SIMULATION MODELS
FORMALIZATION

Pau Fonseca i Casas; pau@fib.upc.edu
The need of a conceptual model
Hypotheses

- What is inside the model?
- Hypotheses
  - Systemic
  - Structural
Advantages of use a conceptual model

- Textual specification is less precise.
- Conceptual model have in a detailed manner, the dynamical relations between the different elements of the interest process.
  - Constitutes an specification by itself.
- Simplifies the dialog between the different parts that are involved in the project.
- Constitutes a representation of the simulation model independent of the selected tool used to build the model.
Conceptual model formalization

- Formalism must be independent from the simulation tools.
- The formalized model must allow some analysis.
  - To determine relations between components.
Conceptual model formalization

- Formalism must allow an easy transformation to the representations supported by the existing simulation frameworks.
  - Simplify the implementation process.
  - To evaluate alternatives.
Some aspects of the model can be no specified, without causing problems in the transformation to other representations. MODULARITY

The model must be defined in terms that no constrain its codification in a particular mechanism of simulation clock update.
Modularity

- The capacity to describe the behavior of each subsystem, independent from the other subsystems that compose the model.
  - Incremental design of the model.
  - Simplifies the verification and the validation of the model.
  - Each different stage ➔ implementation stage.
Assure the Modularity

1. A module cannot access directly to the state of other modules or components.

2. A module must own a set of ports (input/output) to allow the interaction with the other parts of the model.
Conceptual models

- Flow models.
- Queue networks.
- Petri nets
- Colored Petri nets.
- SDL language
- DEVS
- Causal and Forrester diagrams.
Working with different formal languages

- Three of the main mechanisms for doing this:
  - Meta-formalism.
  - Common formalism.
  - Co-simulation.

Meta-formalism

- A formalism that incorporates the different formalisms of the various sub models that makes up the system.
Common formalism

- A mechanism that converts all formalisms to a common formalism.
- Transforming algorithms from:
  - SDL $\rightarrow$ DEVS $\rightarrow$ Petri Nets...
Co-simulation

- Independent simulators that work together
- HLA: The **High Level Architecture** (HLA) is a general purpose architecture for distributed computer simulation systems. Using HLA, computer simulations can interact (that is, to communicate data, and to synchronize actions) to other computer simulations regardless of the computing platforms. The interaction between simulations is managed by a Run-Time Infrastructure (RTI).
Co-simulation with SDL

- We use SDLPS (on the practical sessions)
Some data

How far is modeling used in your activity?

- We do not have any formal models
- We use models for documentation
- We do simulation of our models
- We use model checking technologies
- We generate only code skeletons out of our models
- We generate all the code out of our models
- We generate test cases out of our models
In the coming year do you plan to improve one of the following aspect?

- Documentation
- Traceability
- Software modeling
- Model simulation
- Model checking
- Testing
- Code coverage
- Quality
- CMMI level

2011 Survey
Flow models

Simulation models formalization
Flow models (data)

- Magnetic disc
- Document
- Multiple document
Flows models (Processes)

- State
- Process
- Decision point
Pediatrics example

- Models a pediatrics example.
- If a new emergency arrives a special process takes cure of it.
- If X ray is needed, or blood analysis, is done in a second visit
- Finally the patient release the system.
Flows models
Flows models (best)

- Simple
- Allows to describe the system faster.
Flows models (worse)

- No description about the implementation.
- No description about the events.
- Is not calculable.
- Not structured methodology, not specific of the OR.
Queue networks

Simulation models formalization
Queue networks

split
fork
Join
alocatar
Crear
Destruir
Liberar
Queue networks \((M|M|S)\)
Queue networks (best)

- Simple
- Allows to understand the system faster.
- Specific to describe queue models.
Queue networks (worse)

- No description about the implementation.
- Do not describe too much about the events management.
- Is not always calculable.
  - Some models can be calculated following the queue theory.