CASTOR: un outil d'aide à la conception, l'évaluation et l'audit d'architectures sécurisées

Dominique Chauveau
dominique.chauveau@aql.fr

AQL - Groupe SILICOMP

Journées Systèmes et Logiciels Critiques
Grenoble,14-16 Novembre 2000

Contexte

➨P.E.A. (Programme d'Etude Amont)

- ✔ DGA / CELAR (Maîtrise d'ouvrage)
- ✔ **C**onception d 'Architecture **S**écurisées et **T**echniques d 'Optimisation et de **R**ationalisation.

➨Consortium

- ✔ Sycomore Aérospatiale Matra (Maîtrise d'œuvre)
- ✔ IRISA (projets LANDE et EPATR)
- ✔ TNI
- ✔ AQL - Groupe Silicomp

➨Volonté d'ouverture

- ✔ Création d 'un site CASTOR début 2001
- ✔ ….

Déroulement du projet

Trois Tranches

- ✔ T1 (année 2000):
  - Recueil et analyse des besoins
  - Maquette
- ✔ T2 (année 2001):
  - Prototype
- ✔ T3 (année 2002):
  - Outil

Plan de l'exposé

- Le besoin
- Les concepts CASTOR
- La maquette CASTOR
- La suite

Le besoin

- un outil fédérateur
- un outil de description de systèmes
- un outil de warning et d'assistance

Besoin: outil fédérateur

➡Tout le cycle de vie du système

- ✔ Décideurs
- ✔ Concepteurs
  - Conception réseau/télécom
  - Conception SI
  - Conception d'architectures sécurisés
- ✔ Evaluateurs
  - Evaluation produit
  - Evaluation système
- ✔ Auditeurs
- ✔ Utilisateurs du système

➡Différents niveaux d'expertise

AQL tni Techniques Nouvelles d'Informatique IRISA EADS SYCOMORE DGA

Besoin: outil de description de systèmes

- **Outil graphique permettant de « dessiner » le système**
  - ✔ Des boites
  - ✔ Des flèches
  - ✔ Une bibliothèque de composants
- **Suivant différents points de vue**
  - ✔ Fonctionnel
  - ✔ Physique
  - ✔ Organisationnel
  - ✔ ...

Besoin: outil de warning et d’assistance

➡Un outil de calcul

- ✔ Des fonctions d ’exploitation
- ✔ Configurable en fonction du degré d ’expertise de l ’utilisateur

➡Mais pas un outil « magique »

- ✔ Tous les calculs doivent être tracés et justifiés

➡Connecté à une base de menaces et vulnérabilités

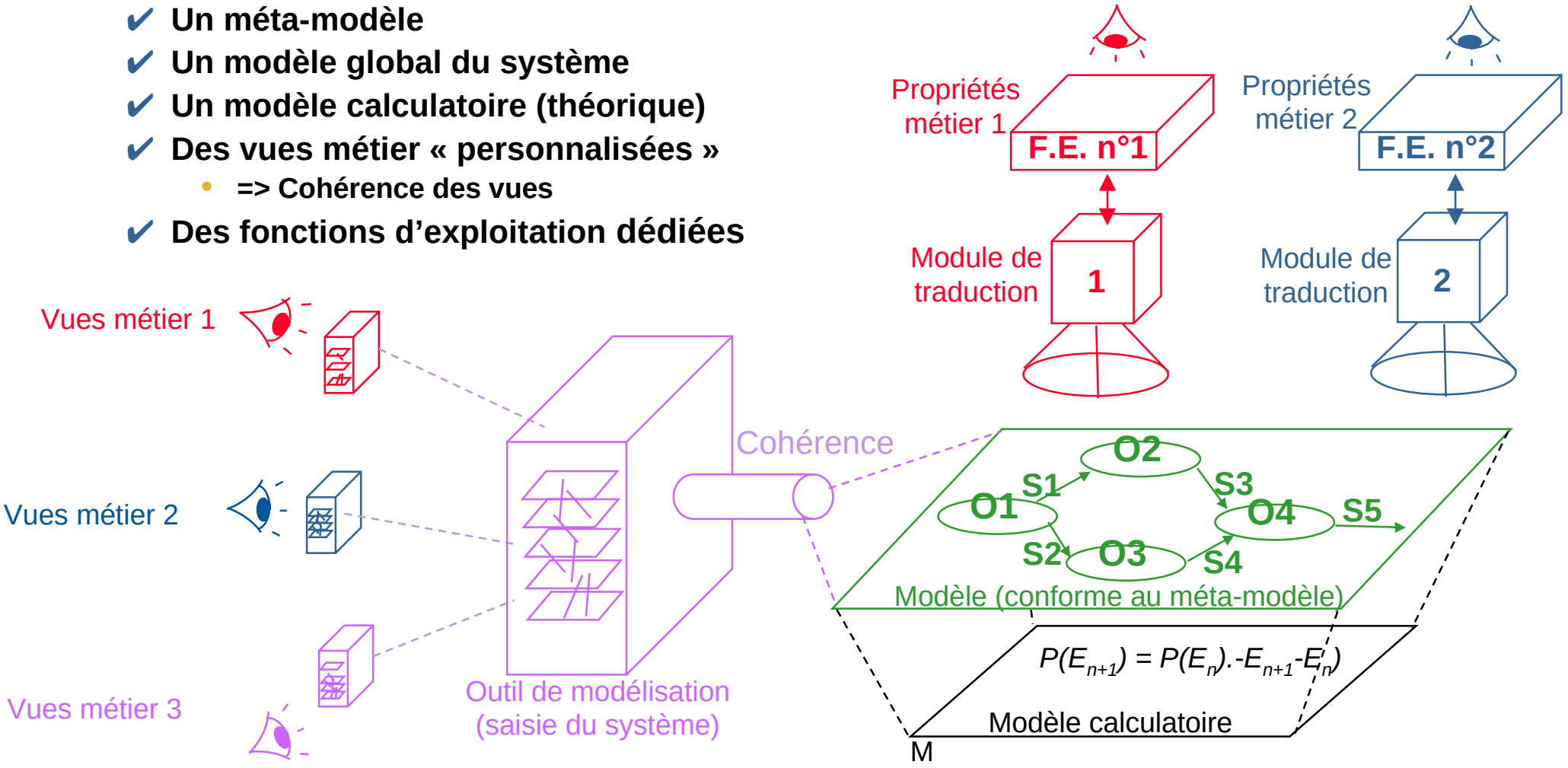
- ✔ Exemples :
  - Vulcain (autre PEA),
  - Vigil@nce (Produit AQL http://vigilance.aql.fr)

Principes de CASTOR

- Un modèle unique, commun à tous les acteurs
- Des vues dédiées (« projections »)
- Des fonctions d'exploitation dédiées

Principes de CASTOR

- ✔ **Un méta-modèle**
- ✔ **Un modèle global du système**
- ✔ **Un modèle calculatoire (théorique)**
- ✔ **Des vues métier « personnalisées »**
  - **=> Cohérence des vues**
- ✔ **Des fonctions d'exploitation dédiées**



Bases de CASTOR

➡Modèle conceptuel

- ✔ Services
- ✔ Multi-vues
- ✔ Modélisation Hiérarchique

➡Méta-modèle

- ✔ UML
- ✔ Outil d ’instanciation du méta-modèle (Techno Opentool de TNI)

➡Exploitation du modèle

- ✔ Modèle théorique (calculatoire)
- ✔ Fonctions d’exploitations dédiées
- ✔ Exportation du modèle (format XMI)

Services

➡Des « objets »

- ✔ Objets de base
  - Matériel
  - Logiciel
  - Fonction
  - Information
  - Support
  - Zone
  - Personnel
  - Organisation
- ✔ + Attributs sur les objets

➡Des « services » entre les objets

- ✔ + Attributs sur les service (Disponibilité, Conformité, ...)

Services

➡Exemples

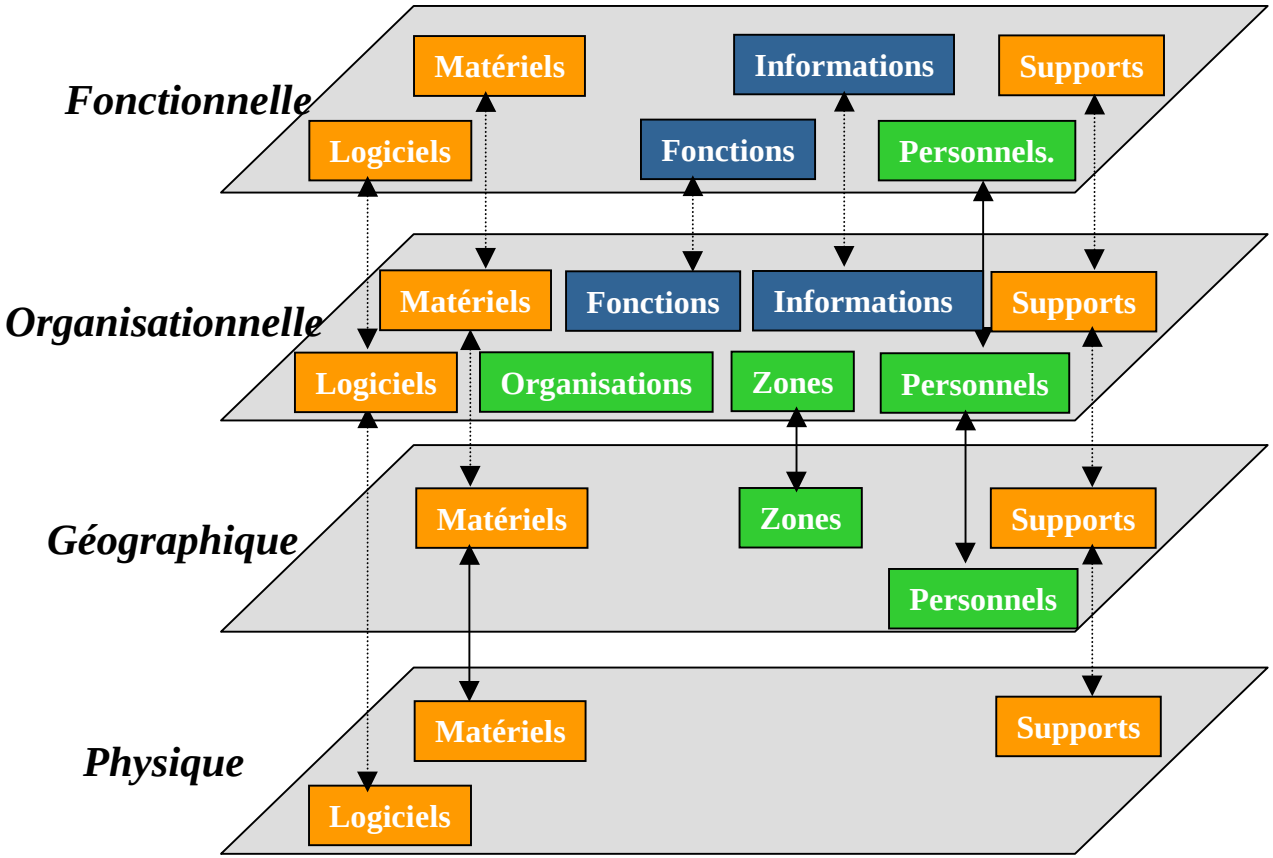
✔ Logiciel	« est administré par »	Personnel
✔ Matériel	« est hébergé par »	Zone
✔ Zone	« est surveillée par »	Personnel
✔ Information	« est inscrite sur »	Support
✔ Fonction	« est mise en œuvre par »	Matériel
✔ Fonction	« est mise en œuvre par »	Support
✔ ...		

+ services inverses (ou réciproques)

✔ Personnel	« administre »	Logiciel
✔ Zone	« héberge »	Matériel
✔ ...		

Multi-vues

- ✔ Une vue =
  - Un ensemble de services
  - Un ensemble d ’objets

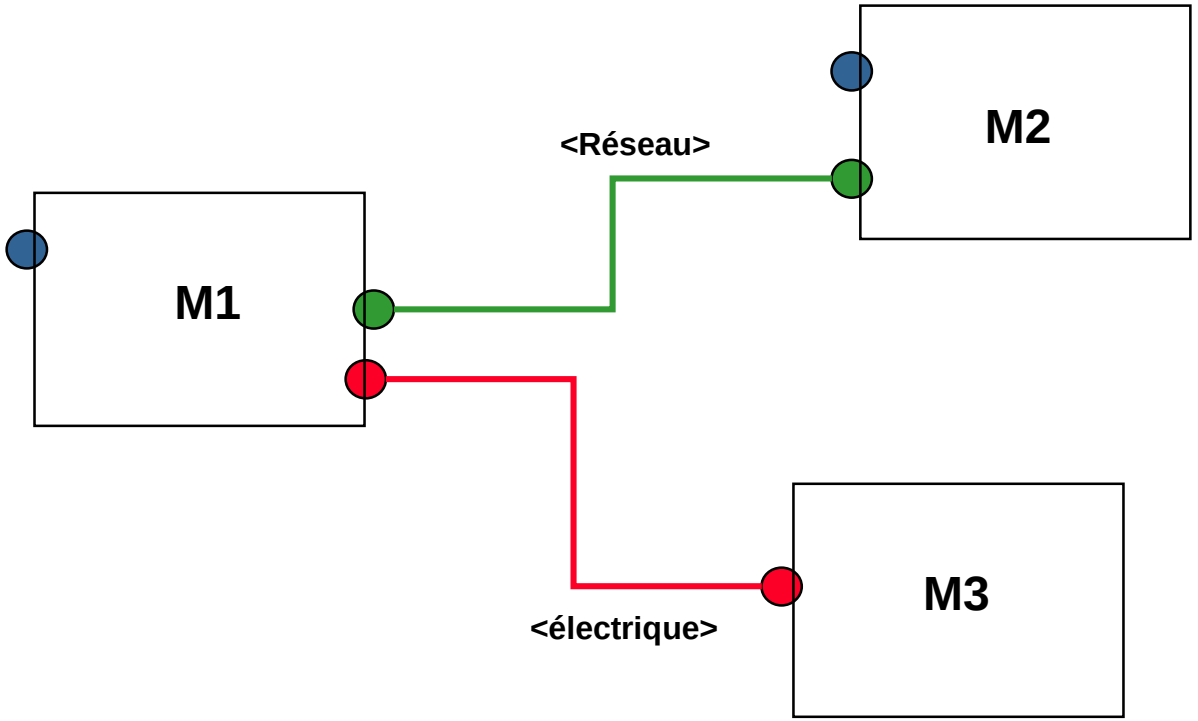


AQL tNi Techniques Nouvelles d'Informatique IRISA EADS SYCOMORE DGA

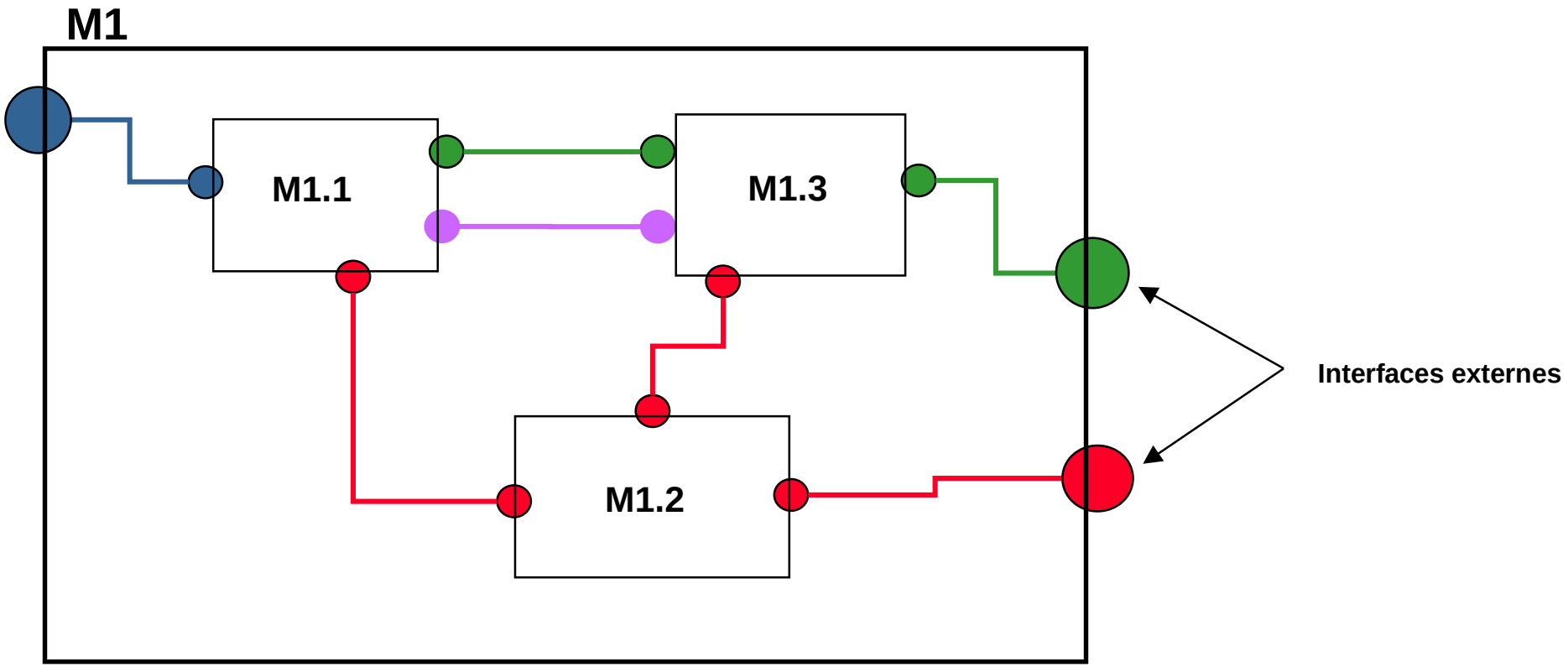
Modélisation hiérarchique

Des connexions et des connecteurs typés

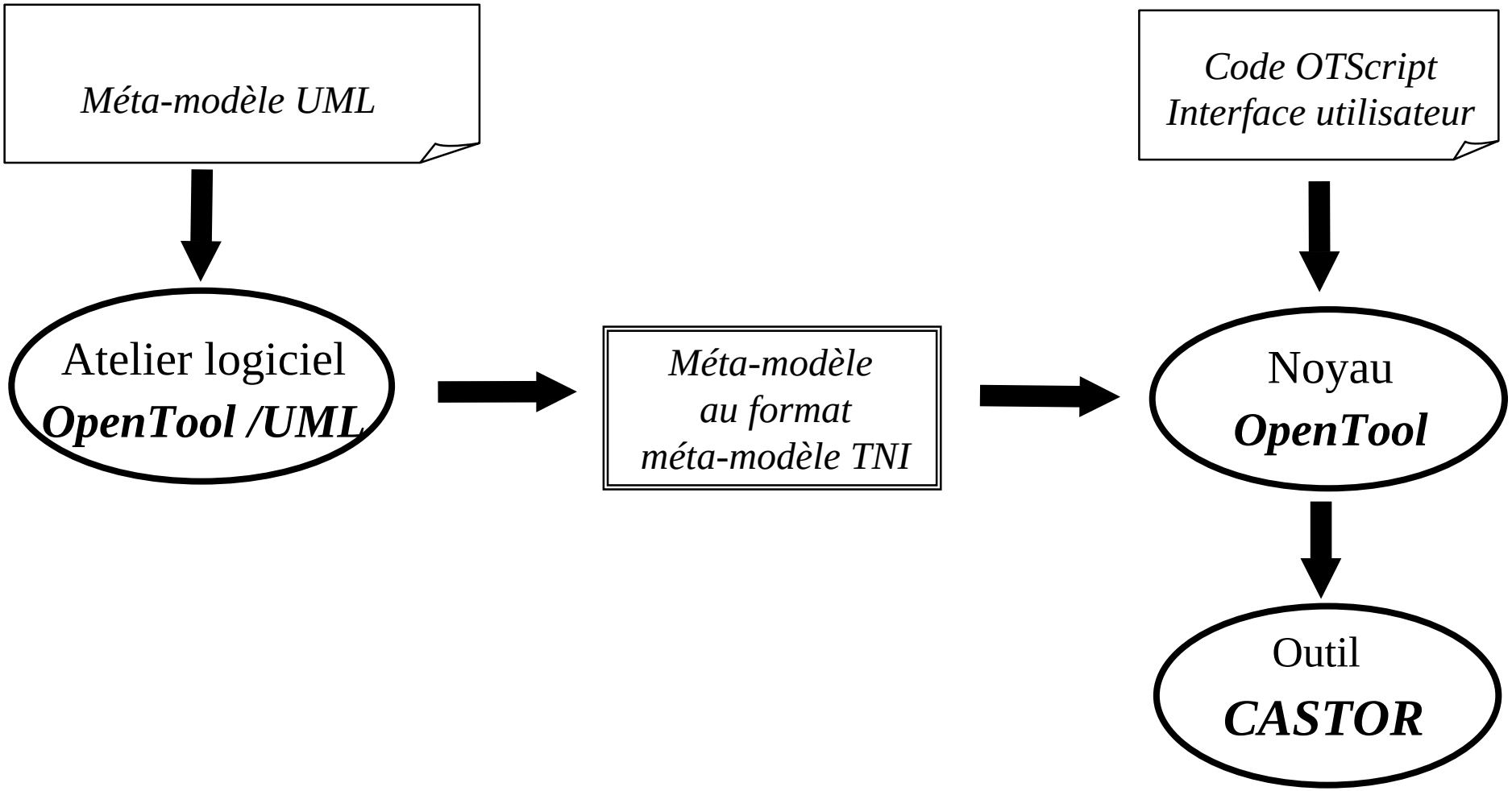
Types =
- Réseau
- Série
- Hertzienne
- électrique
- ...



Modélisation hiérarchique



Méta-modélisation



Modèle théorique

➡ Une vue = quintuplet (N, E, NK, EK, nk)

- ✔ N : ensemble de (noms de) nœuds
- ✔ NK : ensemble des types de nœuds
- ✔ EK : ensemble des types d'arcs
- ✔ E : ensemble des arcs (ek, n1, n2)
  - ek ∈ EK
  - n1 ∈ N
  - n2 ∈ N
- ✔ nk: fonction qui à chaque n ∈ N associe son type t ∈ NK

Modèle théorique

➡ Définition d'une logique pour l'expression des propriétés sur les vues

- ✔ Logique du premier ordre
- ✔ Augmentée de prédicats sur chemins

F::=	**path(PRE, NODE, NODE)**
	| ∃ x.F
	| ∀ x.F
	| F ∧ F
	| F ∨ F
	| ¬ F

PRE	**::= EDGE**
	| (PRE | PRE)
	| (PRE PRE)
	| (PRE)*
	| e

NODE ::= nom | var
EDGE ::= nom |var

AQL tNi IRISA EADS SYCOMORE DGA

Modèle théorique

➡Opérations sur les vues

- ✔ Union de vues
- ✔ projection de vue selon une propriété

➡Cohérence de vues

- ✔ La cohérence est une collection de propriétés exprimées dans la logique définie précédemment.

➡Vues hiérarchiques

- ✔ Introduction d ’un nouveau lien de type *raf*
- ✔ Tous les nœuds sont reliés à leur composant abstrait par un arc de type *raf*.

Modèle théorique

➡ Implémentation en DATALOG

- ✔ Restriction de prolog conçue comme un langage d'interrogation de bases de données déductives
- ✔ Ici
  - Base de donnée = les vues
  - Requêtes à la base = les propriétés

La maquette

➡Description de systèmes: 5 types de vues

- ✔ Physique (modélisation hiérarchique)
- ✔ Fonctionnelle
- ✔ Organisationnelle
- ✔ Géographique
- ✔ Données

➡Bibliothèque de composants

➡Vérification de cohérences simples

La maquette

➡Cohérences imposées par construction (par le méta-modèle)

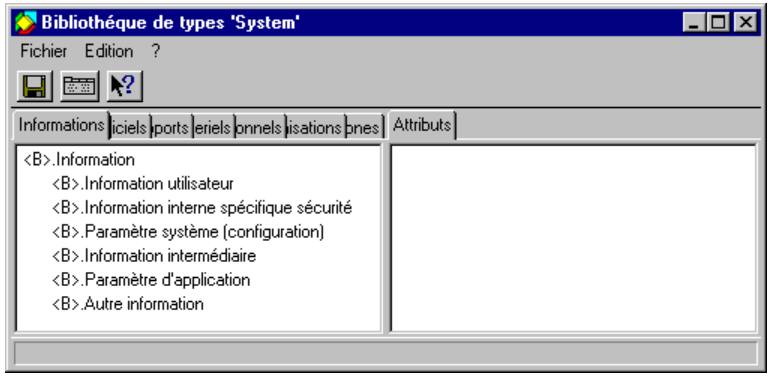
- ✔ **Utilisation de la multiplicité**
- ✔ **Exemples**:
  - Une connexion est reliée à 2 matériels
  - le service « est hébergé par » est relié à une zone

➡Cohérences non-imposées (Warning)

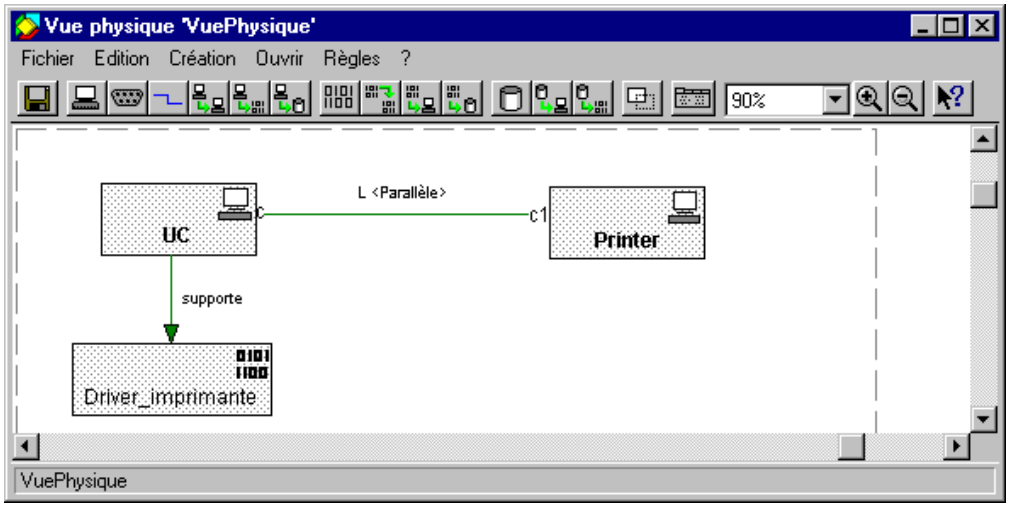
- ✔ **Outil de calcul**
- ✔ **Exemples de règles de cohérence:**
  - un Logiciel « Est supporté par au moins un » Matériel
  - une Information « Est inscrite sur au moins un » Support
  - un logiciel « Est administré par au moins un » Personnel
  - ...

La maquette

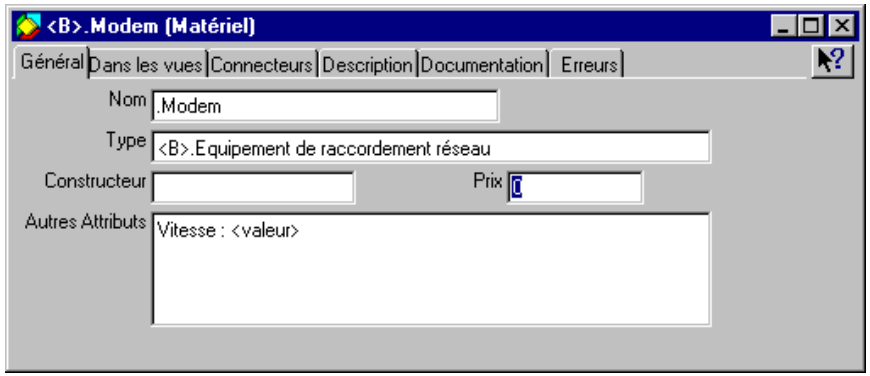
Browser



Diagramme



Formulaire



La suite (T2 et ...)

➡Menaces - Vulnérabilités

- ✔ Attributs ou objets
- ✔ Lien avec une base de vulnérabilité

➡Modélisation hiérarchique

- ✔ Pour d 'autres vues (Fonctionnelle, Géographique, ...)
- ✔ Intégration de la notion de service (Factorisation / Distribution, …)

➡Fonctions d'exploitation

- ✔ Vérification de cohérence (plus poussée)
- ✔ Calcul de disponibilité (Qualitatif dans un premier temps)
- ✔ Périmètres de sécurité (Restitution après une attaque…)
- ✔ Scénario d'attaques
- ✔ Analyse de risques (Méthode à Définir, …)
- ✔ ...

La suite (T2 et ...)

➨Concepts d'emploi (Fonction des métiers)

- ✔ Adéquation des types de vues
- ✔ Adéquation des fonctions d'exploitation (expert / non-expert)

➨Importation

- ✔ De modèle existants (Sous Access, Excel, …)

➨Exportation

- ✔ Vers des outils de bureautique
- ✔ Vers d 'autres outils d 'exploitation

Questions