

SDL: 20 ans de programmation basée modèle

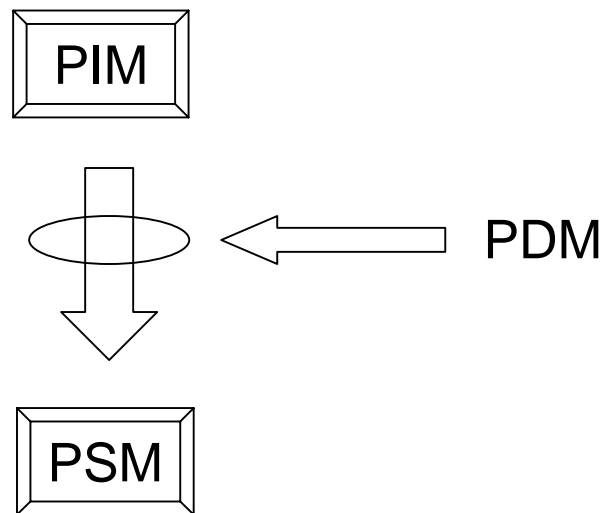
Emmanuel Gaudin

emmanuel.gaudin @ pragmadev.com

Principes

- MDE, MDA et MDD: Approche orienté modèle
 - PIM: Platform Independant Model
 - PDM: Platform Definition Model
 - PSM: Platform Specific Model

Principes



L'intérêt de la démarche est de transformer le modèle abstrait (PIM) à l'aide de la définition de la plateforme (PDM) pour obtenir un modèle concret (PSM).

PIM

- Le modèle abstrait doit être indépendant de la plateforme d'implémentation,
- Le modèle abstrait doit pouvoir être traduit sur une plateforme d'implémentation,
- Pour cela on s'appuie généralement sur une machine virtuelle qui offre:
 - un certain nombre de services,
 - une sémantique suffisamment forte pour être expressive.

SDL: Specification and Description Language

Le **SDL** (Specification and Description Language) est un standard de l'ITU (International Telecommunication Union) émanation de l'ONU sous la référence Z.100.

- Objectif est de décrire de manière non ambiguë les protocoles de télécommunications afin d'assurer l'interopérabilité des équipements,
- Utilisé par l'ETSI (European Telecommunications Standards Institute) pour décrire les protocoles de télécommunications (X25, GSM, UMTS...),
- SDL est mis à jour tous les 4 ans, on admet que la première version mature du langage date de 1988.

SDL: Specification and Description Language

- Les modèles SDL sont indépendants de l'implémentation,
- Le SDL définit une sémantique forte,
- Le SDL intègre des types de données abstraits,
- Orienté temps réel événementiel,
- Graphique,
- Orienté objet depuis sa version de 1992: SDL'92.

SDL: Specification and Description Language

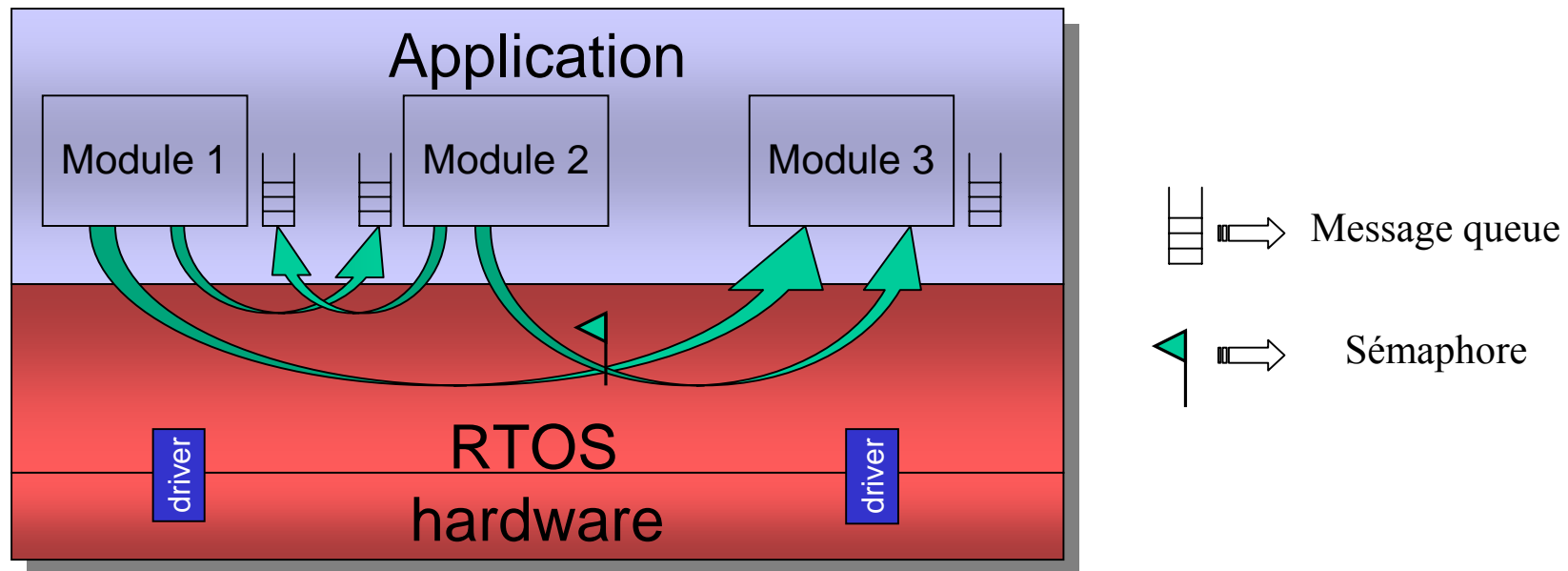
- Les modèles SDL peuvent être informels (une action peut être décrite en langage naturel)
- Les modèles SDL peuvent être incomplets (les opérateurs sont définis à l'extérieur du modèle)
- Les modèles SDL peuvent introduire une certaine forme d'indéterminisme (opérateur ANY)
- Les modèles SDL peuvent être formels (complet et non ambiguë)

SDL: Specification and Description Language

- La sémantique et les services intrinsèques au langage permettent de définir une *SDL virtual machine* qui répond à la définition d'un Modèle Indépendant de la Plateforme (PIM).
- La définition des interfaces des opérateurs et l'implémentation des services SDL définissent le modèle de la plateforme (PDM),
- Il est alors possible de générer le modèle d'implémentation spécifique (PSM) comme par exemple la combinaison d'un langage d'implémentation (C/C++), d'un exécutif temps réel (RTOS), et de drivers sur la cible (opérateurs).

Domaine d'utilisation

- Hormis les protocoles de télécommunications, les concepts du SDL sont proches de ceux que l'on retrouve dans les systèmes temps réel embarqués et en particulier sur les real Time Operating System (RTOS)
- Décomposition en tâches s'exécutant en parallèle
- Communication par :
 - échange de messages
 - interruptions
 - appel de fonctions
 - sémaphores



Gains du SDL

Des années de pratique ont permis de mesurer quantitativement les gains apportés par l'utilisation du SDL

- Code C: 35 à 50 fautes par 1000 lignes
- Code **SDL**: 8 fautes par 1000 lignes
- Réduction globale du cycle de développement de l'ordre de **35%**
 - Réduction de près de 50% dans la branche gauche du cycle en V
 - Réduction moindre dans la branche droite du fait du décalage avec la réalité technique

UML: Unified Modeling Language

- L'**UML** (Unified Modeling Language) standardisé par l'OMG (Object Management Group).
 - Adapté à tous types de systèmes,
 - Graphique,
 - Utilisé à très haut niveau d'abstraction,
 - Informel, i.e. nécessite un autre langage pour le codage (C, C++, Java, SDL),
 - Très orienté objet.

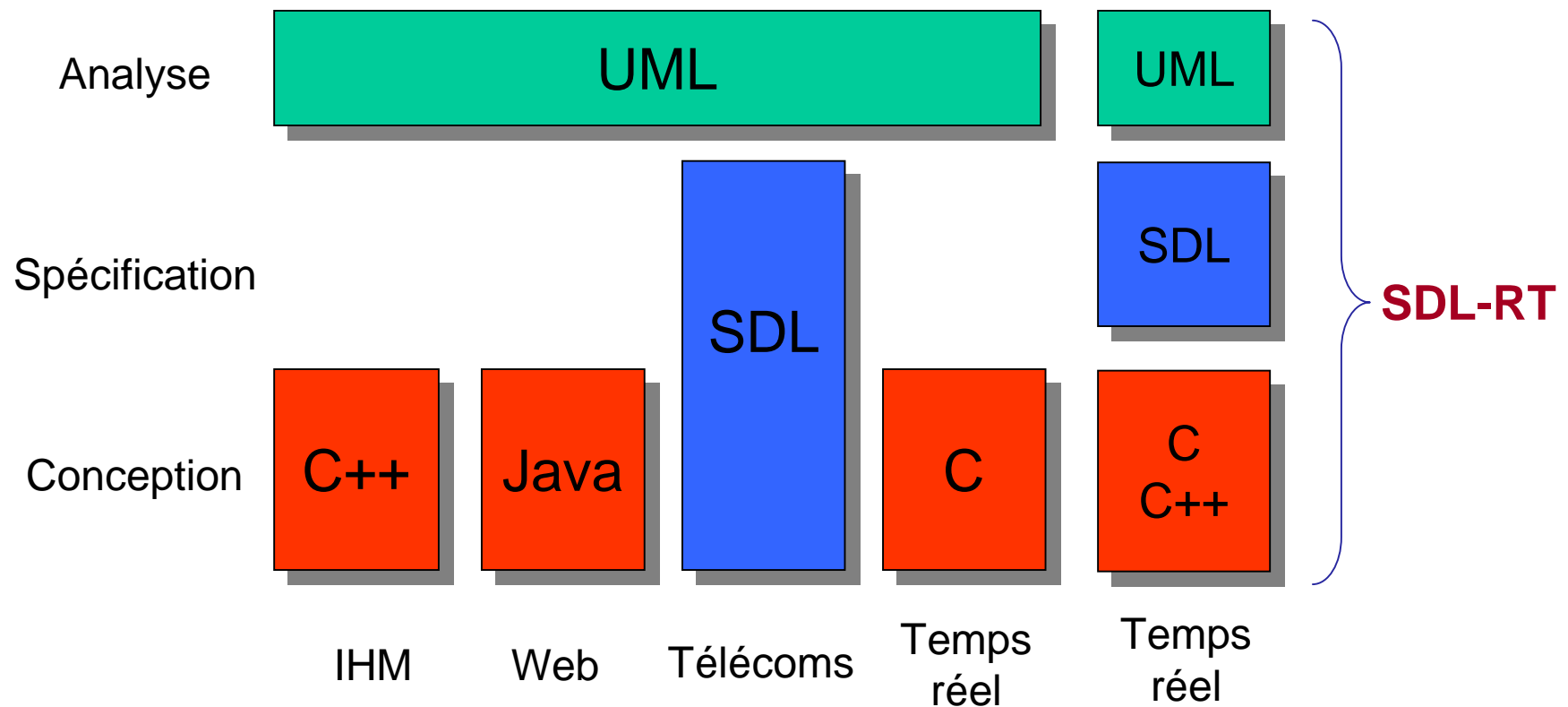
Pas de spécificité temps réel UML

- UML 1 trop générique pour être un PIM,
 - Pas de représentation UML des concepts temps réel classiques: tâches, sémaphore, message, timer...
 - Pas de sémantique,
 - Pas de types de données,
- UML 2 permet de définir un profil pour un domaine d'application utilisable pour modéliser un PIM.

Compatibilité UML

- Il n'existe pas de profil UML 2.0 standard pour le temps réel,
- Les profils UML 2.0 ne sont pas compatibles,
- L'ITU est en cours de définition d'un profil UML 2.0 standard pour les télécommunications basé sur le SDL,
- SDL est ou sera un profil UML 2.0 standard pour les télécommunications.

Positionnement des langages





specification & description language - real time

SDL-RT

- Permet de généraliser l'utilisation de SDL à toutes les applications temps réel s'appuyant sur un RTOS,
- Lisible et basé sur un format de stockage textuel normalisé (XML),
- Proposé à l'ITU pour intégration dans le SDL,
- Est un profil UML pour les applications temps réel.



Diagramme de classe UML et SDL-RT

Relations
entre les
classes
passives
et
actives

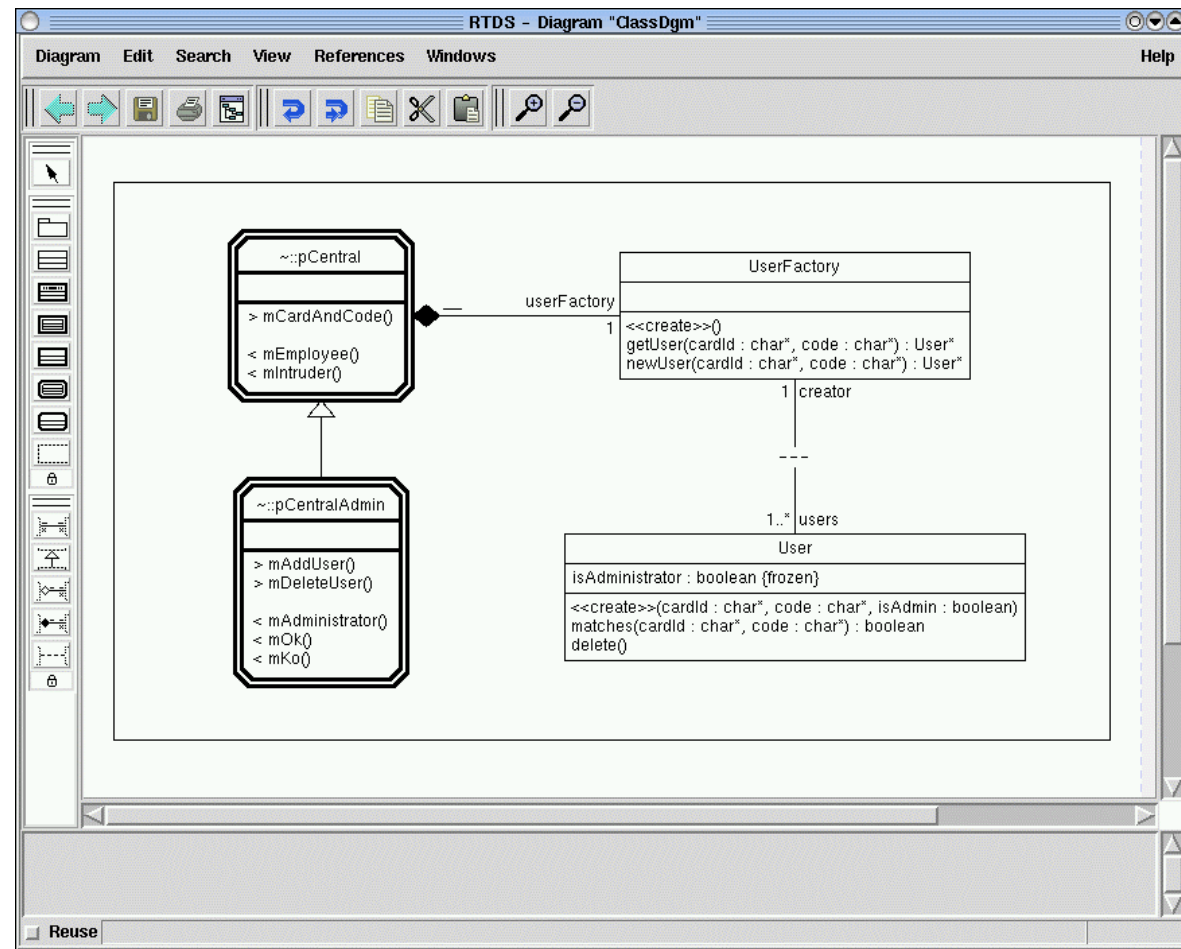
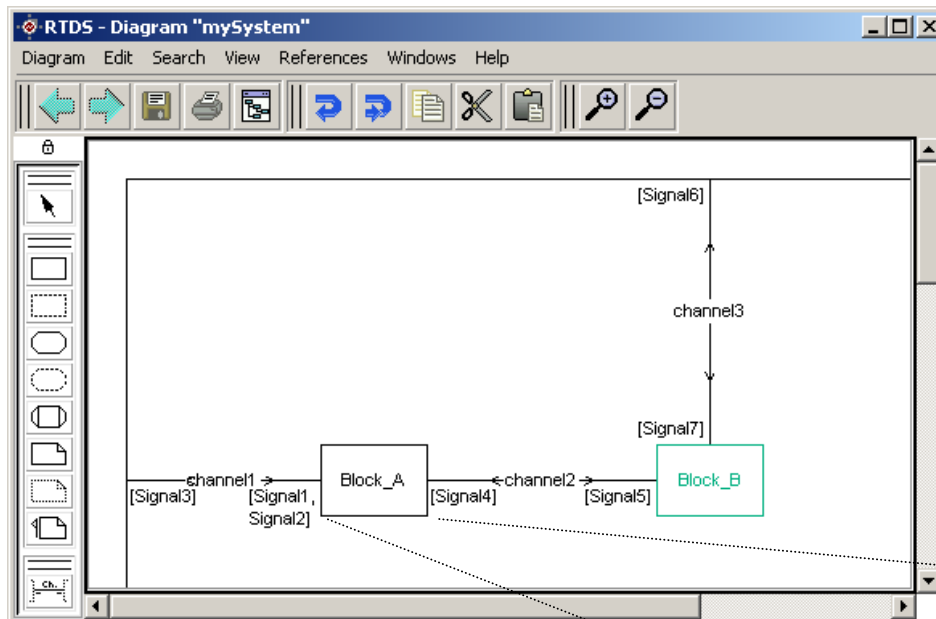


Diagramme d'architecture SDL, UML, et SDL-RT



Architecture
et
Communication

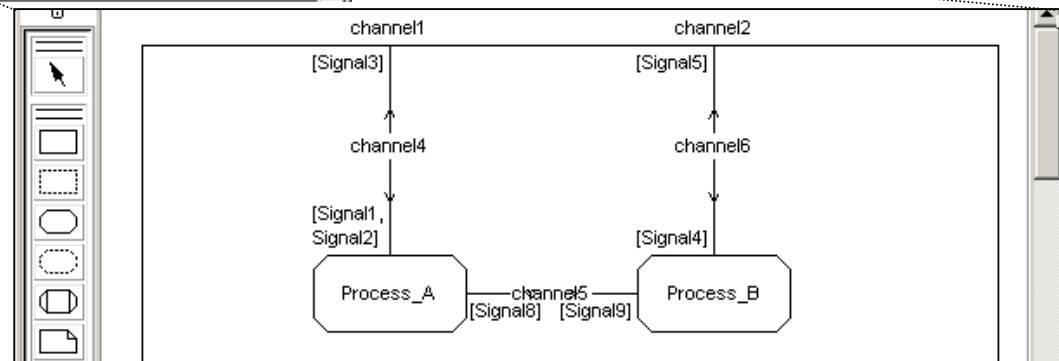
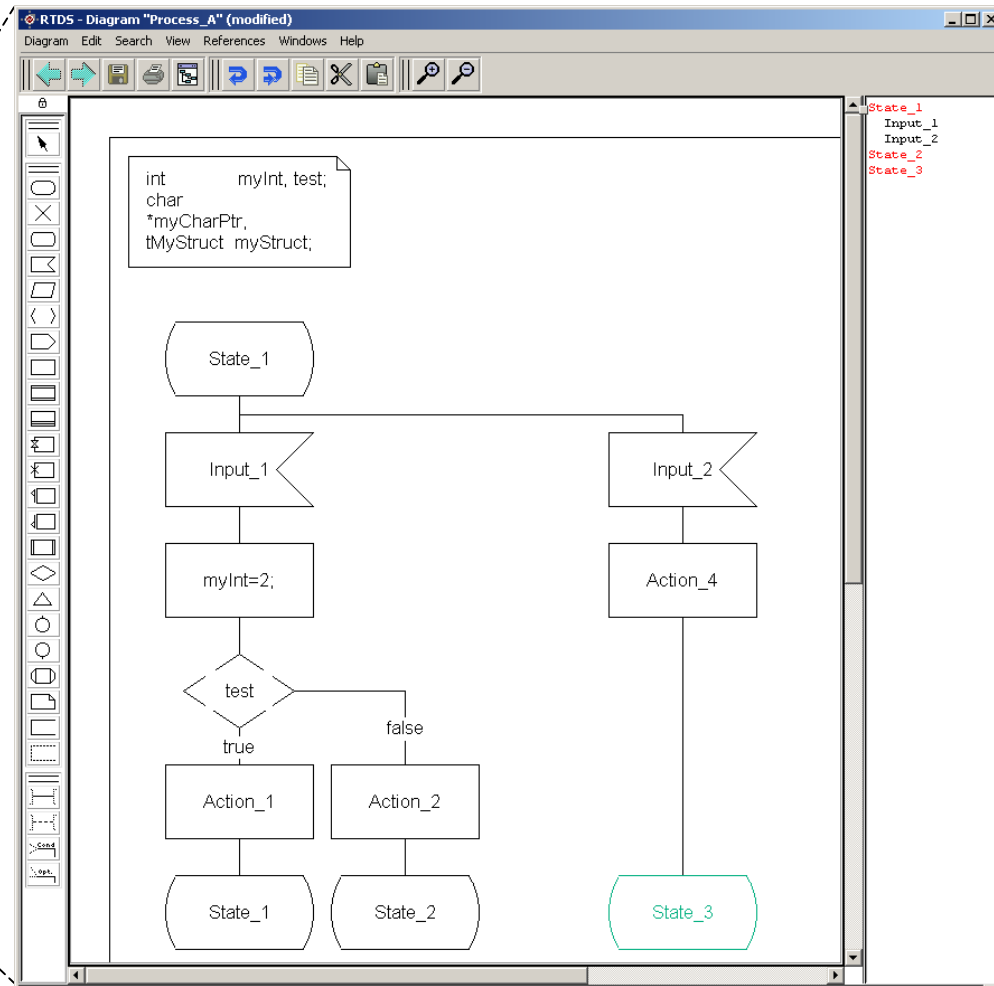


Diagramme de comportement SDL et SDL-RT

Vues
Comportement
et
Données

Process A



Extension temps réel: SDL-RT

Autre exemple
de
comportement

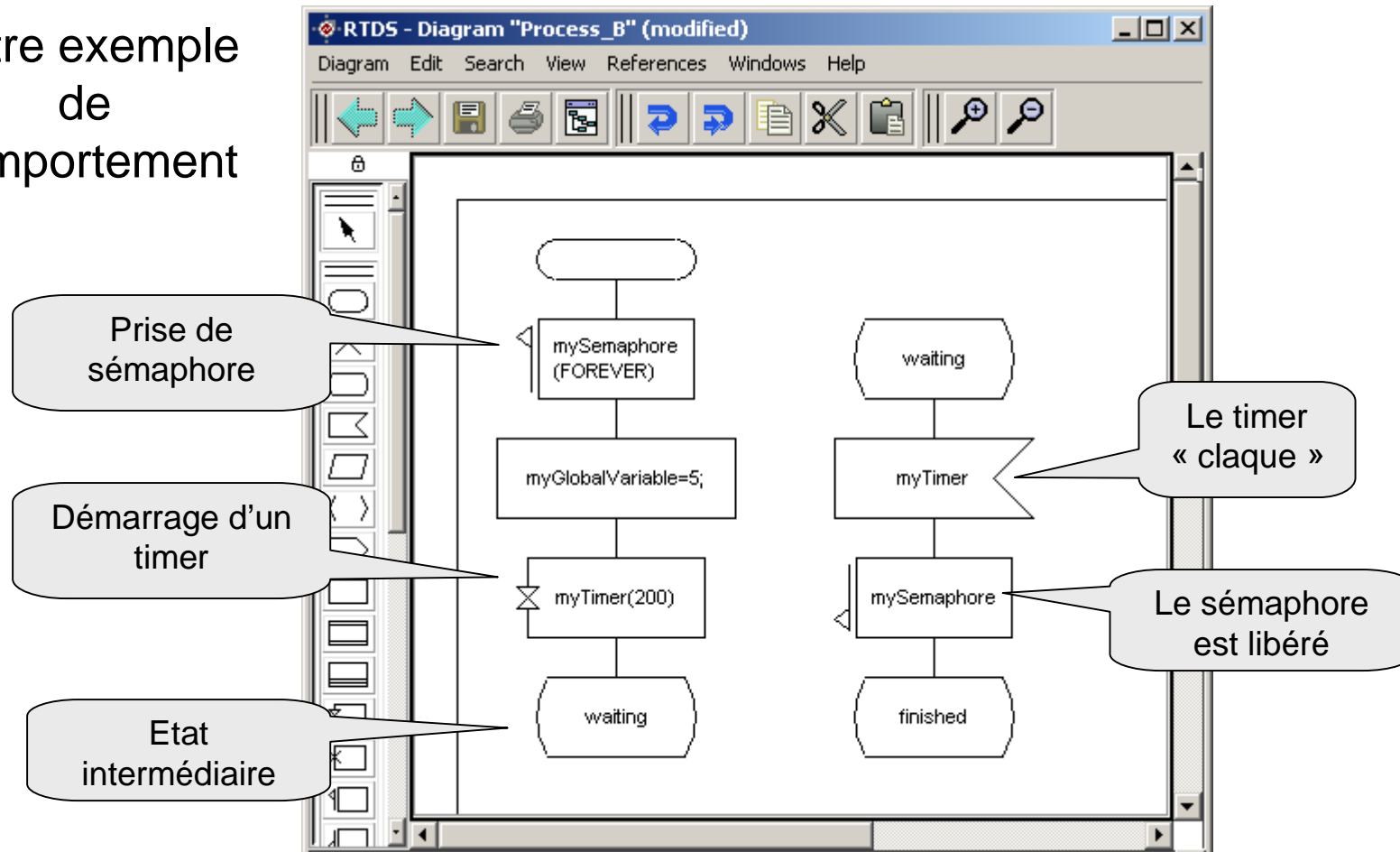
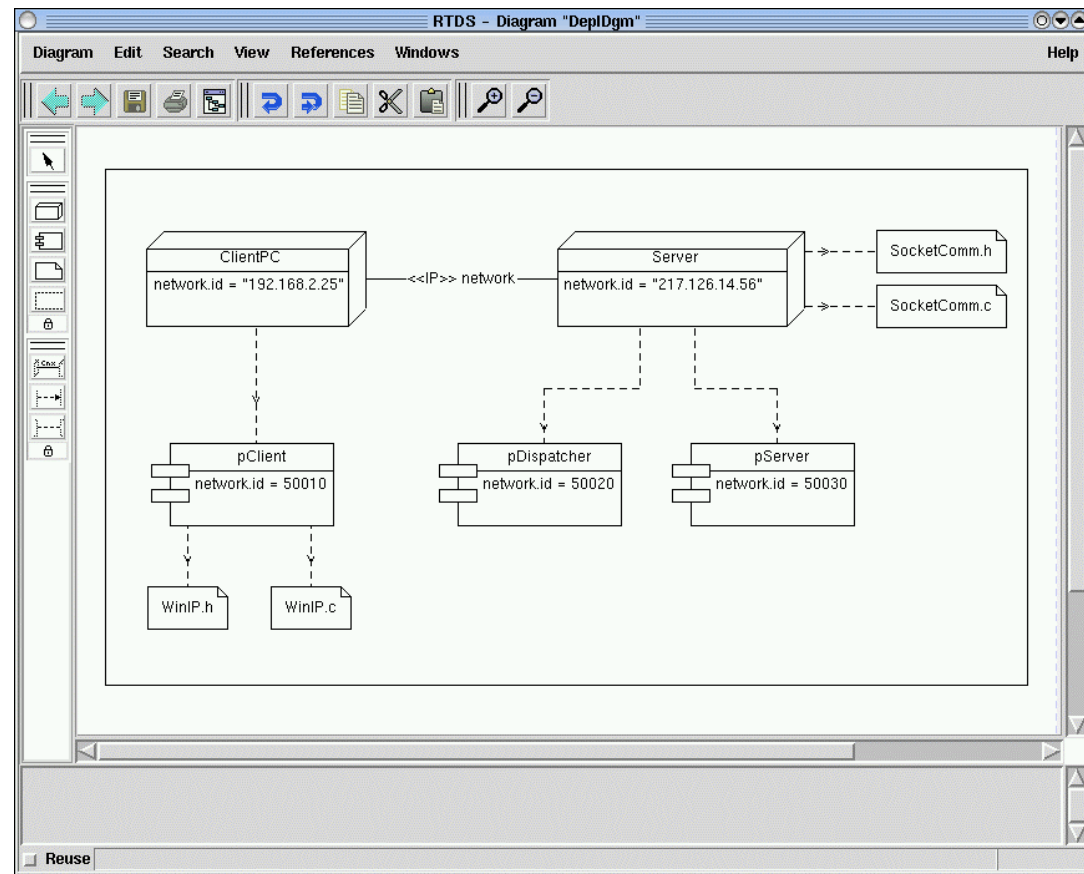
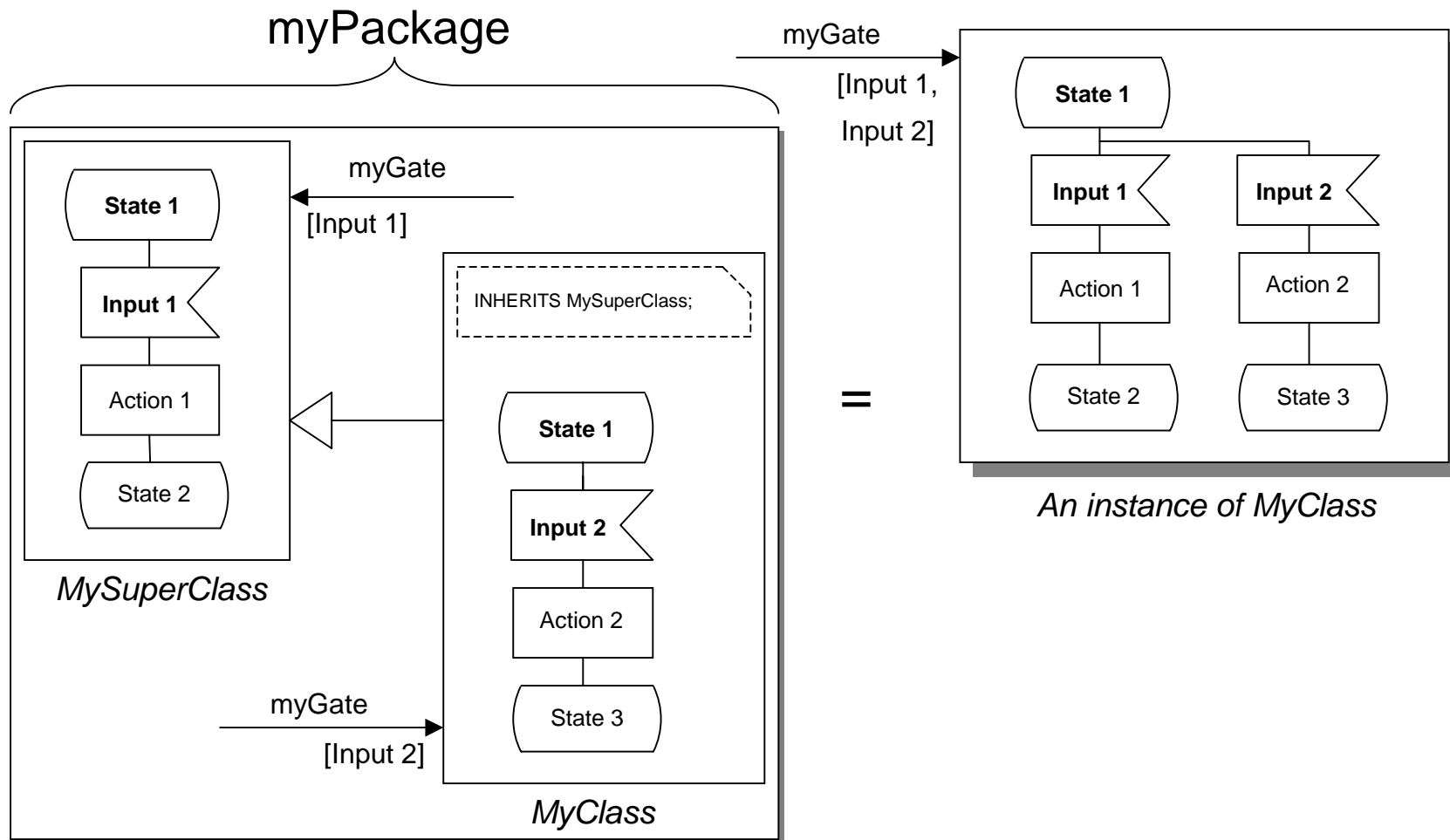


Diagramme de déploiement: UML et SDL-RT

Vue
déploiement
physique

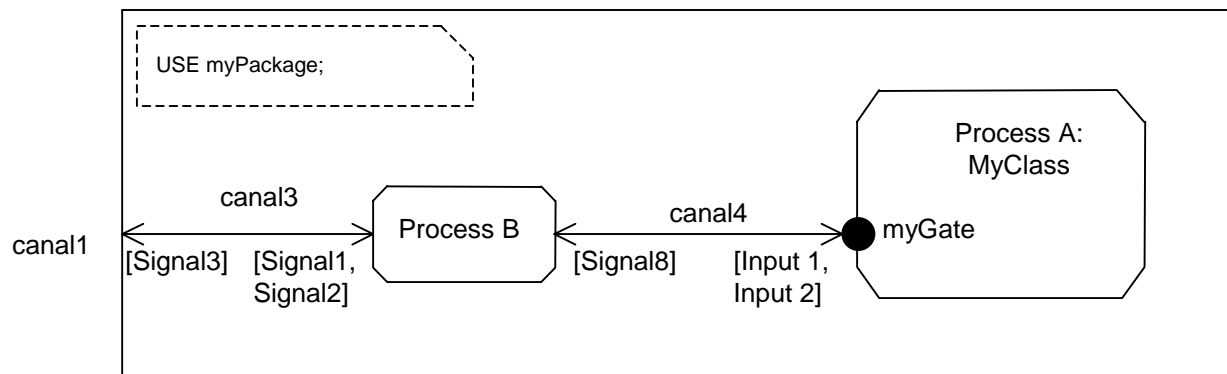


Orientation objet - exemple



Orientation objet – exemple suite

L'instance de la classe s'intègre dans l'architecture du système



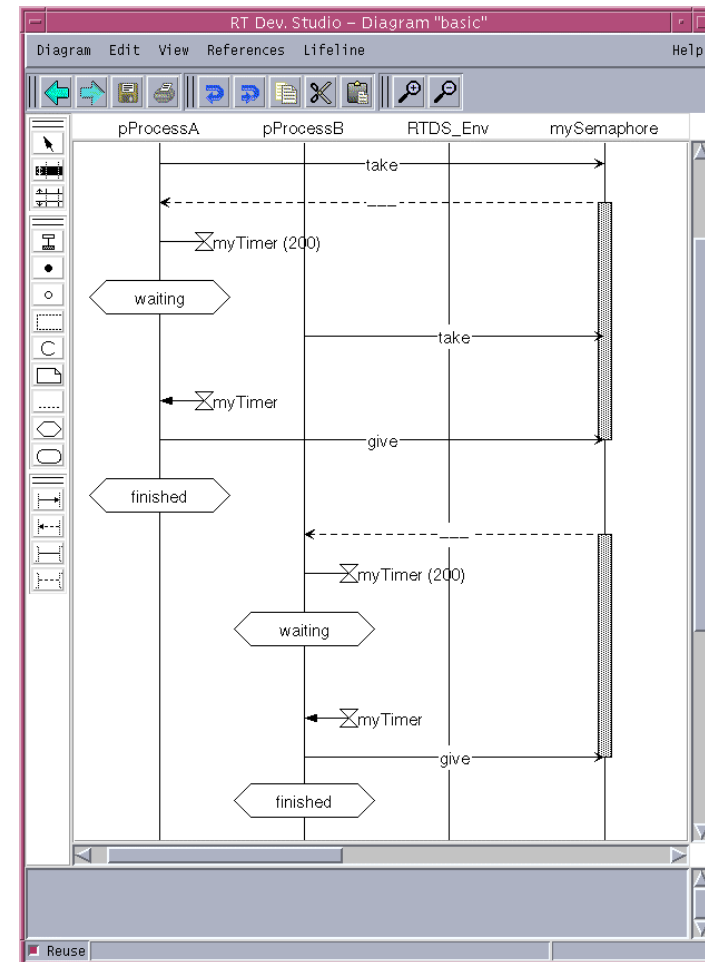
SDL-RT MSC: une vue dynamique

SDL-RT Message Sequence Chart

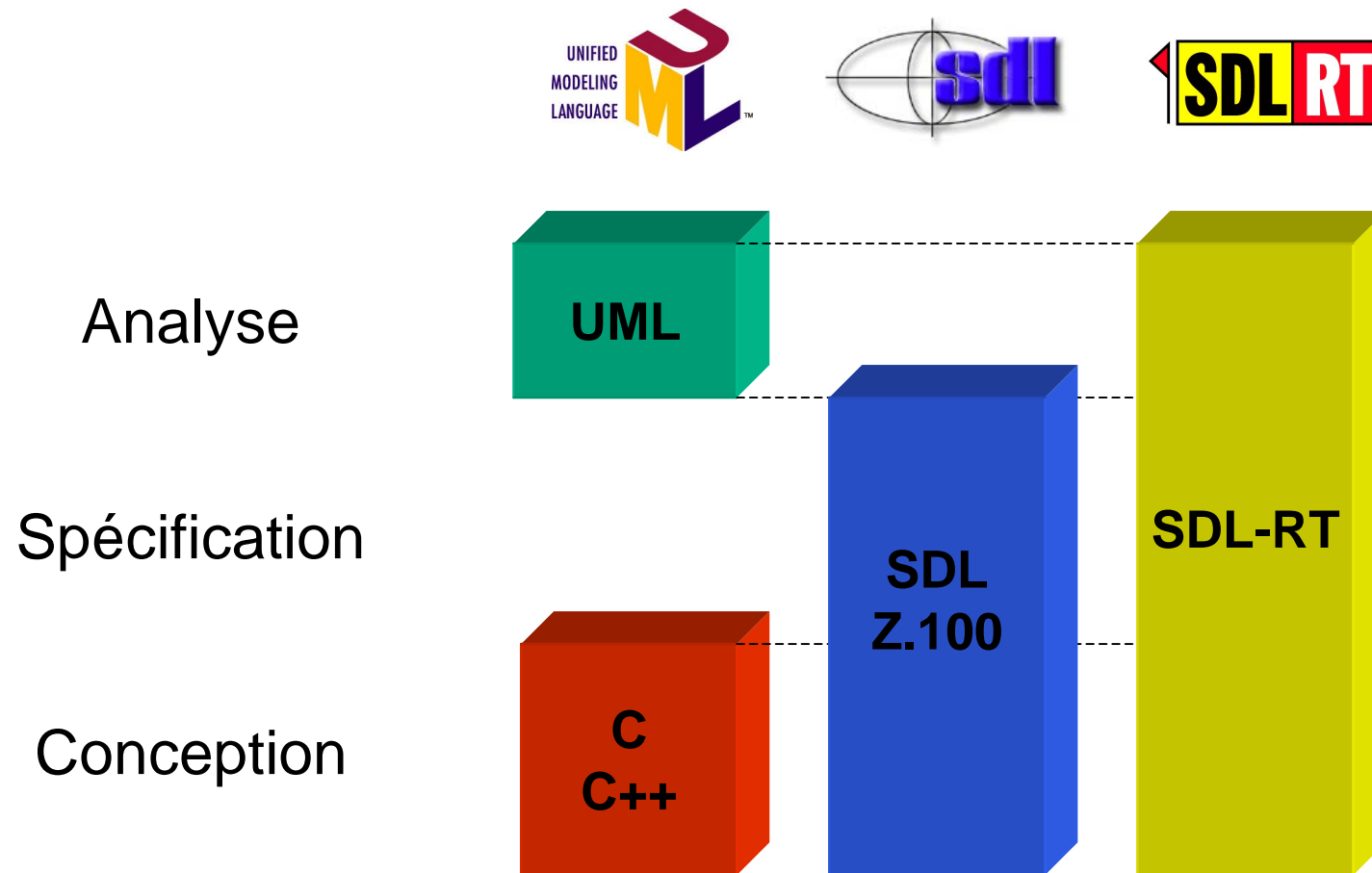
- Les lignes verticales représentent une tâche, l'environnement ou un sémaphore,
- Les flèches représentent des échanges de messages, une manipulation de sémaphore ou de timer.

Utilisations en amont et en aval:

- Spécification dynamique sous forme de scénario
- Trace d'exécution



RTDS: les langages supportés



Conclusion

- SDL permet le développement basé modèles depuis plus de 20 ans dans le domaine des télécommunications
- SDL-RT élargit le domaine à toutes les applications temps réel s'appuyant sur un RTOS
- Les profils UML 2.0 ne sont pas compatibles
- Il n'existe pas de profil UML 2.0 standard pour le temps réel
- L'ITU est en cours de définition d'un profil UML 2.0 standard pour les télécommunications basé sur le SDL
- SDL est ou sera un profil UML 2.0 standard pour les télécommunications

SDL Forum Paris Septembre 2007

- System Design Languages
- Tutorial
- Expériences industrielles
- Recherche sur la modélisation
- <http://www.pragmadev.com/sdl07>

