Journées RNTL

4-5 Octobre 2004

Projet MORSE

Méthodes et **O**utils pour la **R**éalisation et la vérification formelle de **S**ystèmes interopérables **E**mbarqués critiques

VELU Jean-Pierre
Direction Technique et Scientifique
SAGEM SA
jean-pierre.velu@sagem.com

Sommaire

- **Introduction**
  - Objectif du projet
  - Le consortium
- **L'organisation en sous-projets**
- **Conclusion**
- **Les livrables de la première année**

Introduction : Objectif du projet

- **Fournir une méthodologie et des outils prototypes pour le développement d'applications industrielles certifiables caractérisées par les contraintes suivantes :**
  - Exigences de fiabilité
  - Nécessité de certifier l'application
  - Exécution dans un environnement réparti
  - Communications asynchrones
  - Utilisation de standard (UML)
- **Application au domaine de l'informatique embarquée et distribuée**

Introduction : Le consortium

Le projet a été notifié le 1er juillet 2003

- **SAGEM** est un pionnier dans le monde des drones et possède un savoir-faire dans la certification des systèmes aéronautiques
- **AONIX,** éditeur logiciel, est un fournisseur d'AGL et de solutions pour le développement de systèmes critiques
- **Le LIP6** (thème Systèmes Répartis et Coopératifs) est un expert dans la modélisation et la vérification formelle de systèmes répartis interopérables
- **Le LaBRI** est un expert dans les techniques de vérification formelles symboliques

L'Organisation en sous-projets

1 Méthodes

2 Modèle

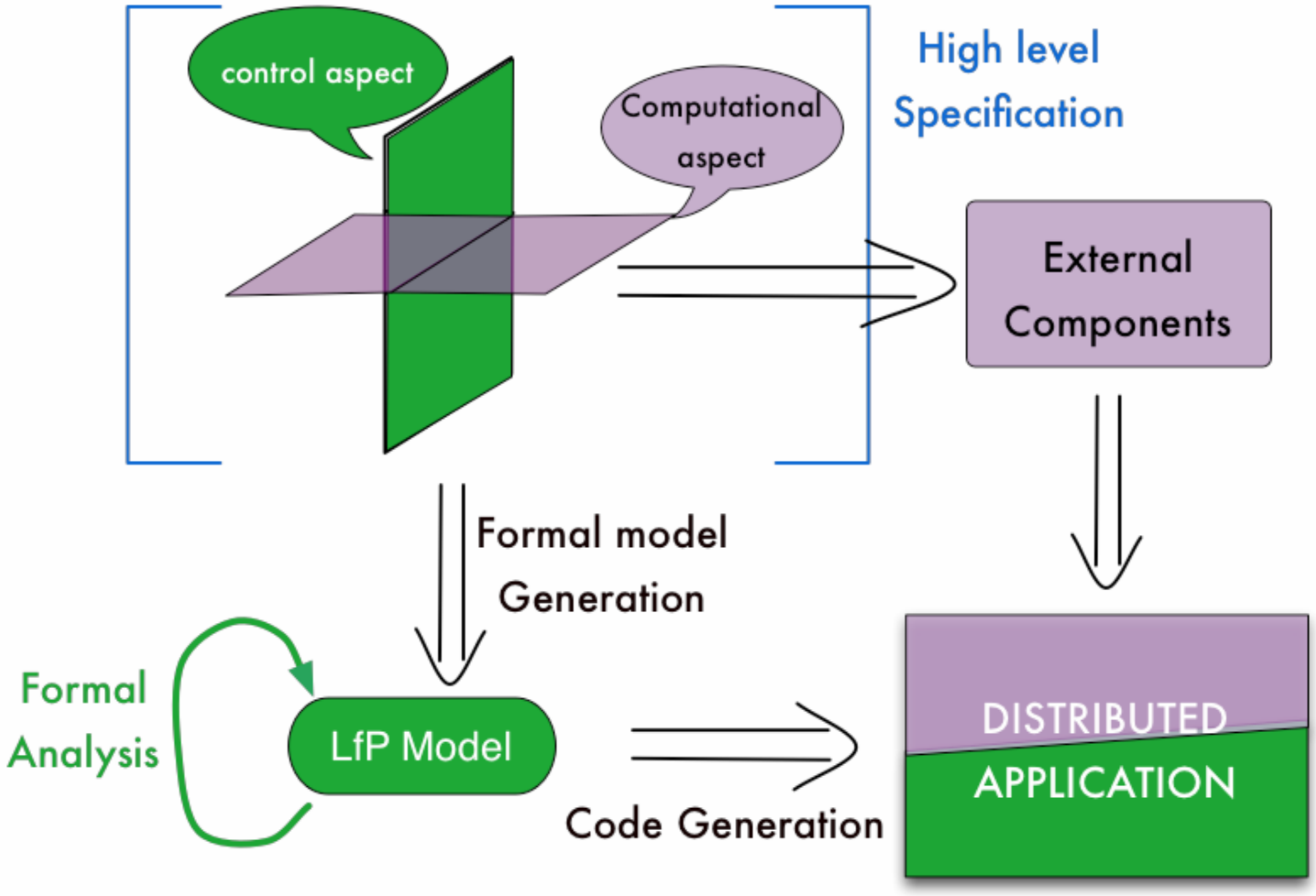
Conduite : Le projet est « tiré » par des exemples de plus en plus compliqués

3 Vérification

4 Prototypage

5 Application témoin

SP-1 : Méthodologie de développement de systèmes répartis embarqués asynchrones



SP-2 : Langage pivot et expression de propriétés sur le modèle (1)

- **Quelques caractéristiques du langage Pivot :**
  - **Les modèles sont structurés**
    - Basés sur des automates hiérarchiques
    - Des composants-types (média/classes)
    - Jeu d'instructions approprié à la modélisation de protocoles
  - **Les modèles sont transparents vis-à-vis de la distribution**
    - La spécification du comportement n'est pas liée au déploiement
  - **Il existe 4 sortes de diagrammes**
    - **Le diagramme d'architecture**
      - Aspects statiques => initialisation de l'application
    - **Le diagramme de comportement**
      - Aspects dynamiques => comportement des composants
    - **Le diagramme des propriétés**
    - **Le diagramme de déploiement**

SP-2 : Langage pivot et expression de propriétés sur le modèle (2)

- **Dans le futur atelier MORSE, le modèle est décrit en UML2.0, puis transformé en LfP**

- **Le concepteur utilise un *profil UML*, qui définit**
  - le sous-ensemble d'UML à utiliser,
  - des restrictions, des règles de modélisation, des extensions,
  - des enrichissements sémantiques ajoutant à UML certaines capacités d'expression de LfP.

- **Un modèle LfP équivalent au modèle UML est généré automatiquement**

SP-3 : Vérification formelle des propriétés (1)

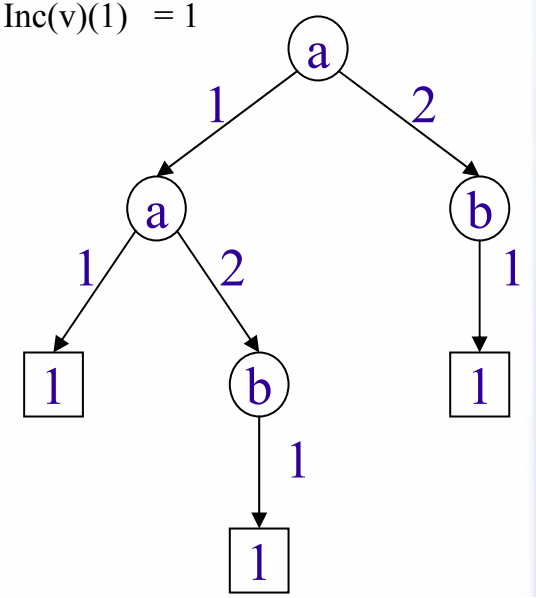
- **Représentation des états accessibles du modèles sous forme de Data Decision Diagram (DDD)**

  - On définit les *homomorphismes* associés aux opérateurs du langage lfp
    - Evaluation des préconditions
    - Envoi de messages
    - Consommation de messages
    - Traitement des transitions avec priorité
    - Insertion/destruction d'instance

  - *Model checking* symbolique de LfP basé sur les DDD
    - Calcul avant de l'ensemble des états accessibles avec prise en compte des priorités des transitions
    - Vérification des propriétés de sûreté
    - Vérification LTL et CTL en cours de développement

$$\mathrm{Inc}(v)(e,x) = \begin{cases} v \xrightarrow{x+1} \mathrm{Id} & \textit{si } v=e \\ e \xrightarrow{x} \mathrm{Inc}(v) & \textit{sinon} \end{cases}$$

$$\mathrm{Inc}(v)(1) = 1$$



SP-3 : Vérification formelle des propriétés (2)

- **Plusieurs expérimentations ont été faites :**
  - L'application serveur/client
  - Synchronisation de trains
- **Des gains notables (~10x) ont été obtenus en réduisant la complexite de l'évaluation de deux façons:**
  - Réduction du nombre de parcours dans l'arborescence
  - Utilisation de DDD hiérarchiques
- **Quelques problèmes :**
  - Le modèle de définit pas la notion d'équité
  - Le modèle n'introduit pas encore la notion de temps
- **Optimisations possibles**
  - Stratégies paresseuses d'évaluation
  - Ordonnancement des variables
  - Optimisations symboliques

SP-4 : Prototypage Synthèse automatique de programmes (1)

- **La génération de code**
  - Elle traduit une spécification LfP en un programme (java)
    - ✓ Etats => "*if then else*" ou un message *wait*
    - ✓ *Call back à la runtime pour :*
      - Les opérations sur les messages
      - Les instantiations dynamiques
      - La destruction des composants
    - ✓ Les instructions standard sont traduites en leur équivalent Java
    - ✓ Les appels aux composants externes à LfP sont implémentés par des appels de méthode.

- **La *runtime* associée réalise principalement :**
  - Le management des objets (instantiation, localisation et destruction)
  - La communication par messages entre les composant
  - Le management des *threads*

Conclusion

- **Premiers résultats acquis sur des toy-examples**
  - Génération automatique de code (non embarquable)
  - Premiers essais de vérification formelle
  - Grandes lignes du profil UML2.0 définies

- **Poursuite des travaux**
  - Définition d'un exemple d'un ordre de complexité supérieur
  - Génération de code pour une plate-forme embarquable
  - Optimisation des outils de vérification
  - Automatisation de la traduction UML2.0 vers LfP

Livrables de l'année 1

- **Sous-projet.1**
  - Site Web MORSE **http://morse.lip6.fr**
  - Doc MORSE-CR-030710-V1.1-JPV
  - Doc MORSE-CR-031114-V1.1-JPV
  - Doc MORSE-CR-031218-V1.1-JPV
  - MAN-METHODOLOGIE-V1.2
  - Doc NOTE031114.1/1
  - SPEC-UML_LfP-V1.0
- **Sous-projet.2**
  - Description de la sémantique du langage LfP
  - BNF de la grammaire des attributs des diagrammes LfP
  - SPEC-METAMODEL_LfP-V1.0
- **Sous-projet.3**
  - MORSE_TRAV32_040628_V1_FB
  - SPEC-VERIF_LfP_DDD-V1.0
  - MORSE_TRAV33_040629_V1_FB
  - IMPL-BIBLIO_DDD-V1.
- **Sous-projet.4**
  - IMPL_CODEGEN_Lfp-V1.0