entities, shared resources, processing nodes, schedulers…
  - Stereotype attributes define pre-defined NFPs
    - e.g. event arrival patterns, end-to-end deadlines, wcet-bcet-acet,…
- **The analysis sub-profiles define model well-formedness rules**
  - It includes "constraints" to construct "analyzable" models, w.r.t…
  - "Analysis Model Viewpoints" (e.g., schedulability analysis viewpoint)
  - Specialized constraints must be refined by technique-specific approaches

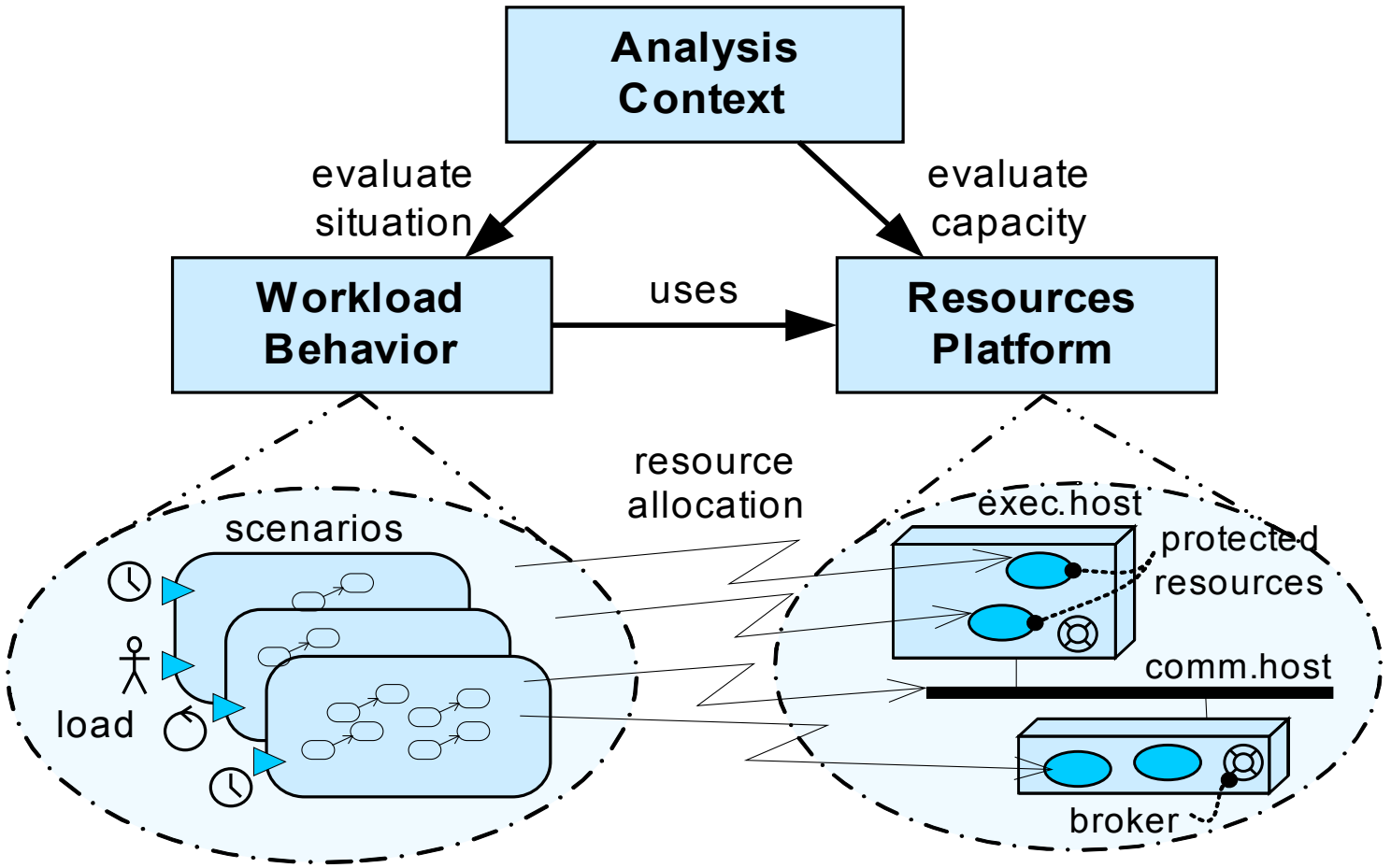
The MARTE analysis sub-profiles provide standard constructs to map UML models on well-established analysis techniques

➔MARTE "Foundations" and "GQAM" allow for extending to further techniques

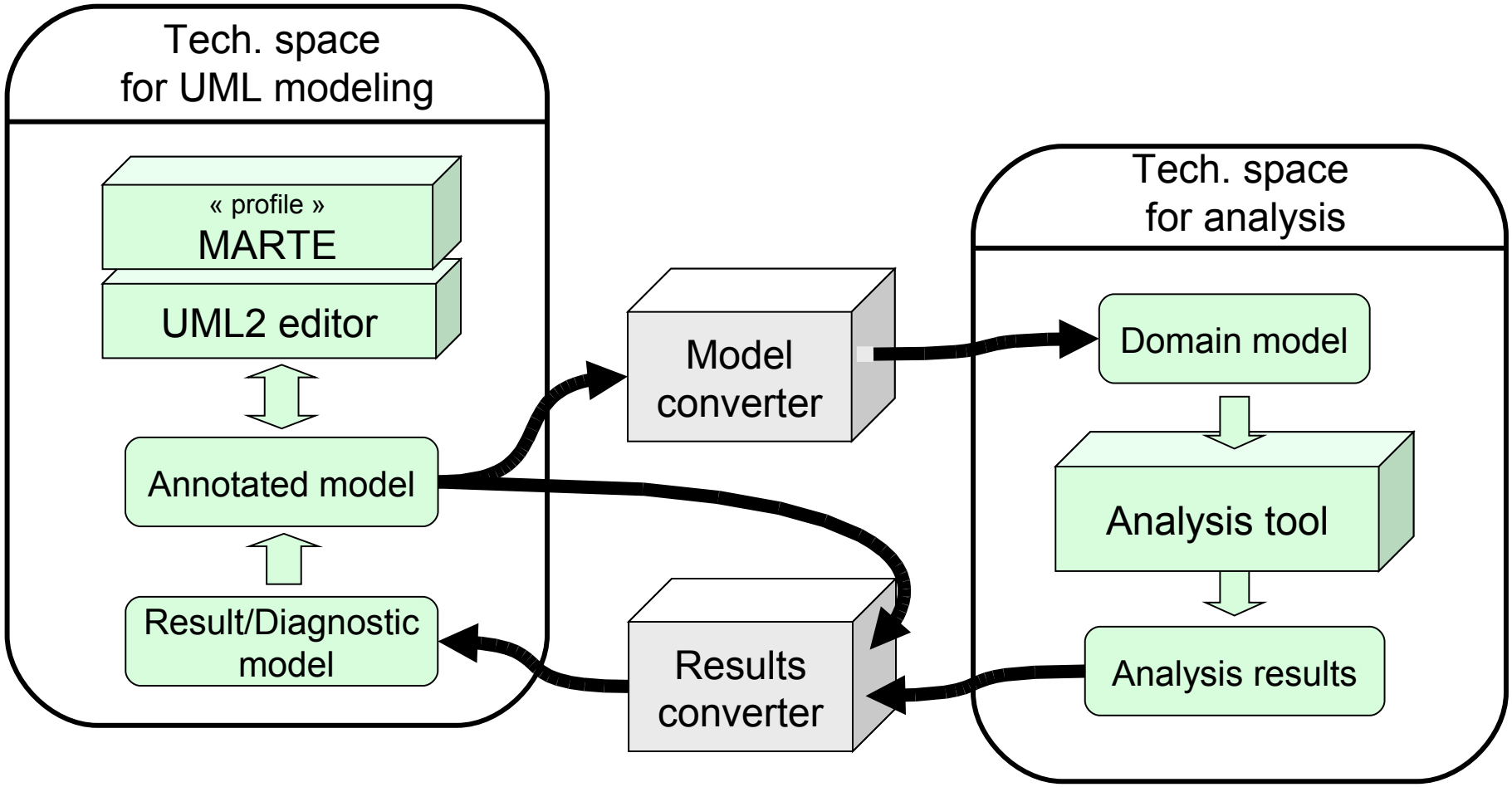
GQAM: Dependencies and Architecture



GQAM: Analysis Modeling Structure



Processing Schema for Analysis

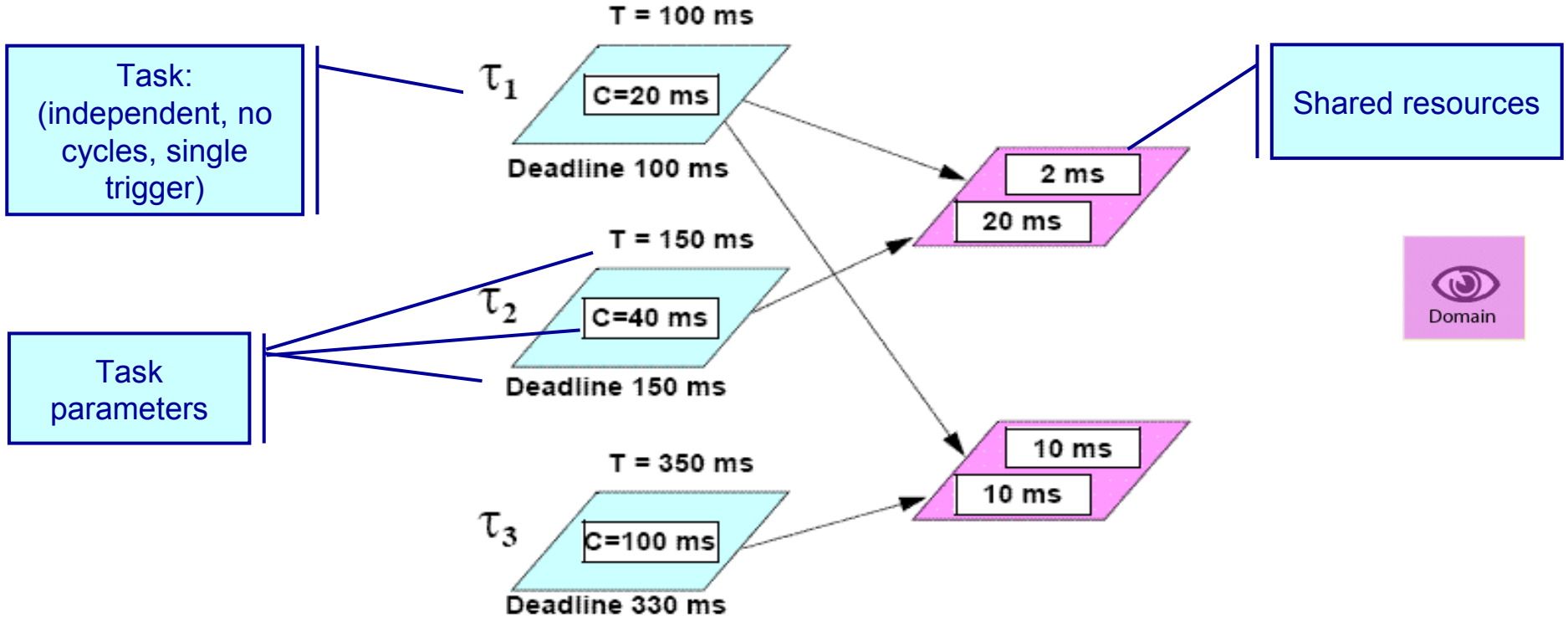


Schedulability Analysis

Provides the ability to evaluate time constraints and guarantee worst-case behavior of a system or particular piece of software

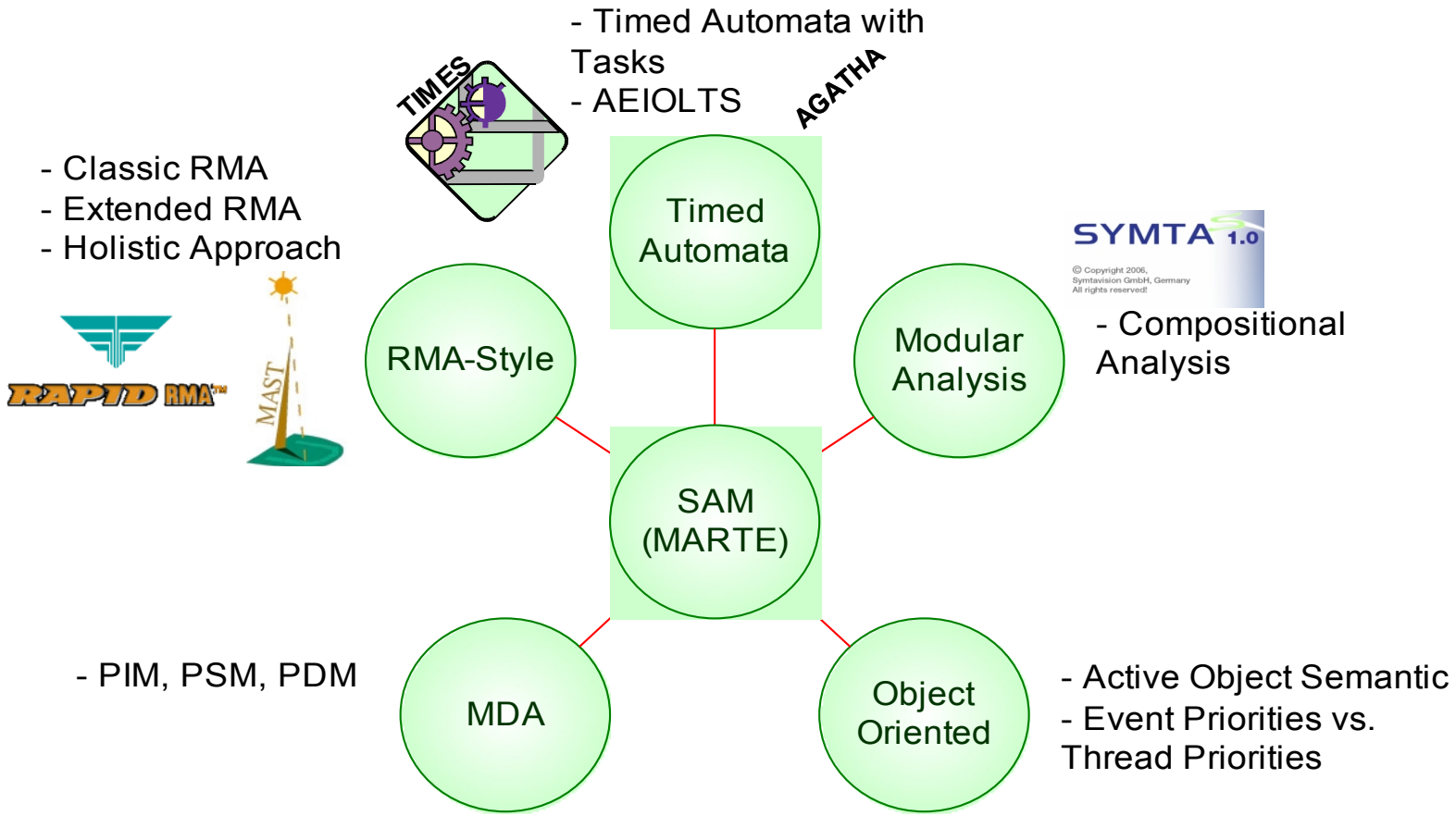
- **Schedulability analysis offers:**
  - Offline guarantees. E.g., worst-case latencies and worst-case resource usage.
  - At different development stages.
    - Early analysis: to detect potentially unfeasible real-time architectures.
    - Later analysis: to discover temporal-related faults, or to evaluate the impact of migrations (e.g., scheduling strategies).
- **Provide answer to questions such as for example...**
  - Will we miss any deadline if we switch a processor from a normal operation mode to a lower-consumption mode?
  - If yes, how can we modify task workloads for allowing our system to still work?

A Simple Example (Classical Scheduling Theory)



- **Three main analysis approaches for verify timeliness:**
  - Critical instant calculation
  - Utilization bound test
  - Response time calculation

SAM: Integration Different Approaches

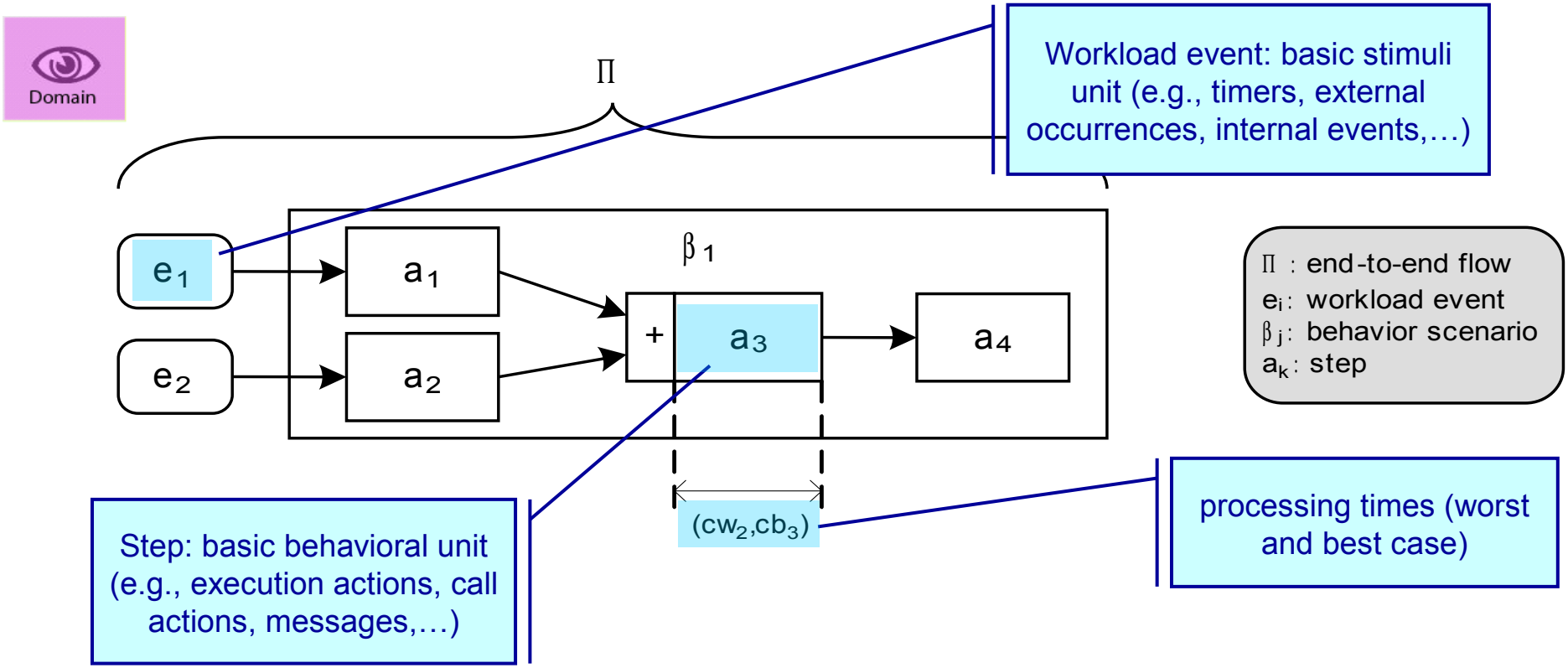


Other Sched. Analysis tools: Livedevices' Real-Time Architect, CoMET from VaST, Vector's CANAlyzer…

SAM: The Notion of End-To-End Flow

An "End-To-End Flow" is the basic workload unit to be evaluate by by schedulability analysis tools.

→ An end-to-end flow refers to the entire causal set of steps triggered by one or more external workload events.

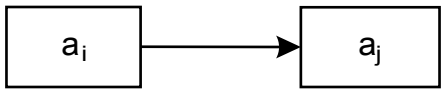


SAM: Precedence Relations

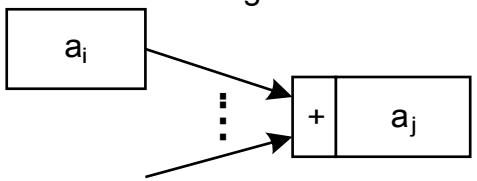
Execution and communication steps may be causally related by one of the following precedence relations:

Domain

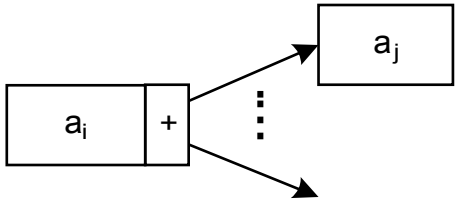
Sequencial



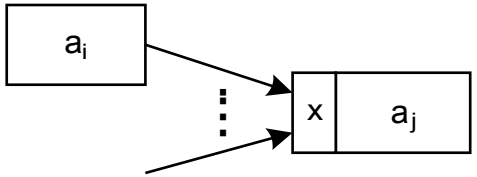
Merge OR:



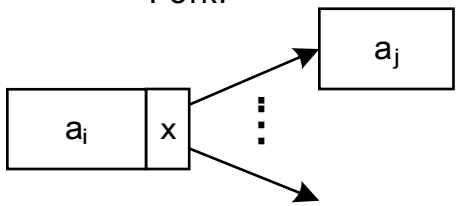
Decision OR:



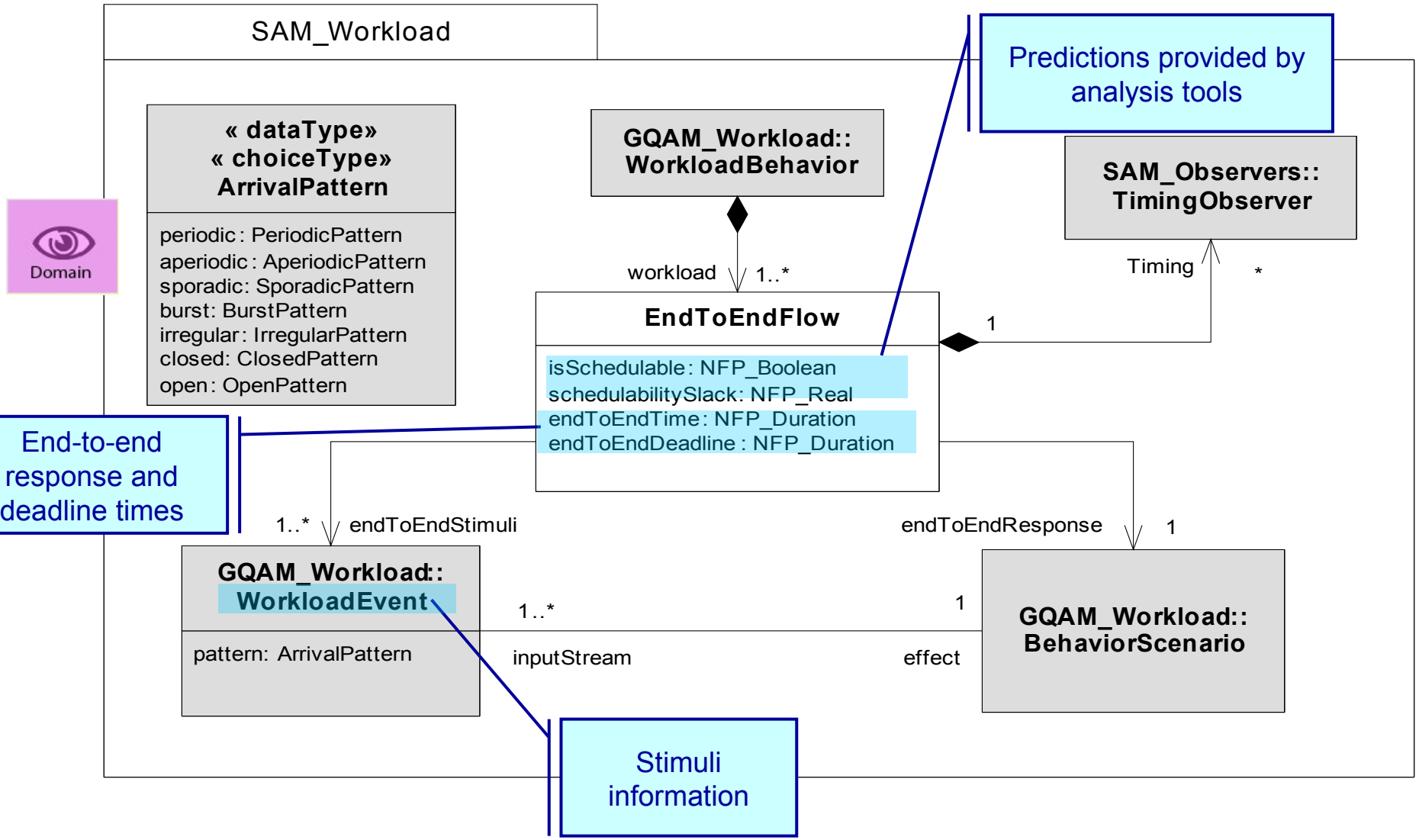
Join:



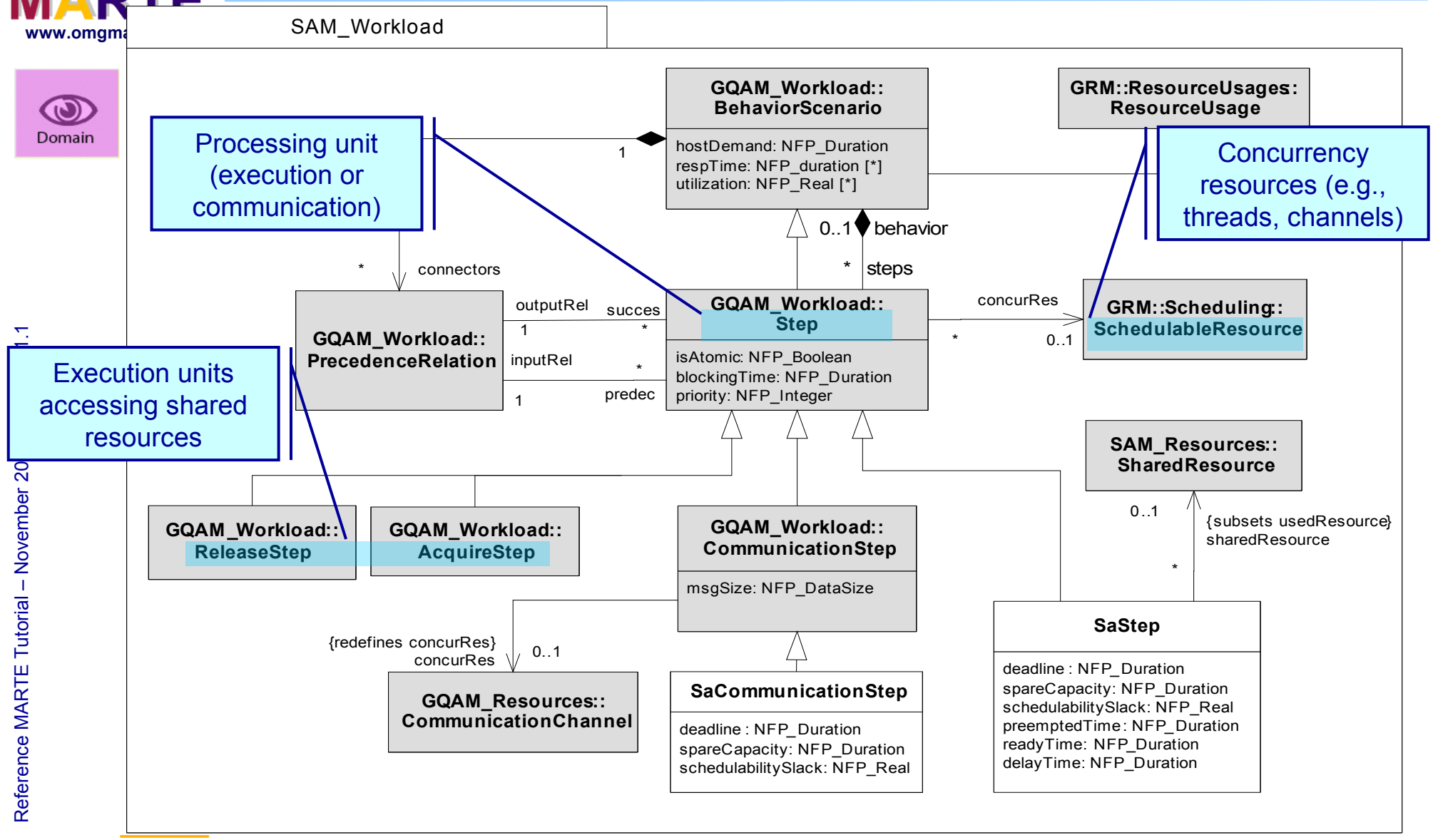
Fork:



SAM: Workload Domain Metamodel (end-to-end)



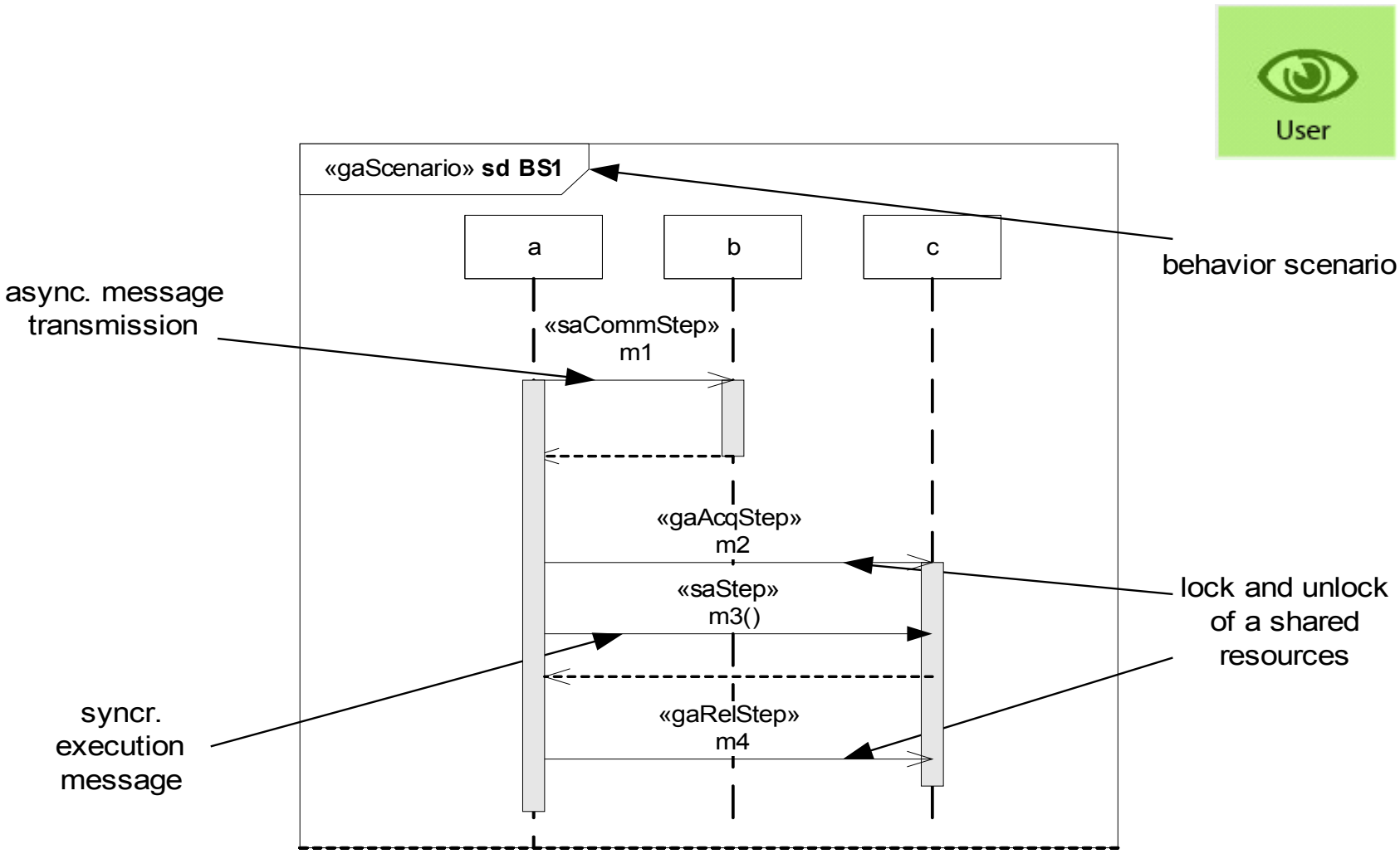
SAM: Workload Domain Metamodel (detailed behav.)



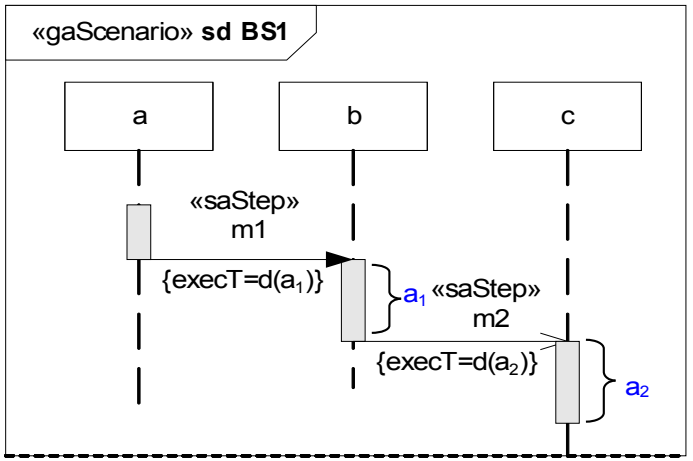
SAM: Example of Stereotype Extensions Usage

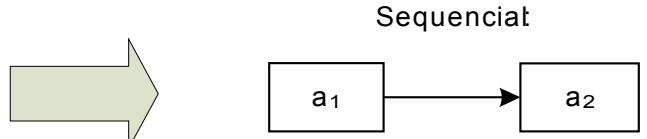
SAM Domain Model	SAM Stereotype	UML Metaclasses	Context
WorkloadBehavior	GaWorkloadBehavior	UML::Interactions::Fragments:: **CombinedFragments**	Modeled in a high-level interaction
EndToEndFlow	SaEnd2EndFlow	UML::Interactions::Fragments:: **InteractionOperand**	Modeled in a high-level interaction
WorkloadEvent	GaWorkloadEvent	UML::Interactions::BasicInteractions:: **Message**	Modeled in a high-level interaction
BehaviorScenario	GaScenario	UML::Interactions::BasicInteractions:: **Interaction**	Modeled as a low-level interaction nested within a higher-level interaction
Step **CommunicationStep** **ReleaseStep** **AcquireStep**	SaStep SaCommStep GaRelStep GaAcqStep	UML::Interactions::BasicInteractions:: **Message**	Messages in low-level interactions

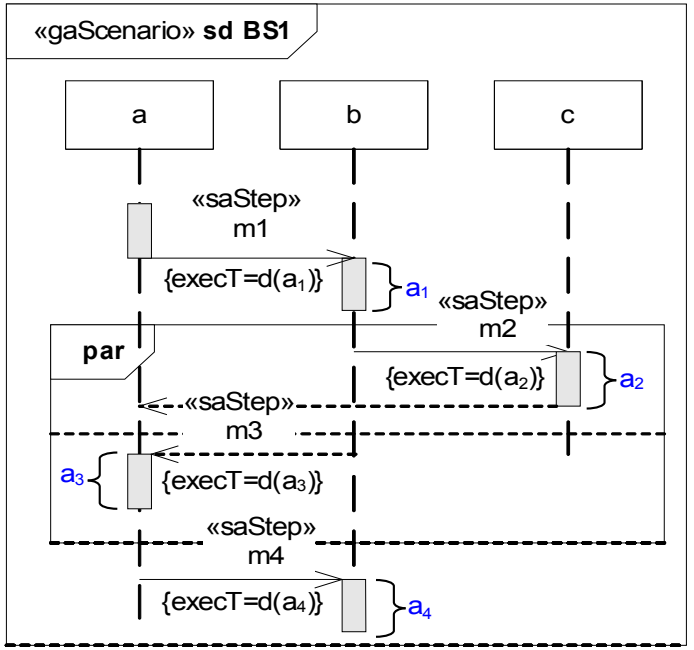
SAM: Examples of Behavior Annotations

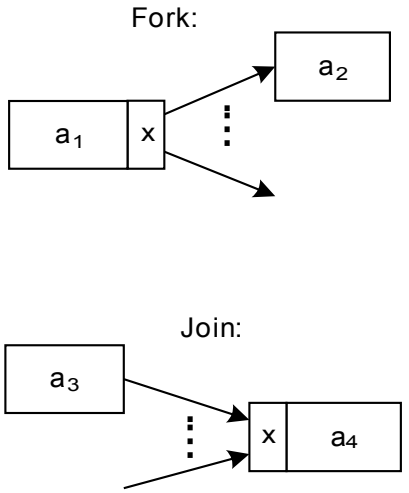


SAM: Example of Precedence Relations Annotation

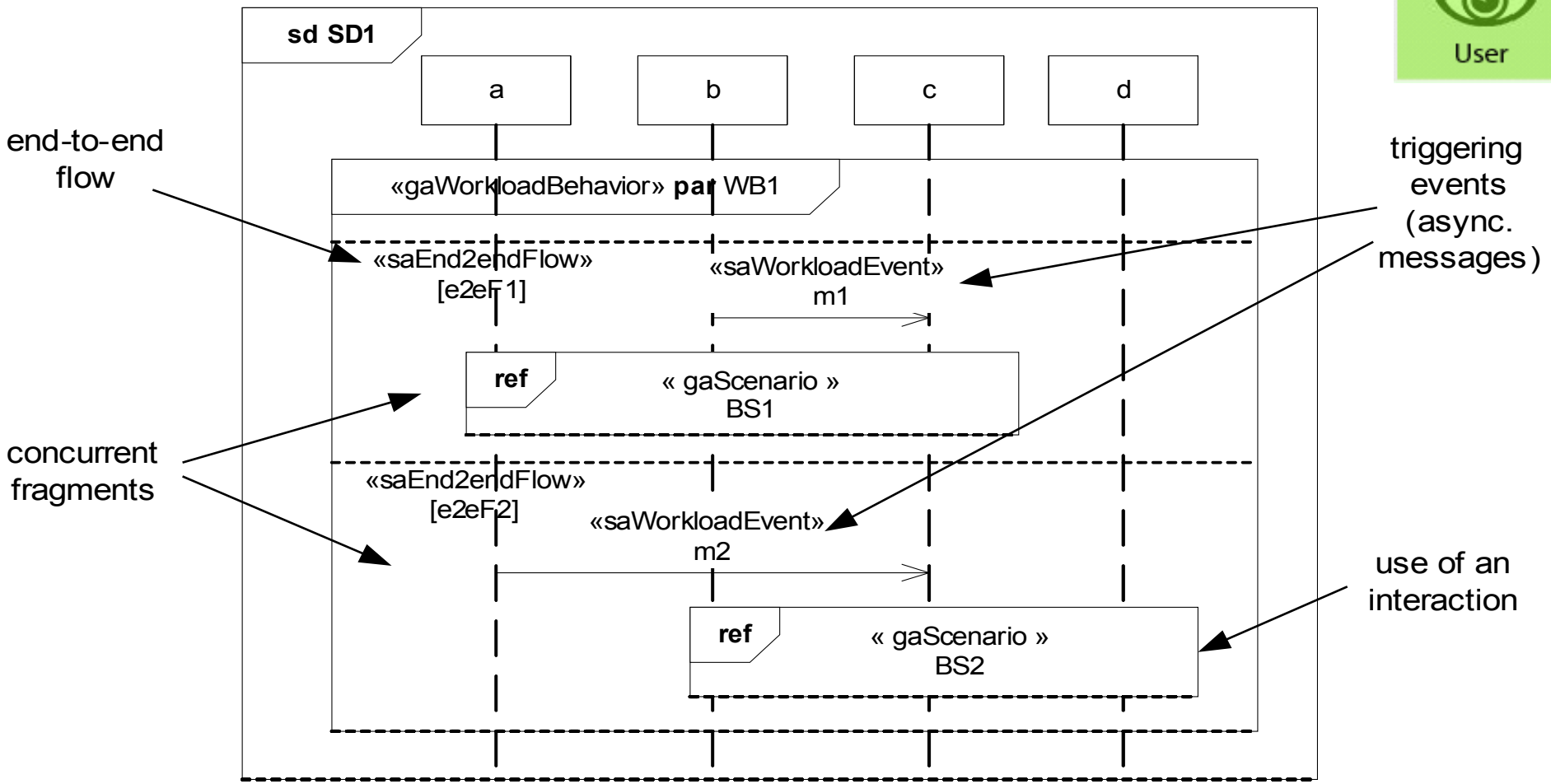






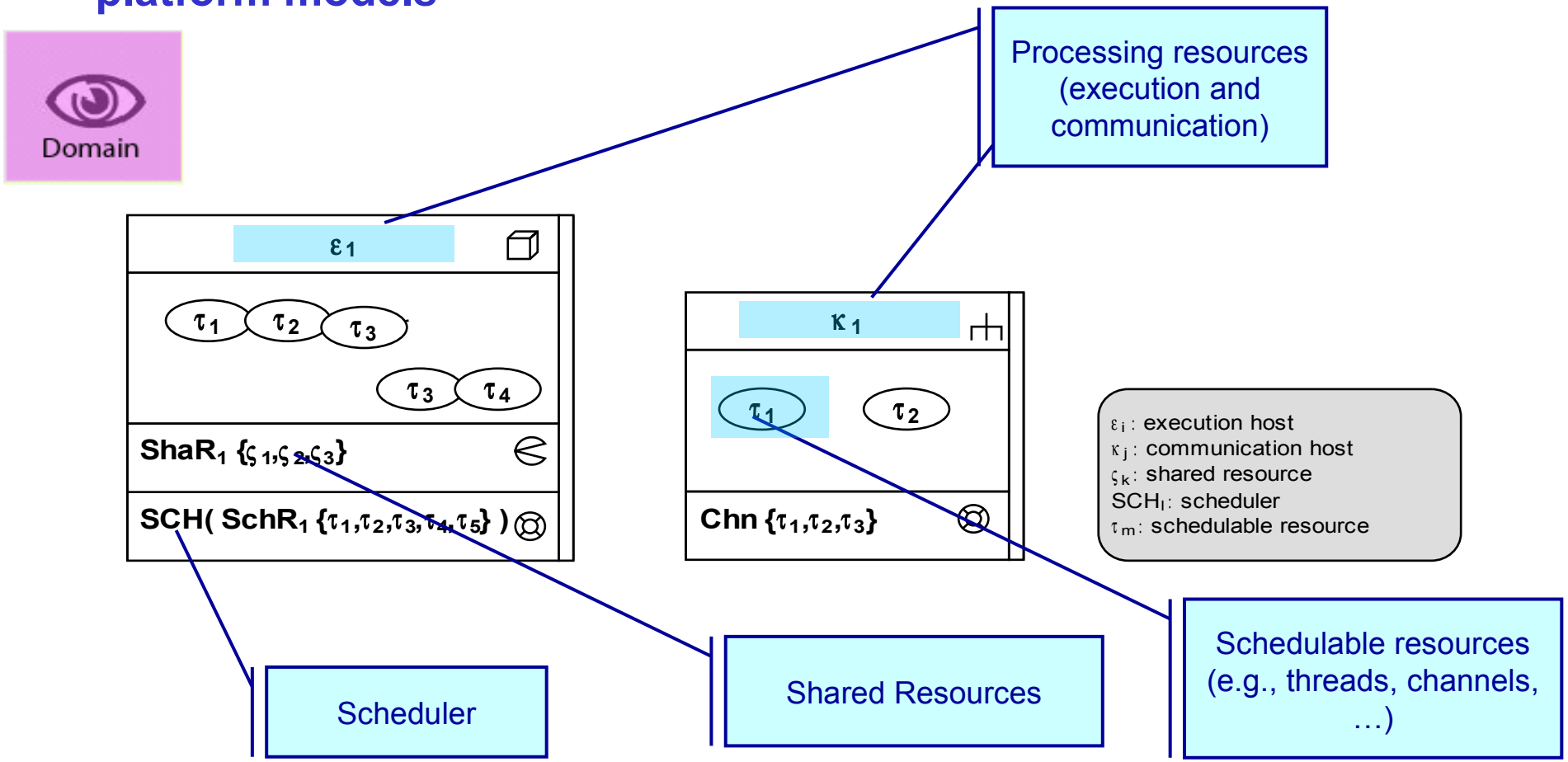


SAM: Example of Workload Annotations

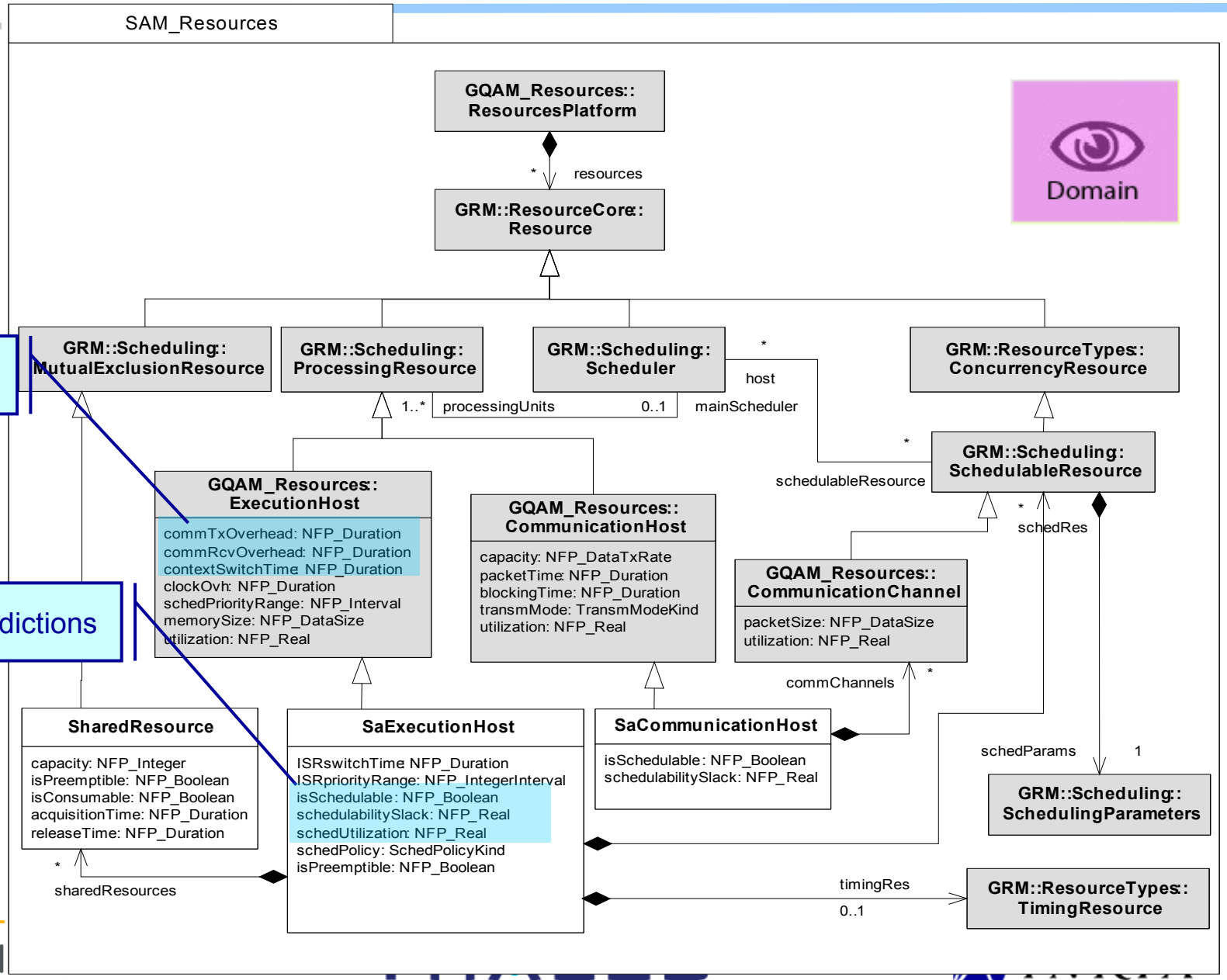


SAM: Resources Concepts

- **Provide additional (analysis-specific) annotations to annotate resources platform models**



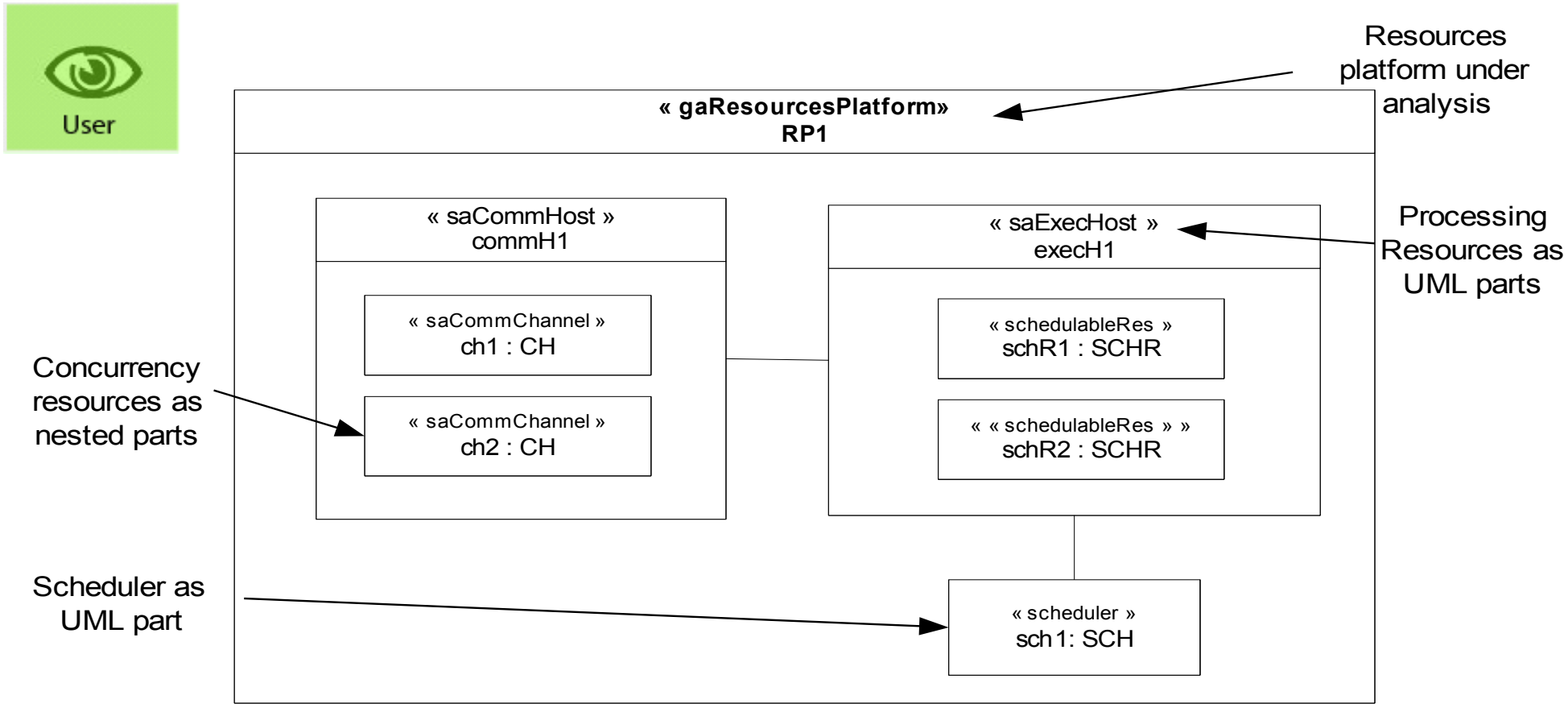
SAM: Resources Domain Metamodel



SAM: Examples of the Stereotypes Usage

SAM Domain Model	SAM Stereotype	UML Metaclasses	Context
ResourcesPlatform	GaResourcesPlatform	UML::StructuredClasses:: **SctructuredClass**	Main container of resources
SaExecutionHost **SaCommunicationHost** **GRM::Scheduler**	SaExecHost SaCommHost Scheduler	UML:: StructuredClasses:: **Property**	Parts of the resources platform
GRM::SchedulableResource **SaCommChannel**	SchedulableRes SaCommChannel	UML:: StructuredClasses:: **Property**	Parts of processing resources

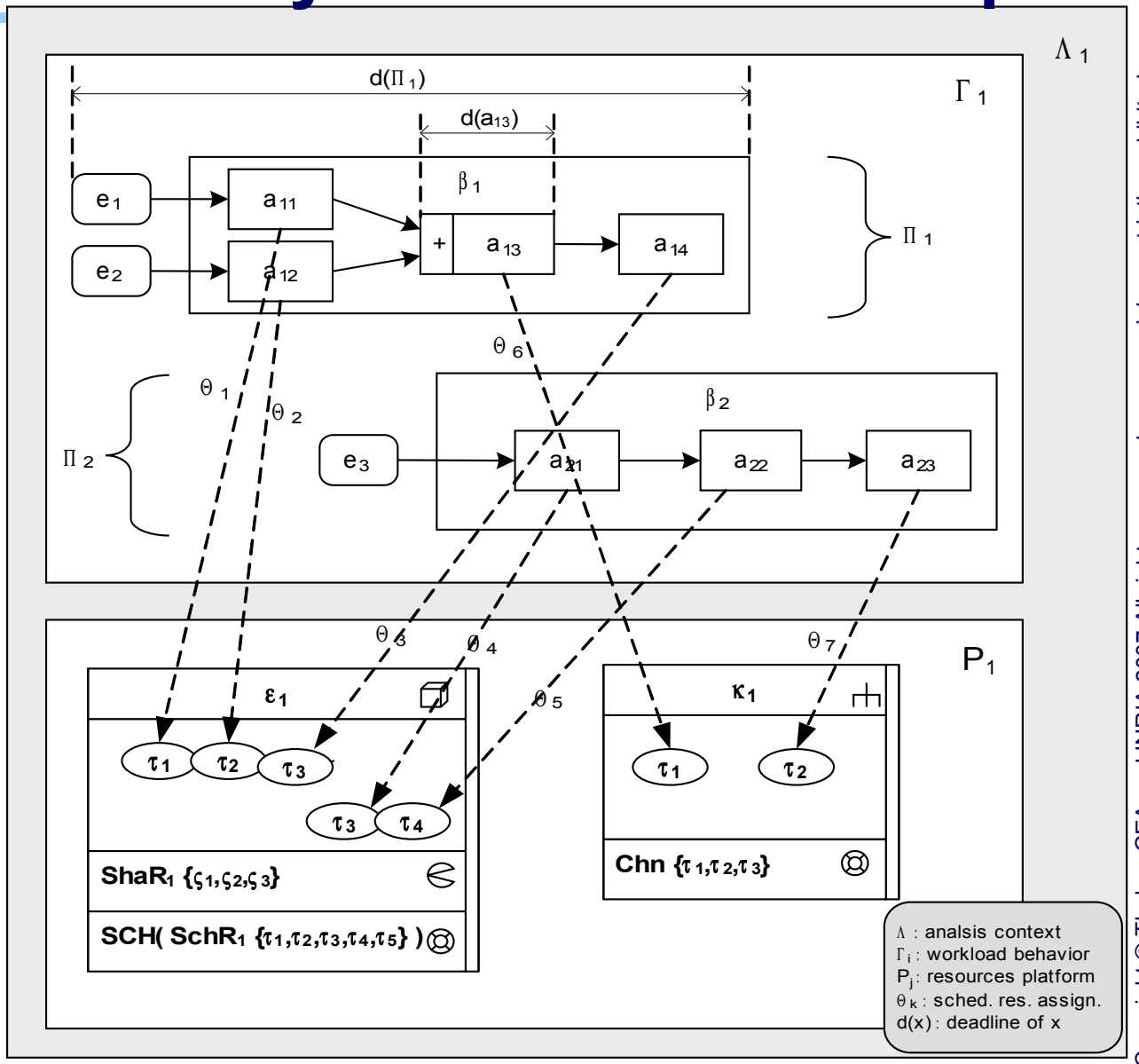
SAM: Example of Resources Stereotype Usage



SAM: Analysis Context concepts

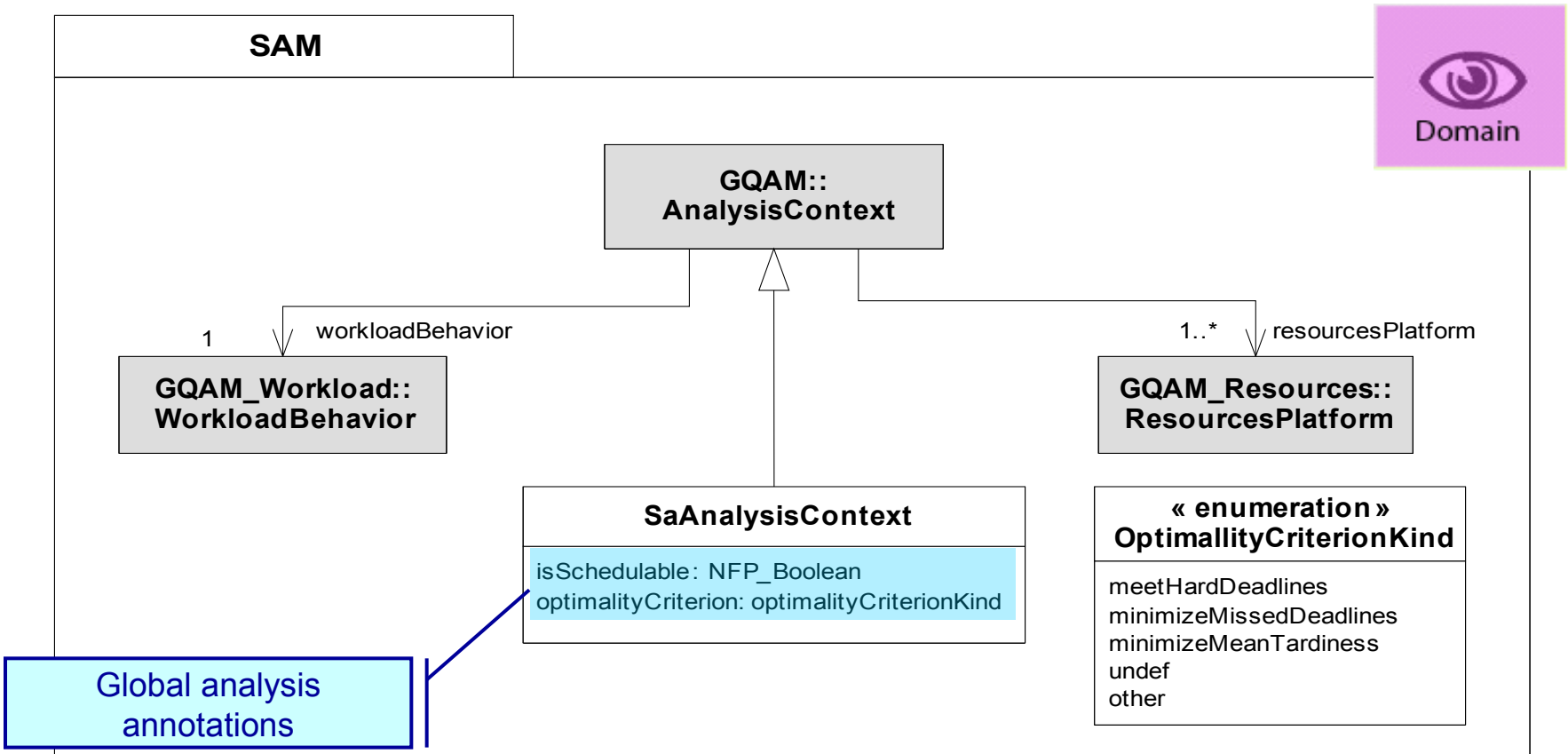
- An analysis context is the root concept used to collect relevant quantitative information for performing a specific analysis scenario.

- An analysis context integrates workload behavior models and resources platform models.

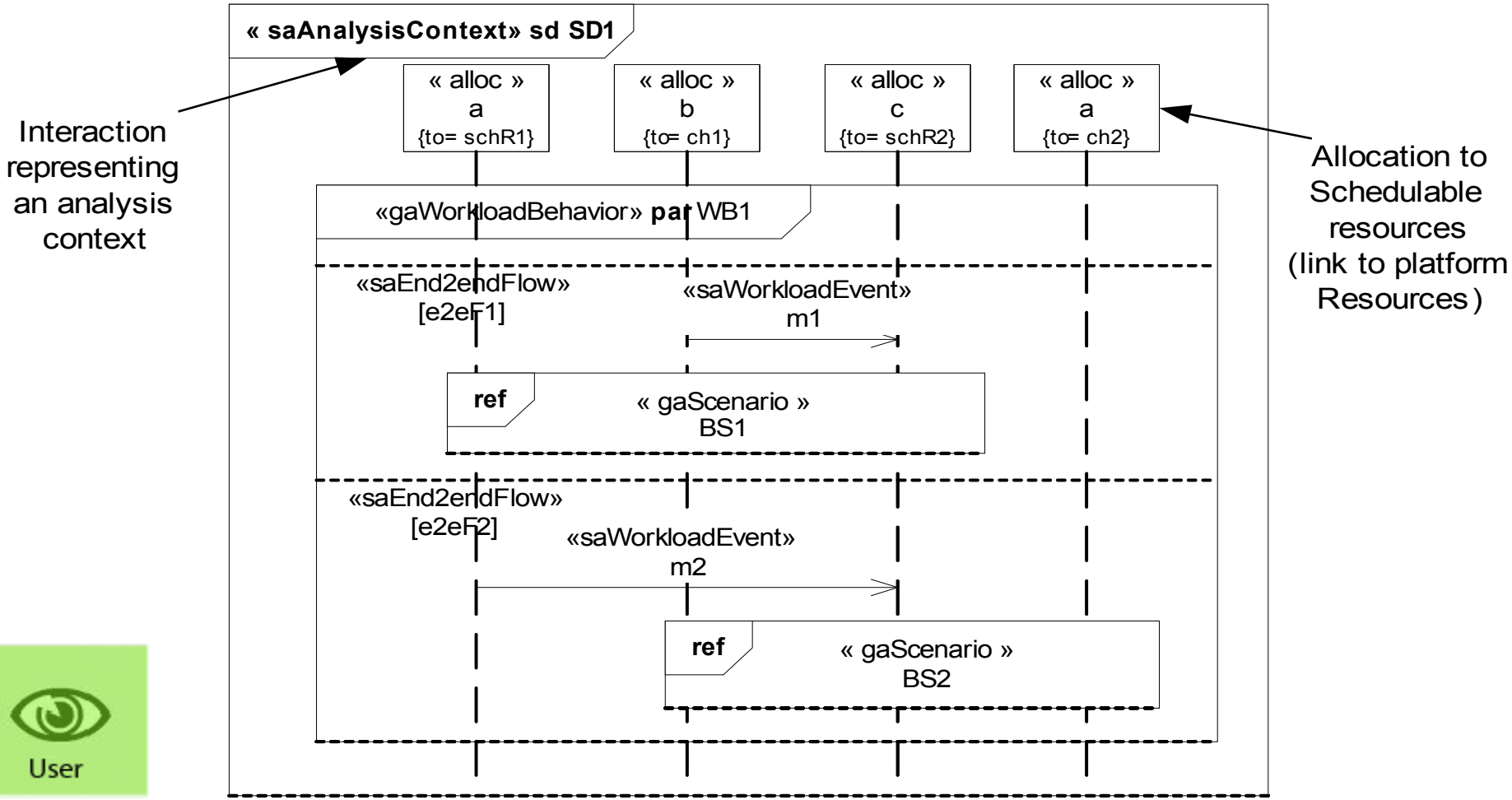


Domain

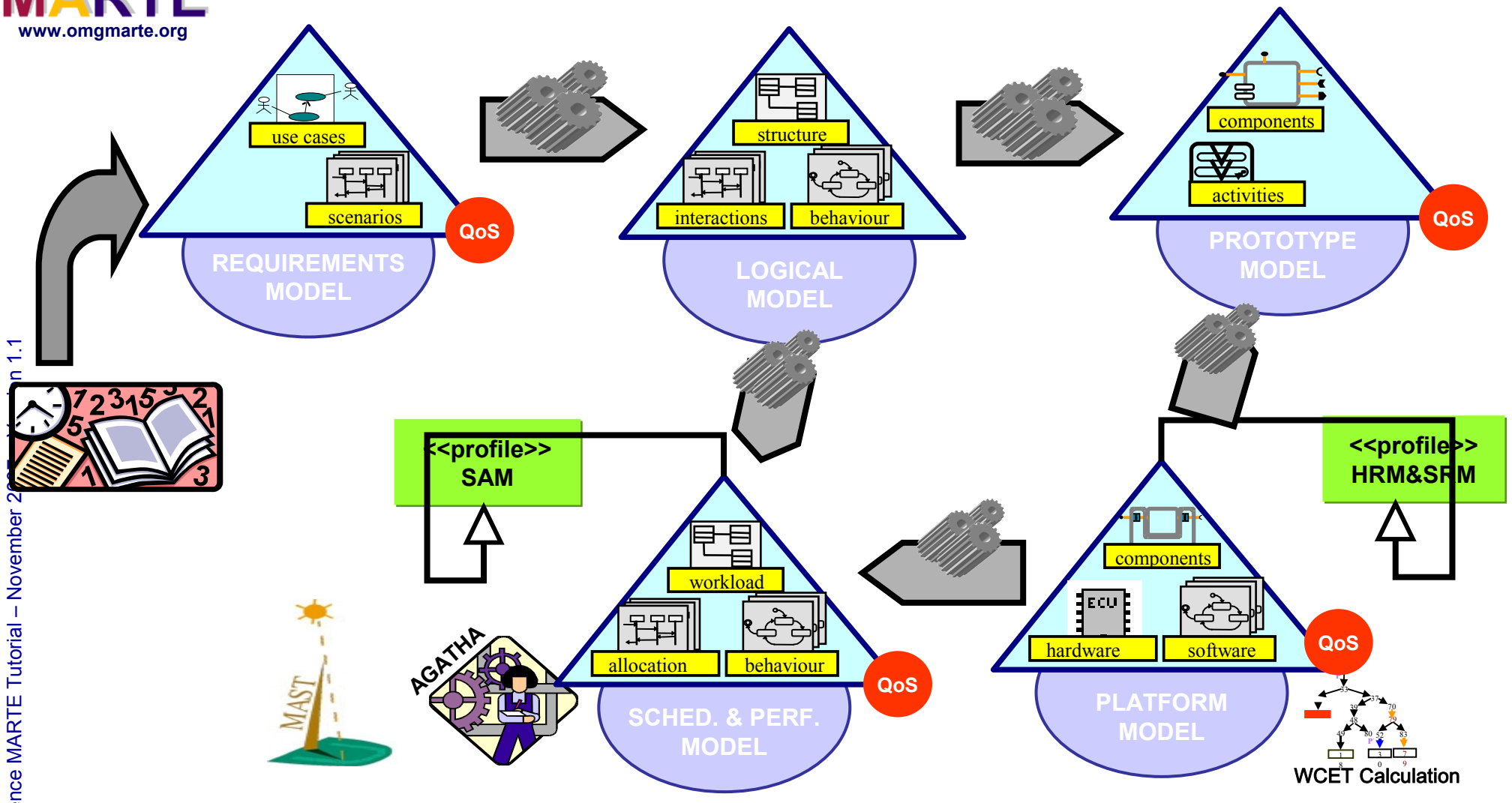
SAM: Analysis Context Domain Metamodel



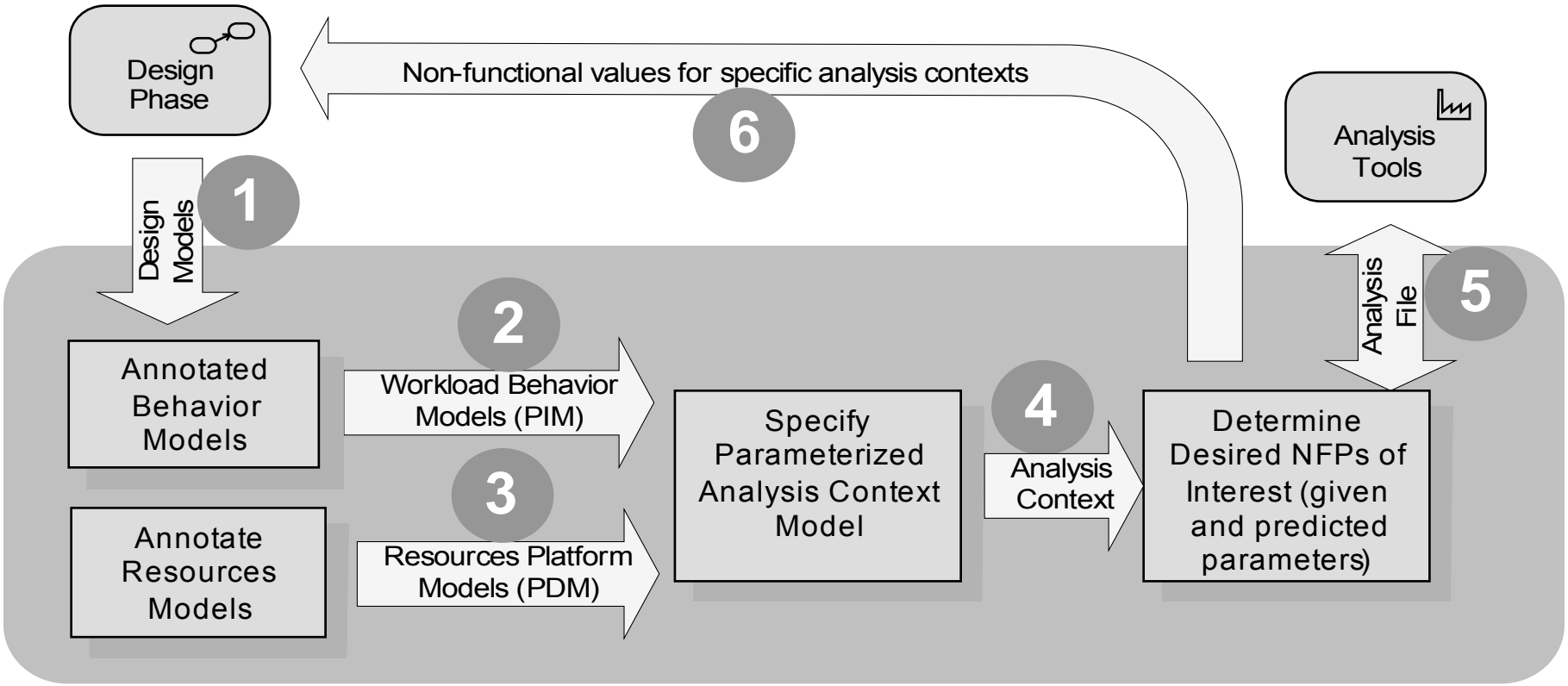
SAM: Example of Analysis Context Stereotype Applic.



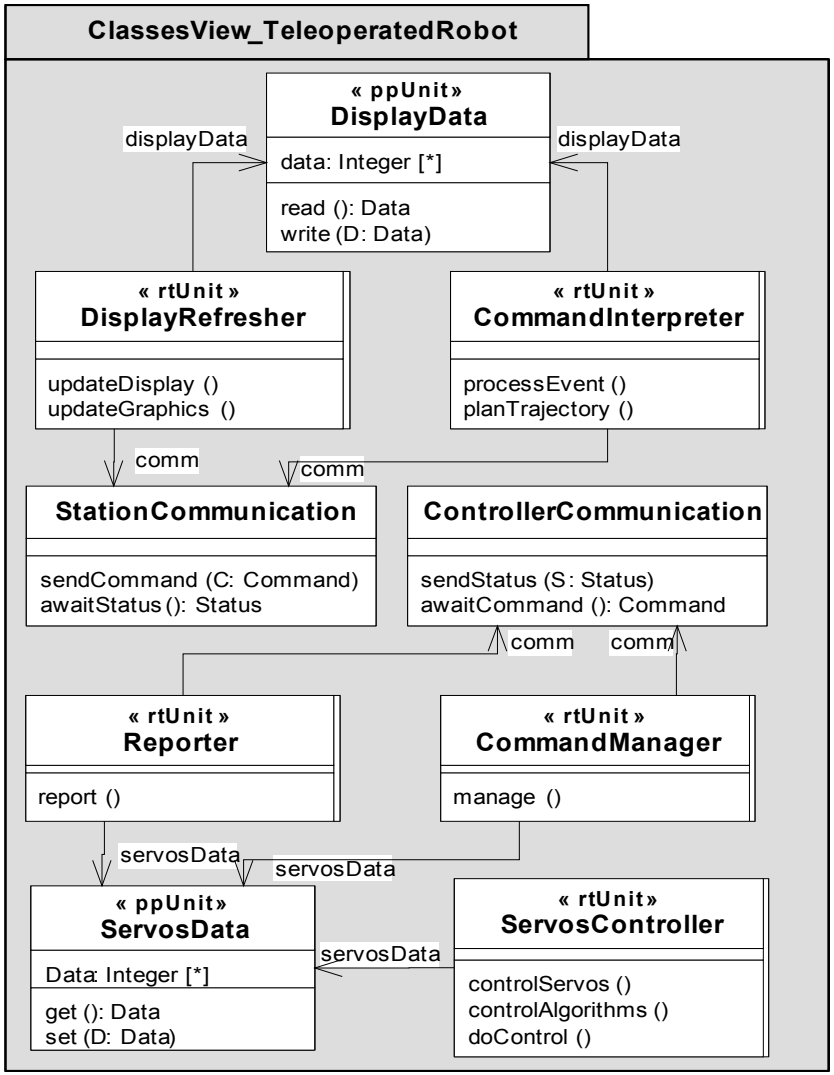
Example of Global Development Process

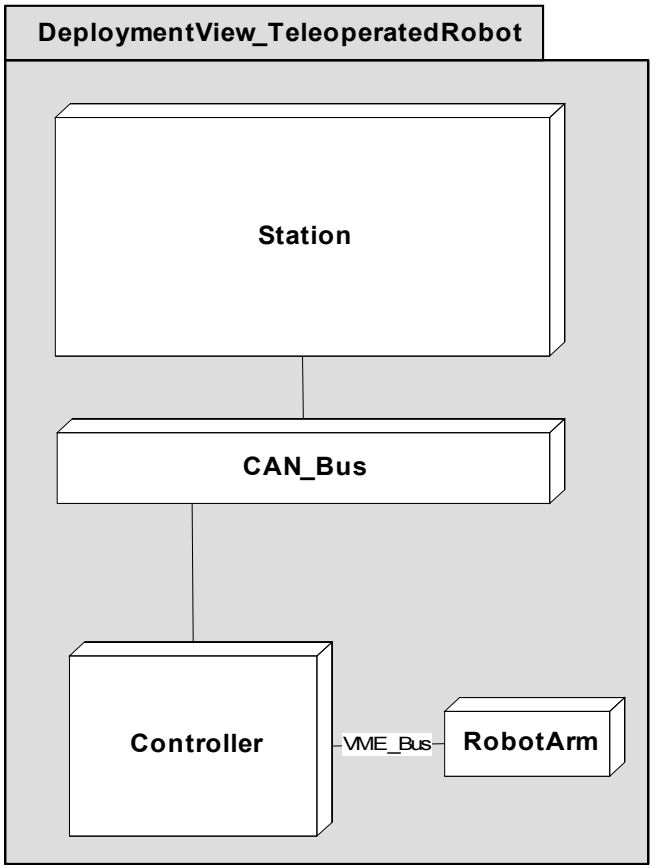


General Procedure to Use the SAM Profile

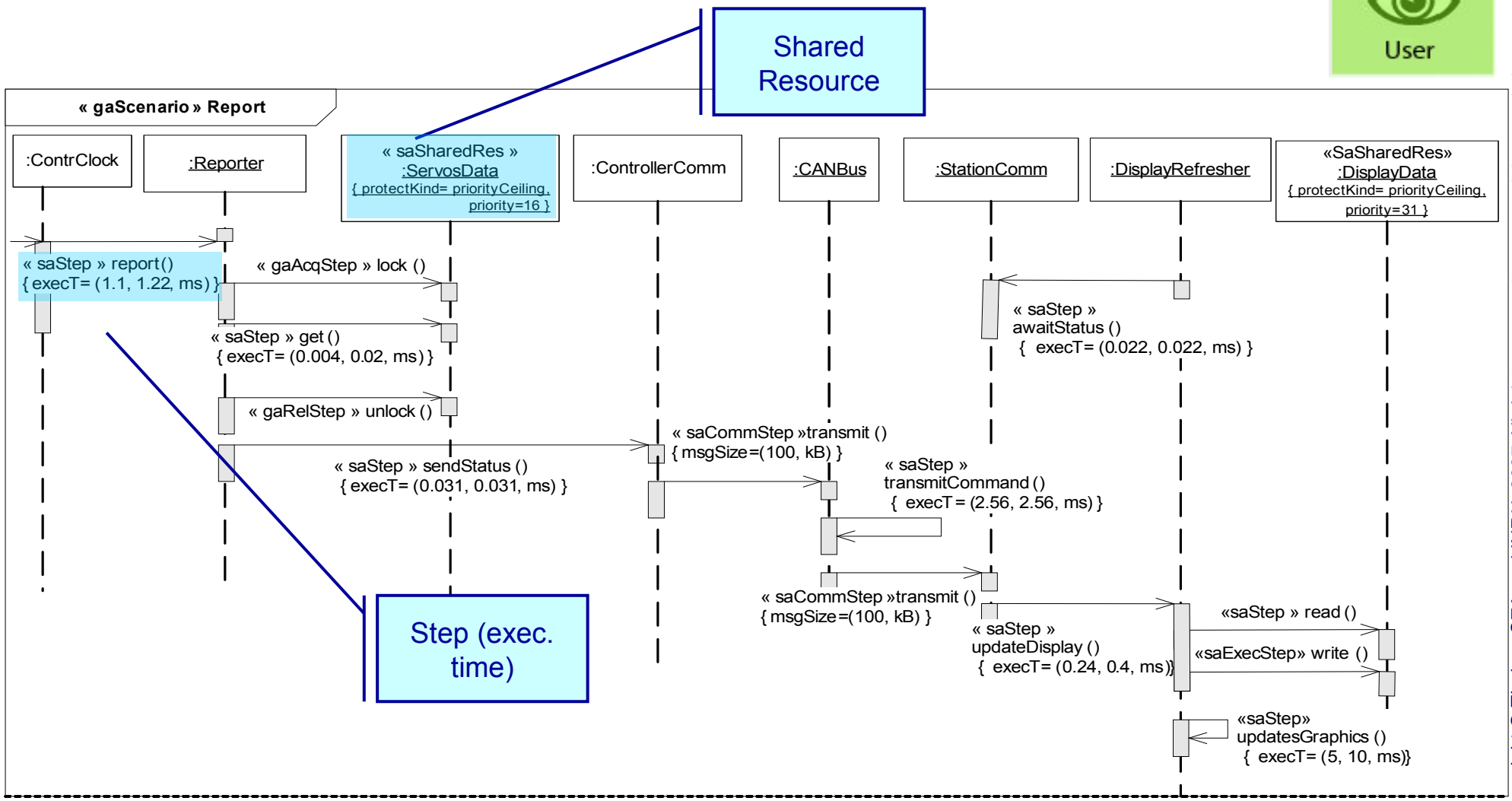


Example: A Teleoperated Robot

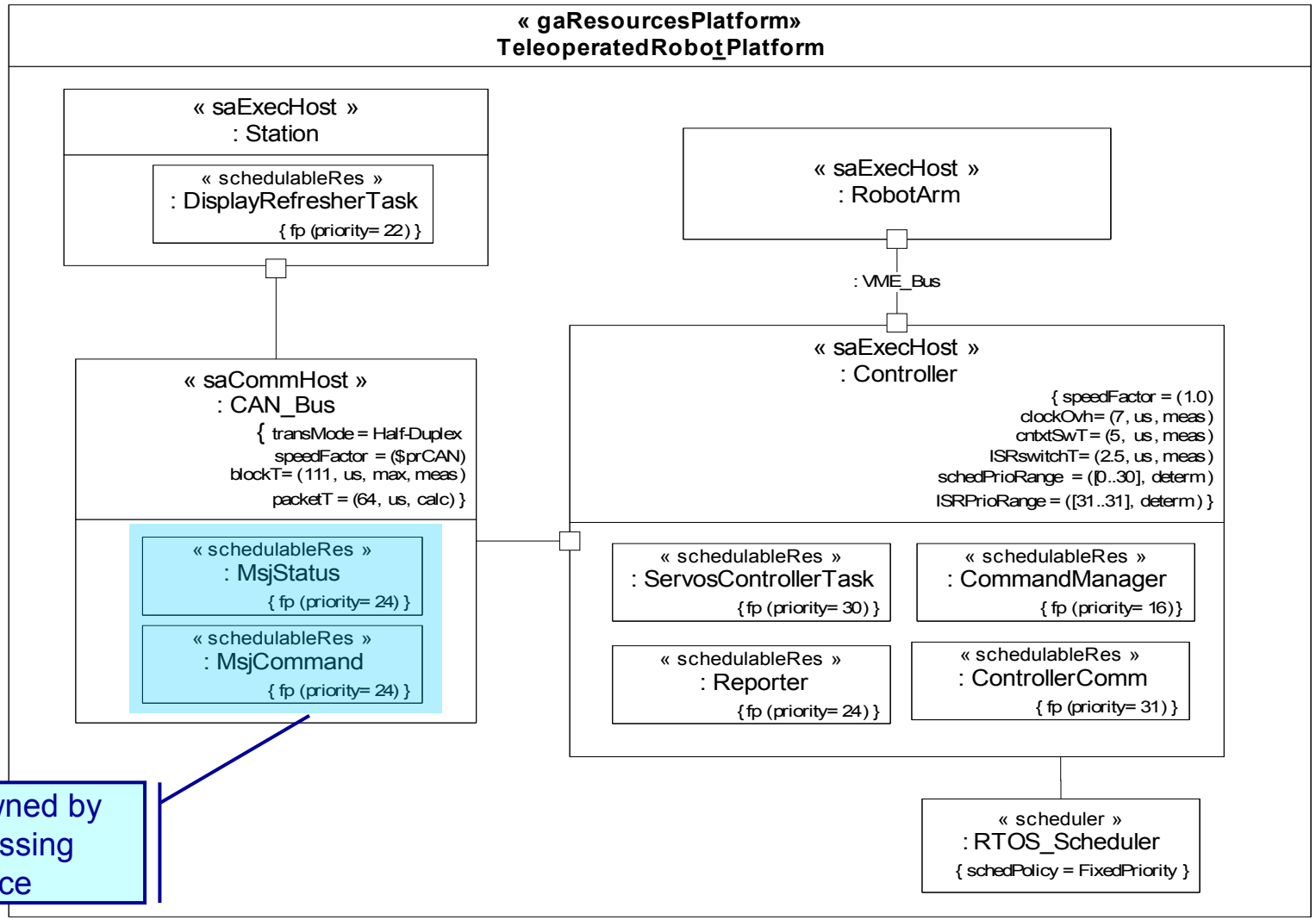




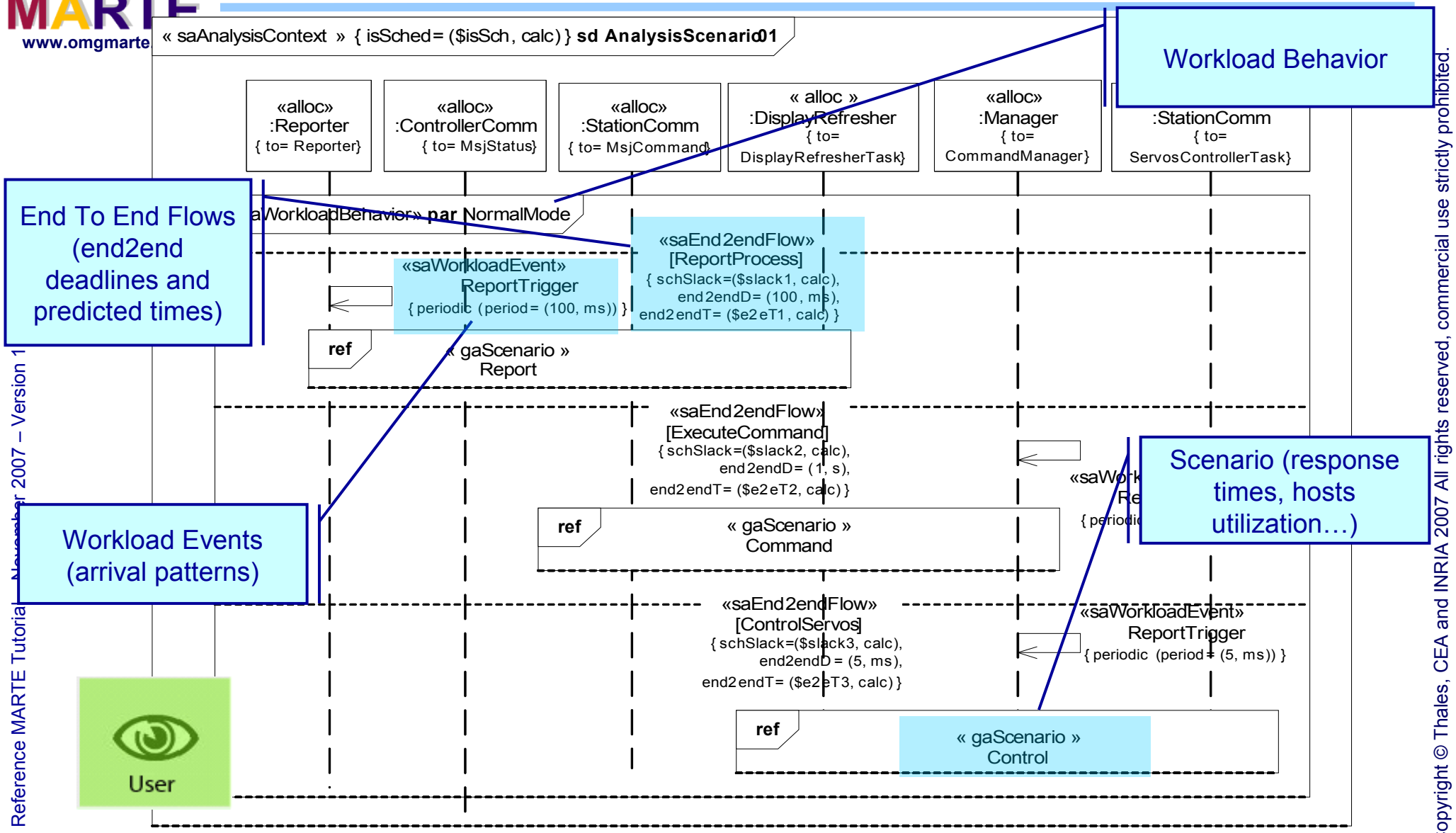
Example of Annotated Scenario with SAM



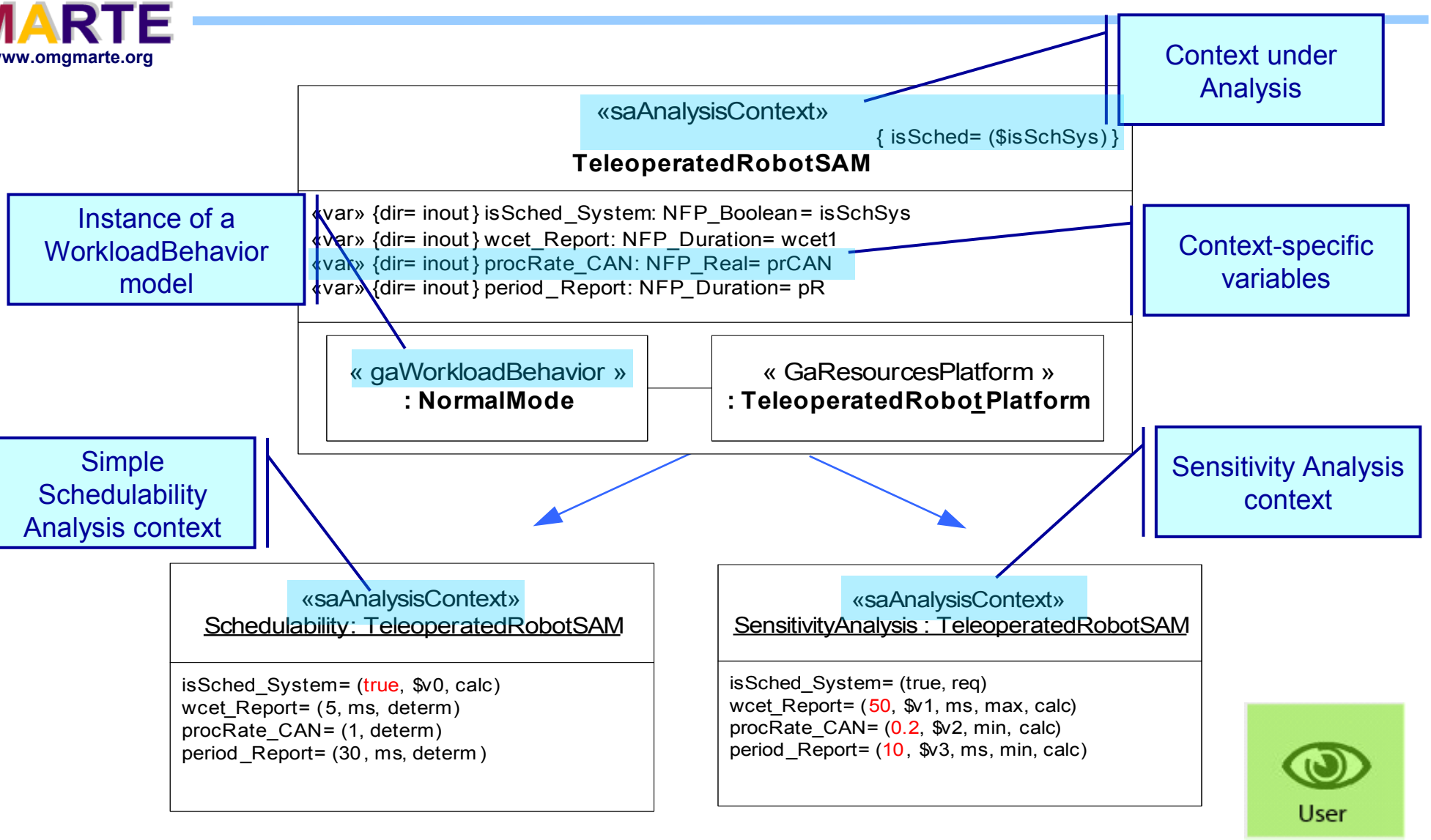
Example of Annotated Resources Model with SAM



Example of Analysis Context Model



Example of Parametric Analysis Context



MARTE Tooling

- **Current Implementations supporting MARTE**
  - Full MARTE Profile & Libraries for Eclipse UML2
  - VSL edition assistant and type checker as a Eclipse plug-in for the UML Papyrus tool and RSA 7.0

- **On-going work:**
  - Eclipse plug-ins to transform UML models annotated with the SAM profile to input files of MAST, SymTA/S, Cheddar and RapidRMA tools

MARTE Open Source Implementation in

UML Papyrus: **www.papyrusuml.org**
IBM RSA: **www.omgmarte.org**

Conclusions on MARTE's Analysis

- **Industrial Use of V&V can benefits from MDE**
  - Analysis task must be cohesively integrated with Design tasks
  - Application of individual analysis techniques should be regarded as an essential part of an integrated V&V methodology

- **Methodological support is still under way:**
  - Complex analysis scenarios for Interface-Based Design, Multiobjective Design Space Exploration…
  - Means to manage NFP measurement models
  - Methods to map/transform MoCCs into analysis models