Simulation Multi-Niveau : Propositions Méthodologiques

Jérôme Plouchart

E-mail: jploucha@sophia.inria.fr

31 mai 2000

Introduction

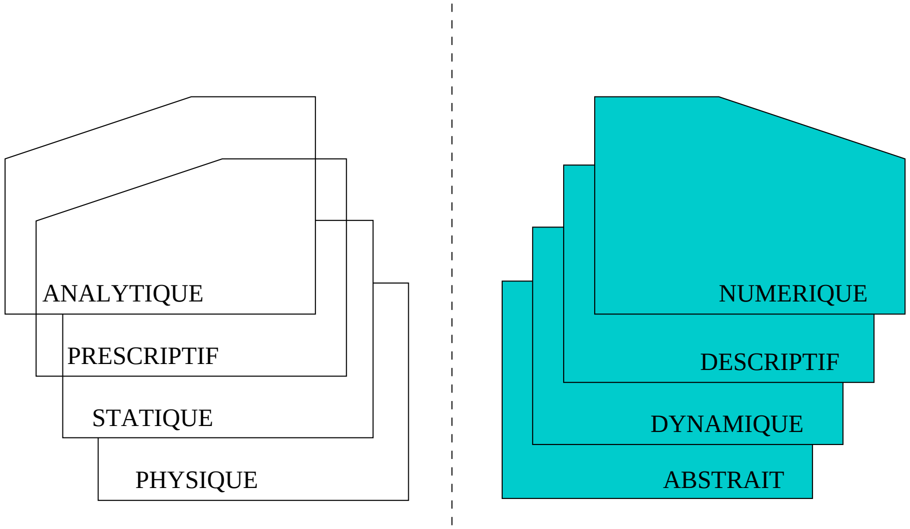
Le contexte actuel :

- Croissance de la complexité des systèmes à simuler.
- Lourdeur des simulateurs en temps de calcul.
- Rigidité des modèles simulés.

Les besoins et thèmes de réflexion

- Niveau de granularité variable : représentation plus ou moins détaillée des différents composants du système.
- Possibilité d'interchanger des “blocs” de modèle.
- Philosophie de la simulation, framework.

Schéma de classification de la simulation



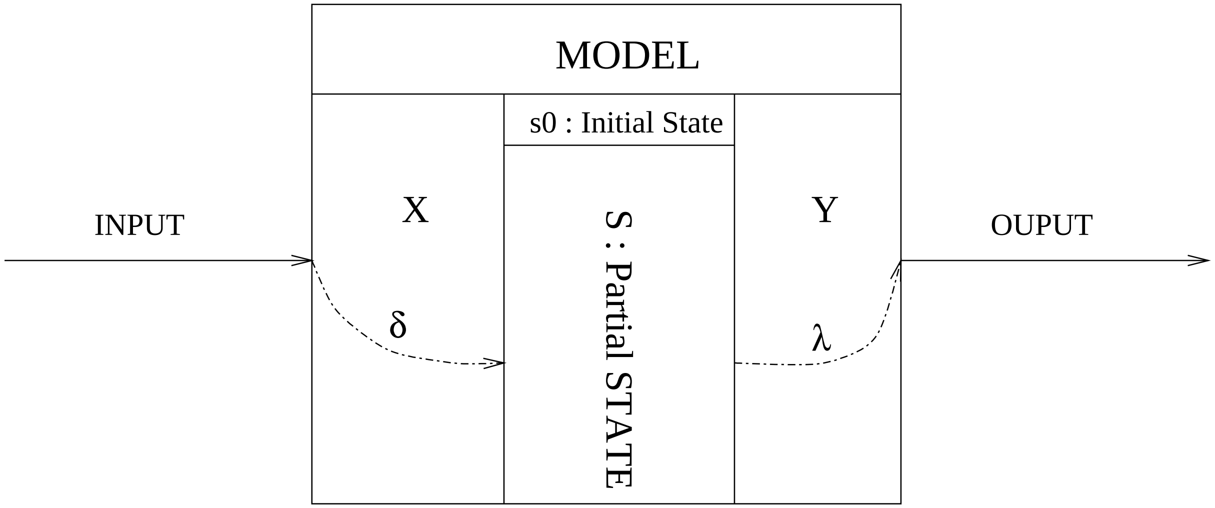
Le framework

- Outil de filtrage monde réel/esprit du concepteur ⇒ modèle
- squelette amenant la dynamicité et les restrictions.
- Exemple :
  – GPSS en termes de transactions, stores, facilities, queues.
  – SIMSCRIPT en termes de entities, attributes, sets, events.

Un exemple de formalisme : DEVS

- Modèle de base de DEVS[1] : $M = (X, s_0, S, Y, \delta, \lambda, \tau)$
  - $Q = (s, e)$: l'ensemble des états avec “e” le temps écoulé depuis la dernière transition
    $q_0 = (s_0, 0)$ état initial
  - $X^{\phi} = \{X \cup \phi\}$: soit une entrée arrive, soit rien ne se passe $(= \phi)$
  - $\delta : Q * X^{\phi} - > S$: fonction de transition
  - $\tau(s)$: fonction Time advance.

[1] Discret Event System Specification


MODEL
s0 : Initial State
X
S : Partial STATE
Y
INPUT
OUPUT
δ
λ

- Formalisme SDE[2] : $\mathfrak{S} = (X, s_0, S, Y, k, I, \{\delta_i | i \in I\}, \{\lambda_i | i \in I\}, \{\tau_i | i \in I\})$
  SDE $\Leftrightarrow$ DEVS & k : $S \rightarrow I$, fonction d'indexation.
- Formalisme DSDE[3] : $DSDEN_N = (X_N, Y_N, \mathcal{X}, M_{\mathcal{X}})$
- Avantage de DEVS : Hiérarchique, Parallèle, Flexible.

[2] Structured Discret Event System Specification
[3] Parallel Dynamic Structure Discret Event System Specification

SSF

f(Ordre de Magnitude) = Comportements différents.

Deux contraintes :

1. La magnitude des ressources requise ⇒ cf loi de Moore & les technologies de simulation parallèle.
2. L'invention de techniques de modélisation pour gérer la complexité (ex : nb d'éléments dans le réseau) ⇒ SSF[4]

La construction d'un framework plus petit et plus générique qui n'impose pas une vue du monde trop restrictive de par les modèles engendrés.

Caractéristiques :

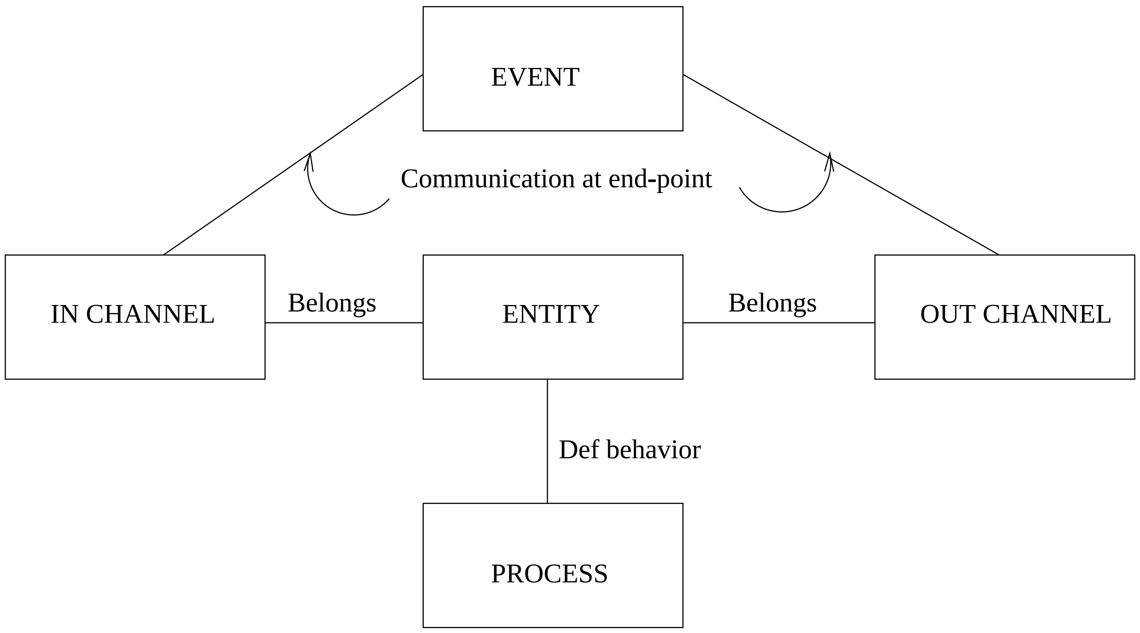
– compact, haut niveau pour l'utilisateur du simulateur.

– Cacher les détails de simulation (threads, processeurs, queues, synchronisation) pour le modélisateur.

[4] Scalable Simulation Framework

Le framework apparaît sous forme de 5 classes :

- ENTITY : conténeur définissant les relations d'alignement parmi les parties du modèle.
- EVENT : quantum d'information échangé.
- IN OUT Channel : points de communication pour les échanges d'événements (EVENT).
  - Multicast IN Channel (Many to one)
  - Multicast OUT Channel (One to Many)
  - Bus Style channel mapping (Many to Many)
- PROCESS : décrit le comportement d'une entité (ENTITY).


EVENT
Communication at end-point
IN CHANNEL
Belongs
ENTITY
Belongs
OUT CHANNEL
Def behavior
PROCESS

ASIMUT

- ASIMUT[5] projet commun RNRT constellation.
- Un des buts : développement de techniques de modélisation permettant d'introduire un fort niveau de généricité dans les modèles.

[5] ASIMUT : Atelier de Simulation de système MUltiédia par satellitTes

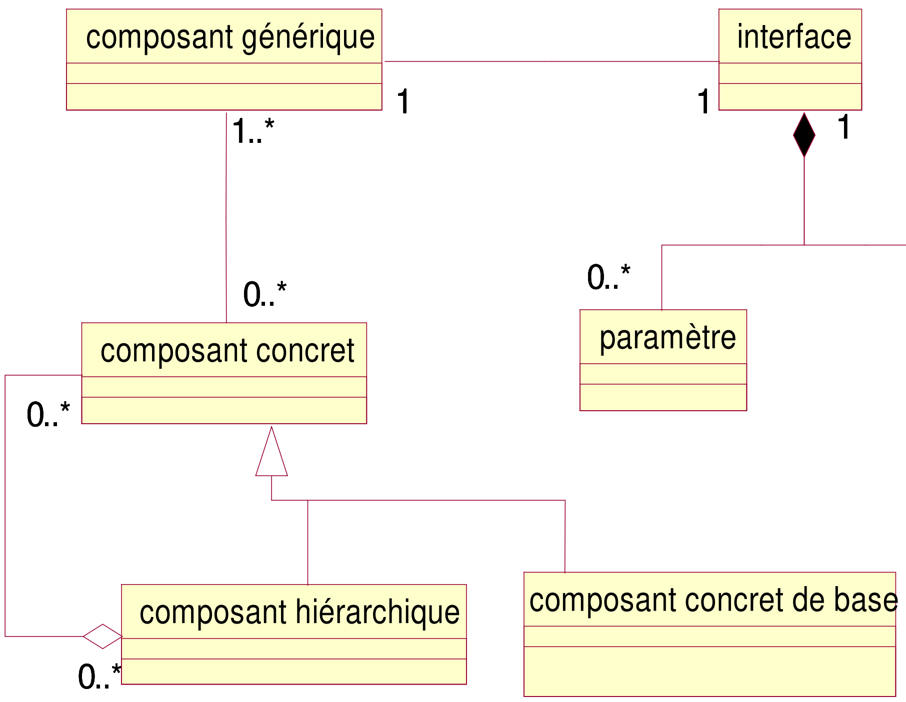
MASCOTTE

Dépendance

- Cyclique : interdépendance .
- Temporelle : t+1 = f(t)
- Spaciale : Pour un même temps donné t (à un δ prés), deux sites localisés dans une même zone vont avoir des influences l'un sur l'autre. (exemple : cas du hand-hoff)

Concept de l'atelier

- Définition des différentes zones d'un composant :
  - Zone descriptive : nom, version, identifiant du rôle générique
  - Zone export : services proposés ou événements émis aux autres objets.
  - Zone Import : services utilisés ou événements traités sur les autres objets.
  - Zone Etat : ensemble des variables caractérisant le composant.
  - Zone Privée : Algo.


composant générique
interface
1
1
1..*
1
0..*
0..*
0..*
composant concret
paramètre
port E/S
0..*
composant hiérarchique
composant concret de base
0..*

Les vues des composants :

– vue verticale : interaction et dépendance fonctionnelle entre composants, organisation hiérarchique du modèle.

– vue horizontale : interaction logique entre composants de même niveau.

HIERARCHIE

- 4 types de hiérarchie dans les modèles :
  - Représentation : dépendance forte du niveau supérieur pour ses niveaux inférieurs.
  - Composition : existence autonome du niveau supérieur qui utilise les niveaux inférieurs dans son comportement.
  - Substitution : plusieurs éléments rassemblés amenant à une simplification du modèle.
  - Spécification : mécanisme d'héritage.

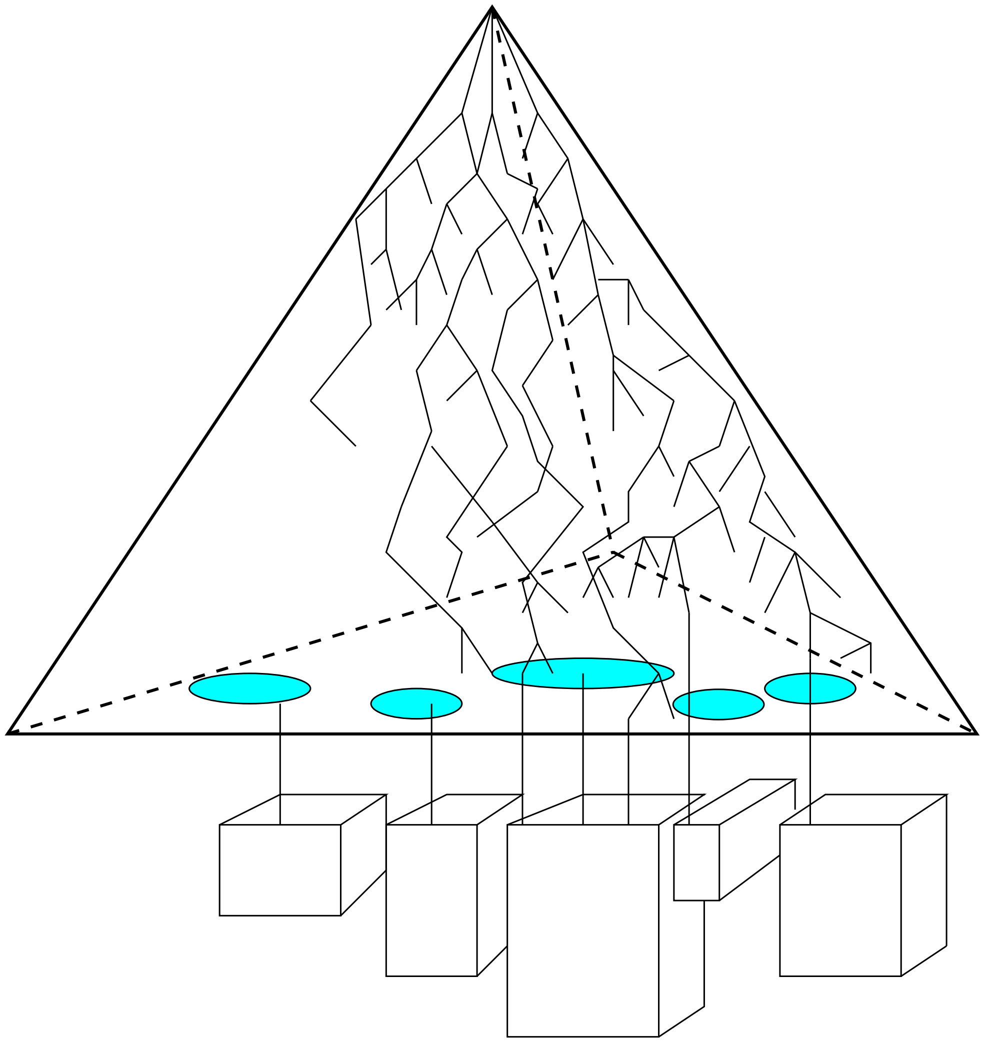
GRANULARITE

- Vue de l'entité :
  - variable
  - gestion indépendante de son comportement interne (cf squelette comportementale).
- Agrégation des paramètres, des fonctionnalités, des entités.
- Hiérarchie des Fonctionnalités, Objets, Indexation des Granularités.
- Base de donnée des espaces d'état d'entrée et de sorties.

GRANULARITE (2)

Précision

- Déterminer le besoin de la précision en fonction du détail du modèle construit.
- Cohérence des échelles de précision.



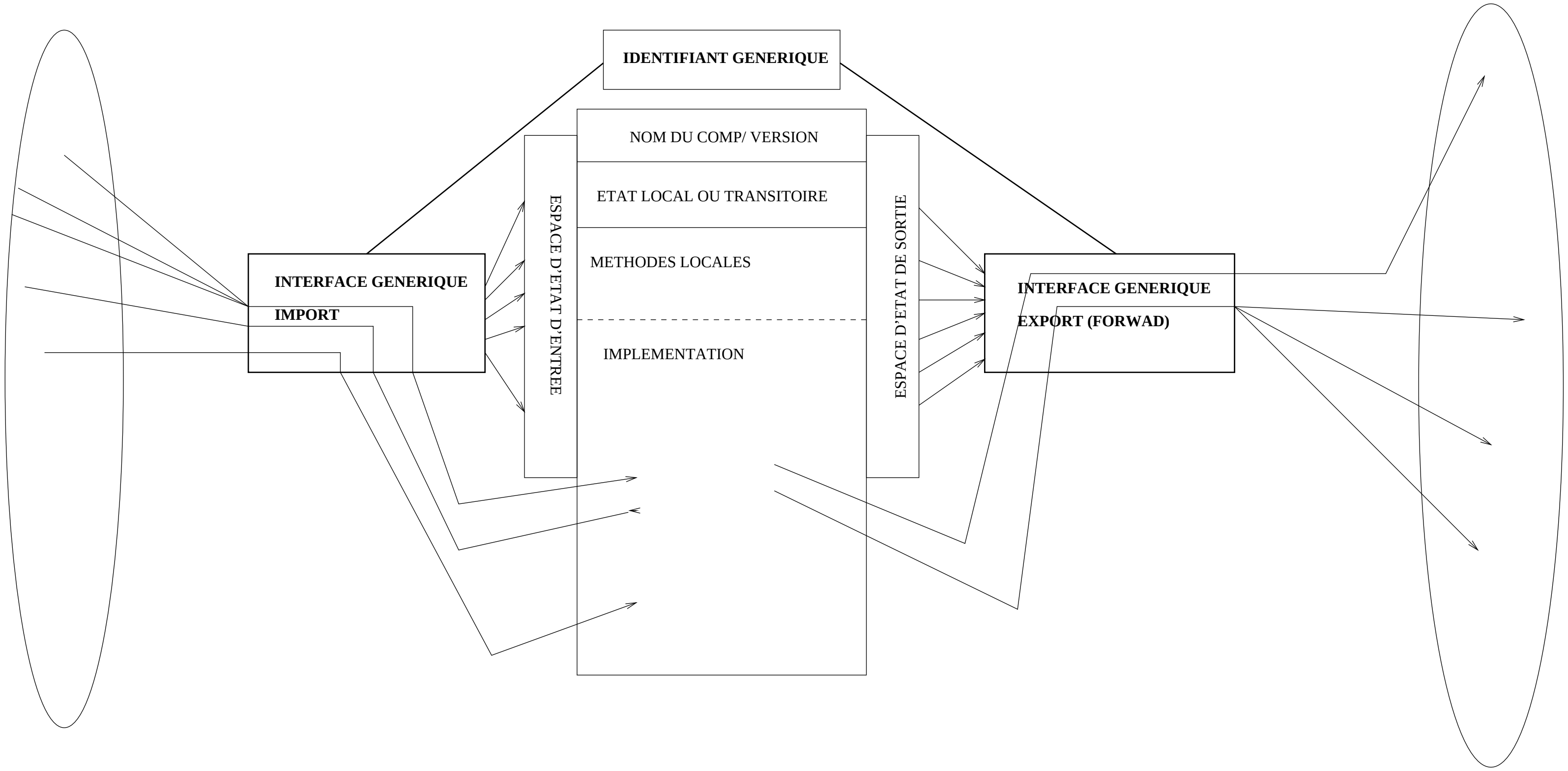
INTERCHANGEMENT

- Problème principal : Dynamicité de l'interaction entre composants.
- Séparer l'implantation des interactions de l'entité avec son environement et celle de son comportement interne $\Rightarrow$ Forwardeur = entités d'interaction.

INTERCHANGEMENT (2)

Contraintes :

- Gestion automatique de certaines contraintes des entités interchangées.
- Equivalence des modèles : équivalence E/S ou équivalence de l'objectif à atteindre $\Rightarrow$ définition d'un tel mécanisme.


IDENTIFIANT GENERIQUE
NOM DU COMP/ VERSION
ETAT LOCAL OU TRANSITOIRE
METHODES LOCALES
IMPLEMENTATION
ESPACE D'ETAT D'ENTREE
ESPACE D'ETAT DE SORTIE
INTERFACE GENERIQUE
IMPORT
INTERFACE GENERIQUE
EXPORT (FORWAD)