Vérification des EFFBDs :
Model-checking en Ingénierie Système

Charlotte Seidner

mardi 3 novembre 2009

Directeur de thèse:	Olivier (H.) Roux	Université de Nantes (IRCCyN)
Rapporteurs:	Fabrice Kordon	Université P. & M. Curie (LIP6)
	François Vernadat	INSA Toulouse (LAAS)
Examinateurs:	Charles André	Université Nice-Sophia Antipolis (I3S)
	Jean-Philippe Lerat	Sodius
	Éric Niel	INSA Lyon (Ampère)
Invité:	Jean-Luc Wippler	C-S

Préambule

- thèse sous financement CIFRE :
  - équipe Systèmes Temps-Réel de l'IRCCyN (Nantes)
  - Sodius, jeune PME en Ingénierie Système (Nantes, Paris)
- Travaux débutés en octobre 2006
- Fil directeur : intégrer un outil de vérification formelle à une plate-forme de conception déployée industriellement

Préambule

- thèse sous financement CIFRE :
  - équipe Systèmes Temps-Réel de l'IRCCyN (Nantes)
  - Sodius, jeune PME en Ingénierie Système (Nantes, Paris)
- Travaux débutés en octobre 2006
- Fil directeur : intégrer un outil de vérification formelle à une plate-forme de conception déployée industriellement

Préambule

- thèse sous financement CIFRE :
  - équipe Systèmes Temps-Réel de l'IRCCyN (Nantes)
  - Sodius, jeune PME en Ingénierie Système (Nantes, Paris)
- Travaux débutés en octobre 2006
- Fil directeur : intégrer un outil de vérification formelle à une plate-forme de conception déployée industriellement

Préambule

- thèse sous financement CIFRE :
  - équipe Systèmes Temps-Réel de l'IRCCyN (Nantes)
  - Sodius, jeune PME en Ingénierie Système (Nantes, Paris)
- Travaux débutés en octobre 2006
- Fil directeur : intégrer un outil de vérification formelle à une plate-forme de conception déployée industriellement

Préambule

- thèse sous financement CIFRE :
  - équipe Systèmes Temps-Réel de l'IRCCyN (Nantes)
  - Sodius, jeune PME en Ingénierie Système (Nantes, Paris)
- Travaux débutés en octobre 2006
- Fil directeur : intégrer un outil de vérification formelle à une plate-forme de conception déployée industriellement

Plan de la soutenance

Plan de la soutenance

Contexte (1/3) : l'Ingénierie Système (IS)

- Développement de grands projets d'ingénierie complexes (NASA, années 1960)
- Besoins en outils, méthodes et processus garantissant leur maîtrise sur le cycle de vie
- Nombreux domaines d'application :
  - aérospatiale ;
  - défense ;
  - systèmes d'information ;
  - télécommunications...

Contexte (1/3) : l'Ingénierie Système (IS)

- Développement de grands projets d'ingénierie complexes (NASA, années 1960)
- Besoins en outils, méthodes et processus garantissant leur maîtrise sur le cycle de vie
- Nombreux domaines d'application :
  - aérospatiale ;
  - défense ;
  - systèmes d'information ;
  - télécommunications...

Contexte (1/3) : l'Ingénierie Système (IS)

- Développement de grands projets d'ingénierie complexes (NASA, années 1960)
- Besoins en outils, méthodes et processus garantissant leur maîtrise sur le cycle de vie
- Nombreux domaines d'application :
  - aérospatiale ;
  - défense ;
  - systèmes d'information ;
  - télécommunications. . .

Contexte (2/3) : quelques définitions

Système [Meinadier 1998]

Un système est un ensemble composite de personnels, de matériels et de logiciels organisés pour que leur interfonctionnement permette, dans un environnement donné, de remplir les missions pour lesquelles il a été conçu.

→ un système est complexe, hétérogène, dynamique

Ingénierie Système [Meinadier 1998]

L'Ingénierie Système est une démarche méthodologique proposant une approche globale et pluridisciplinaire de conception et de mise en œuvre des systèmes complexes.

Contexte (2/3) : quelques définitions

Système [Meinadier 1998]

Un système est un ensemble composite de personnels, de matériels et de logiciels organisés pour que leur interfonctionnement permette, dans un environnement donné, de remplir les missions pour lesquelles il a été conçu.

→ un système est complexe, hétérogène, dynamique

Ingénierie Système [Meinadier 1998]

L'Ingénierie Système est une démarche méthodologique proposant une approche globale et pluridisciplinaire de conception et de mise en œuvre des systèmes complexes.

Contexte (2/3) : quelques définitions

Système [Meinadier 1998]

Un système est un ensemble composite de personnels, de matériels et de logiciels organisés pour que leur interfonctionnement permette, dans un environnement donné, de remplir les missions pour lesquelles il a été conçu.

→ un système est complexe, hétérogène, dynamique

Ingénierie Système [Meinadier 1998]

L'Ingénierie Système est une démarche méthodologique proposant une approche globale et pluridisciplinaire de conception et de mise en œuvre des systèmes complexes.

Contexte (3/3) : la vérification des systèmes

- Processus normalisé : MIL-STD-499 (1969), IEEE 1220 (1995), ISO 15288 (2003)...
- Vérification : « A-t-on construit un bon système ? »
- Limite des techniques traditionnelles de simulation (non exhaustivité)
- Recours à des méthodes formelles dont le model-checking :

Principe du model-checking

1. *Étant donnés un modèle formel Σ du système S...*
2. *... et un modèle formel Φ de la propriété P...*
3. *... déterminer si Σ vérifie Φ : $\Sigma \models \Phi$*

Contexte (3/3) : la vérification des systèmes

- Processus normalisé : MIL-STD-499 (1969), IEEE 1220 (1995), ISO 15288 (2003)...
- Vérification : « A-t-on construit un bon système ? »
- Limite des techniques traditionnelles de simulation (non exhaustivité)
- Recours à des méthodes formelles dont le model-checking :

Principe du model-checking

1. *Étant donnés un modèle formel Σ du système S...*
2. *... et un modèle formel Φ de la propriété P...*
3. *... déterminer si Σ vérifie Φ : $\Sigma \models \Phi$*

Contexte (3/3) : la vérification des systèmes

- Processus normalisé : MIL-STD-499 (1969), IEEE 1220 (1995), ISO 15288 (2003)...
- Vérification : « A-t-on construit un bon système ? »
- Limite des techniques traditionnelles de simulation (non exhaustivité)
- Recours à des méthodes formelles dont le model-checking :

Principe du model-checking

1. *Étant donnés un modèle formel Σ du système S...*
2. *... et un modèle formel Φ de la propriété P...*
3. *... déterminer si Σ vérifie Φ : $\Sigma \models \Phi$*

Contexte (3/3) : la vérification des systèmes

- Processus normalisé : MIL-STD-499 (1969), IEEE 1220 (1995), ISO 15288 (2003)...
- Vérification : « A-t-on construit un bon système ? »
- Limite des techniques traditionnelles de simulation (non exhaustivité)
- Recours à des méthodes formelles dont le model-checking :

Principe du model-checking

1. *Étant donnés un modèle formel Σ du système S...*
2. *... et un modèle formel Φ de la propriété P...*
3. *... déterminer si Σ vérifie Φ : $\Sigma \models \Phi$*

Contexte (3/3) : la vérification des systèmes

- Processus normalisé : MIL-STD-499 (1969), IEEE 1220 (1995), ISO 15288 (2003)...
- Vérification : « A-t-on construit un bon système ? »
- Limite des techniques traditionnelles de simulation (non exhaustivité)
- Recours à des méthodes formelles dont le model-checking :

Principe du model-checking

1. *Étant donnés un modèle formel Σ du système S...*
2. *... et un modèle formel Φ de la propriété P...*
3. *... déterminer si Σ vérifie Φ :* $\Sigma \models \Phi$

Problématiques de la thèse

Problématique principale

Concevoir un outil de vérification formelle d'architectures fonctionnelles utilisable en conception système

Participation de Sodius au projet Kimono (DGA)

Problématique dérivée

Concevoir un outil de vérification formelle d'architectures dysfonctionnelles utilisable en conception système (ajout de pannes)

Problématiques de la thèse

Problématique principale

Concevoir un outil de vérification formelle d'architectures fonctionnelles utilisable en conception système

Participation de Sodius au projet Kimono (DGA)

Problématique dérivée

Concevoir un outil de vérification formelle d'architectures dysfonctionnelles utilisable en conception système (ajout de pannes)

Problématiques de la thèse

Problématique principale

Concevoir un outil de vérification formelle d'architectures fonctionnelles utilisable en conception système

Participation de Sodius au projet Omotesc (DGA)

Problématique dérivée

Concevoir un outil de vérification formelle d'architectures dysfonctionnelles utilisable en conception système (ajout de pannes)

Problématiques de la thèse

Problématique principale

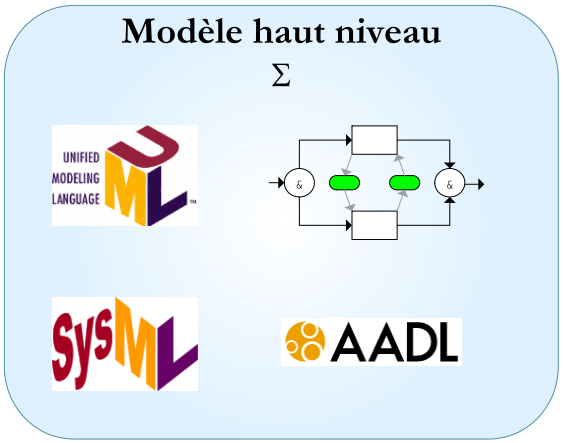
Concevoir un outil de vérification formelle d'architectures fonctionnelles utilisable en conception système

Participation de Sodius au projet Omotesc (DGA)

Problématique dérivée

Concevoir un outil de vérification formelle d'architectures dysfonctionnelles utilisable en conception système (ajout de pannes)

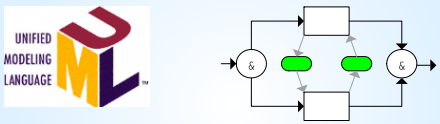
Démarche générale



Démarche générale

Modèle haut niveau

Σ



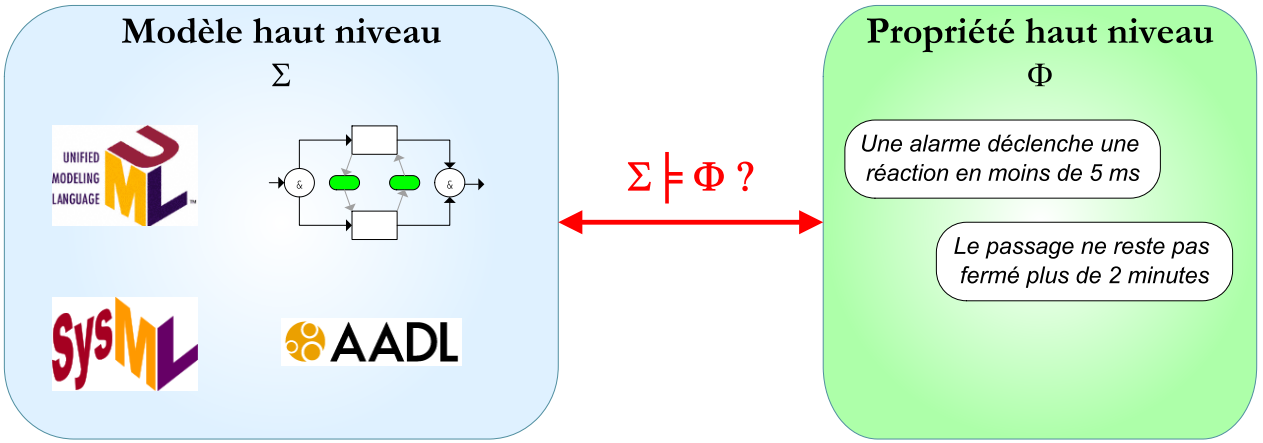
Propriété haut niveau

Φ

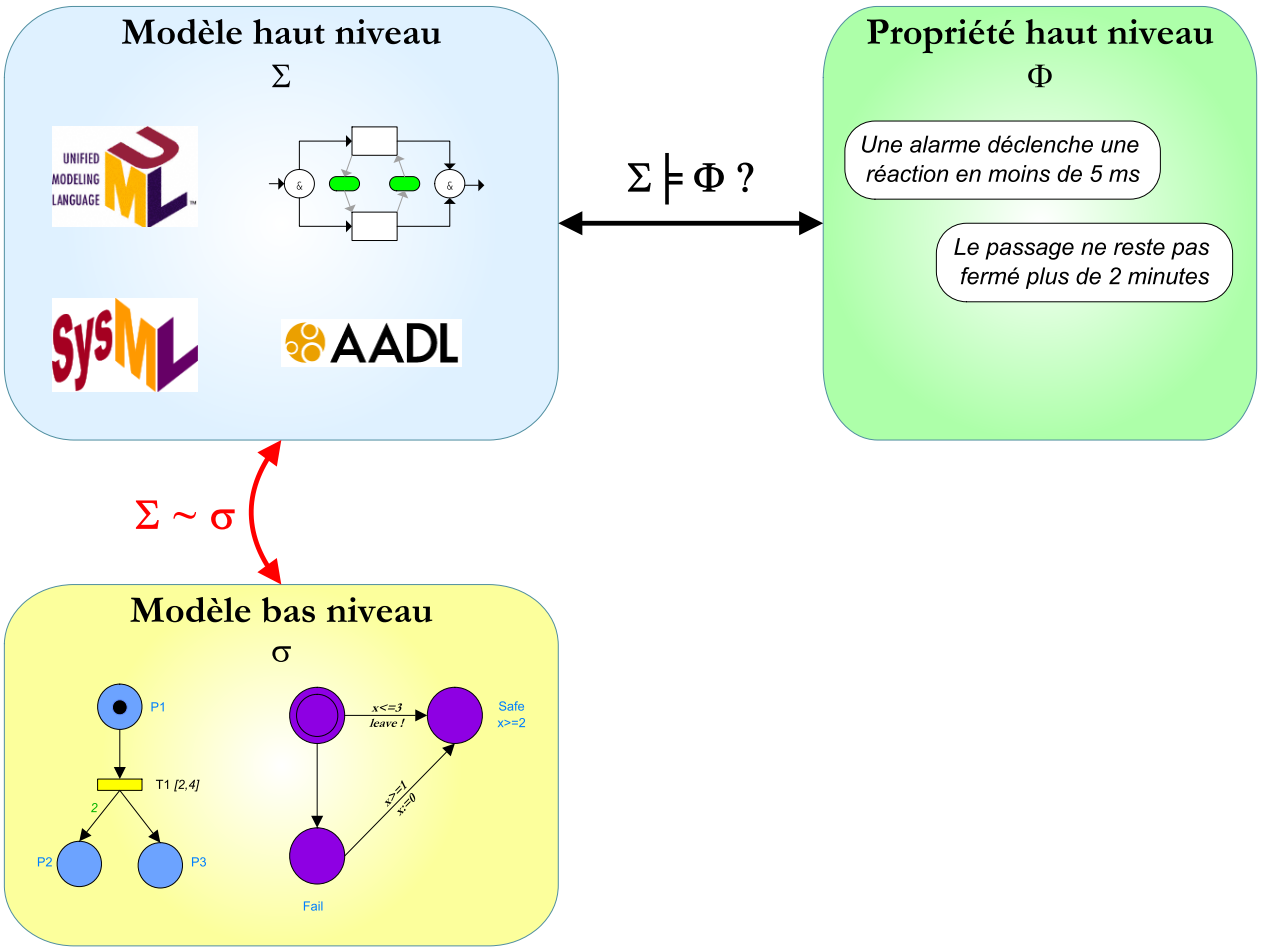
Une alarme déclenche une réaction en moins de 5 ms

Le passage ne reste pas fermé plus de 2 minutes

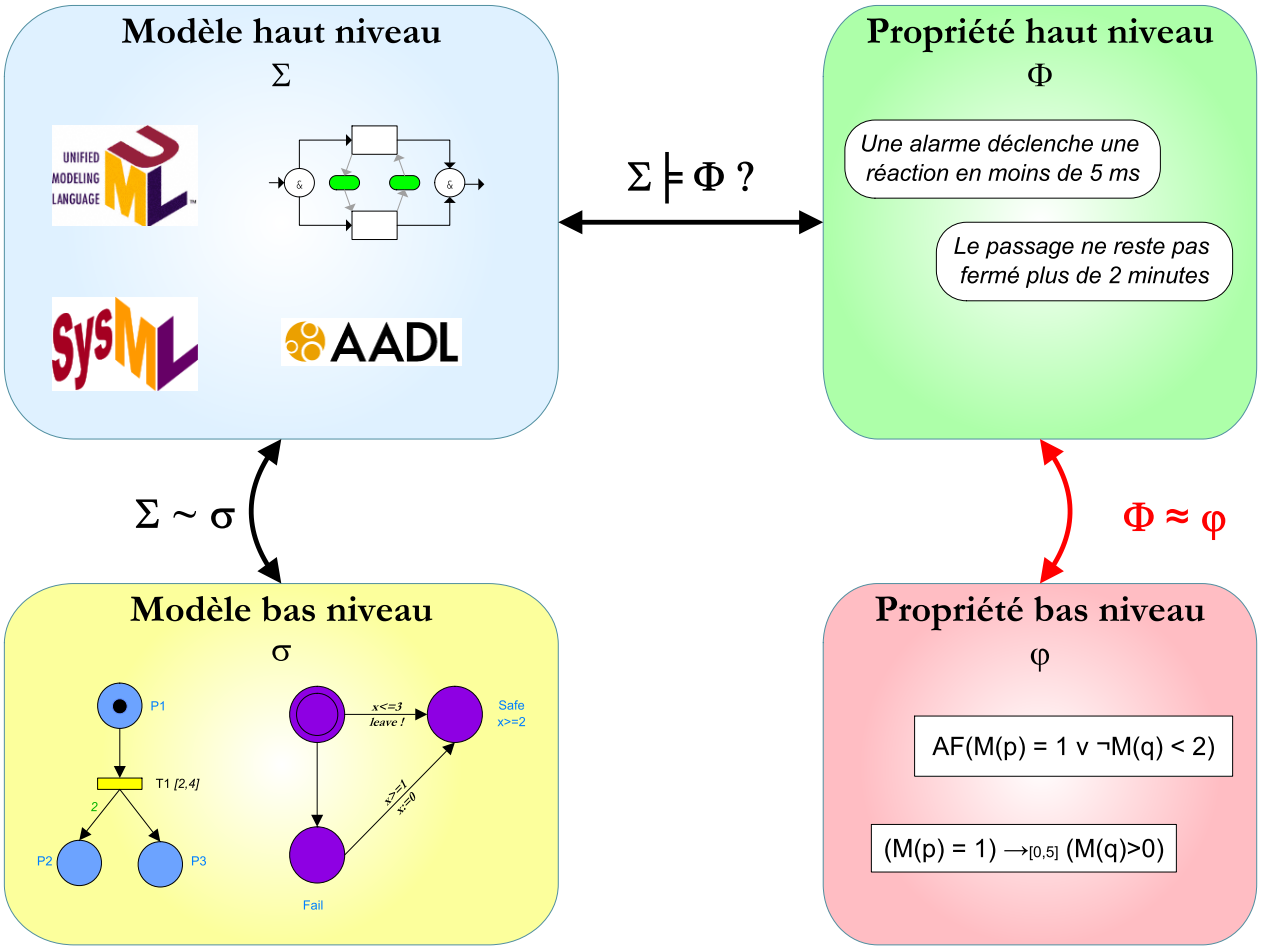
Démarche générale



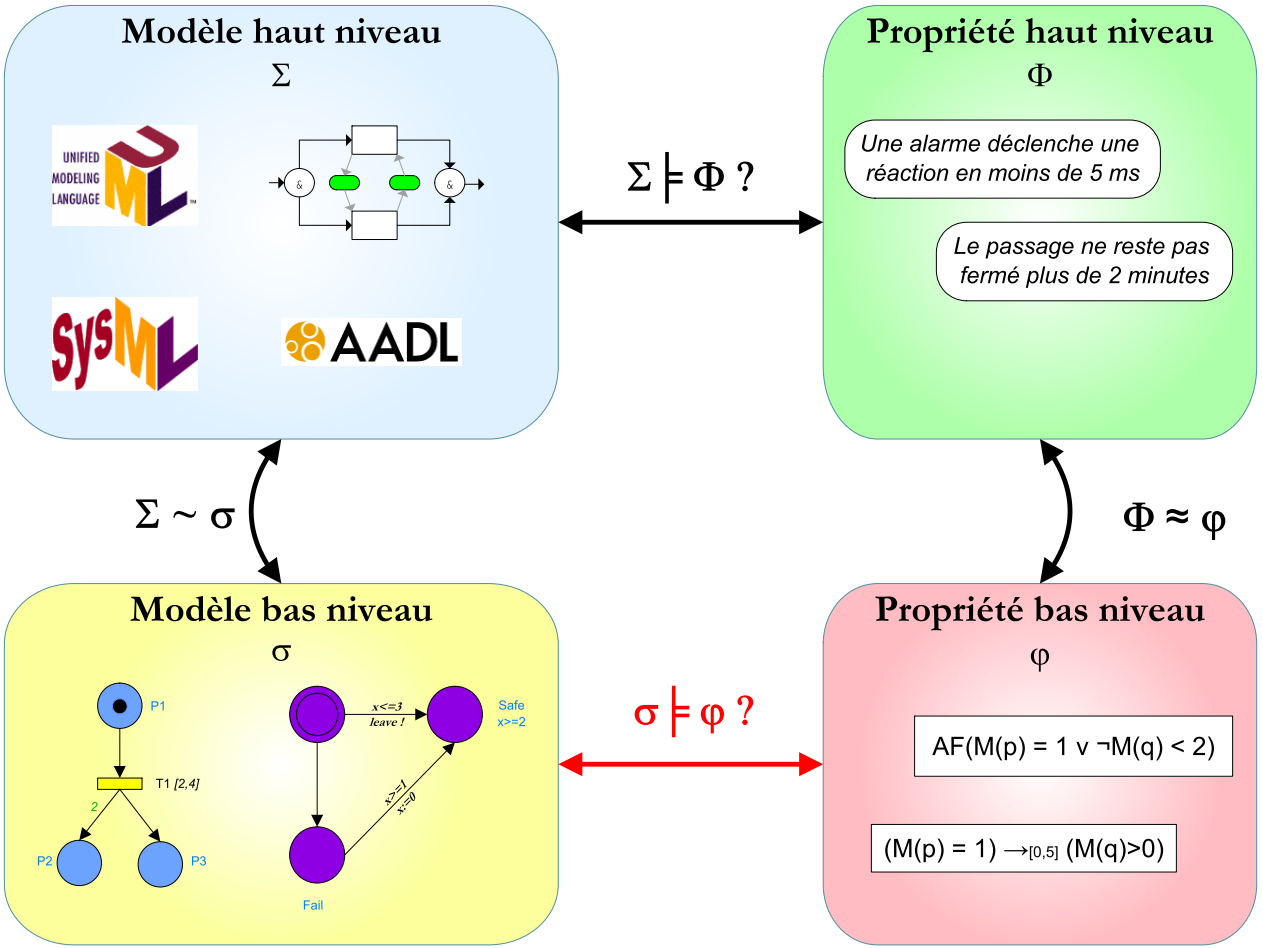
Démarche générale



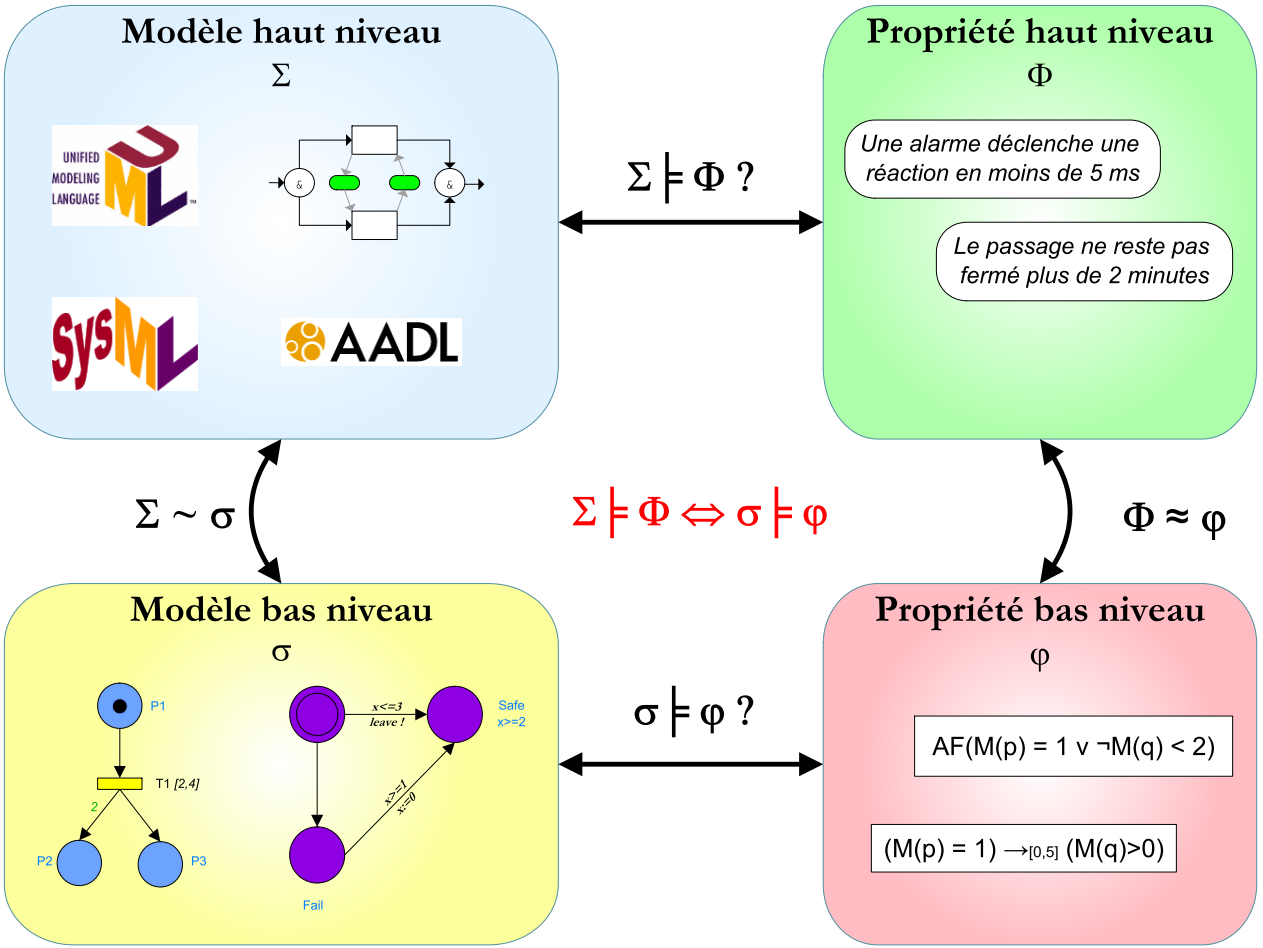
Démarche générale



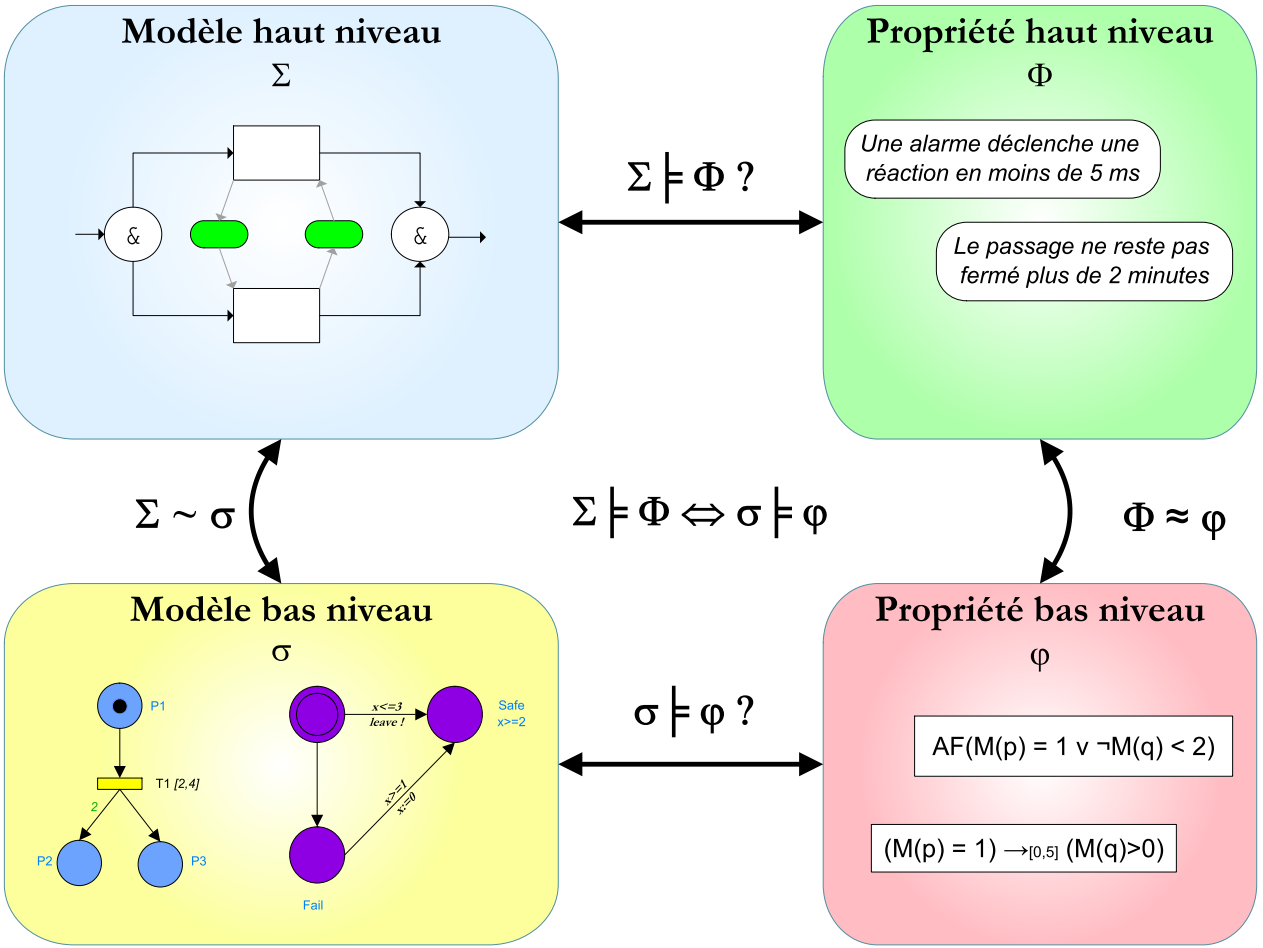
Démarche générale



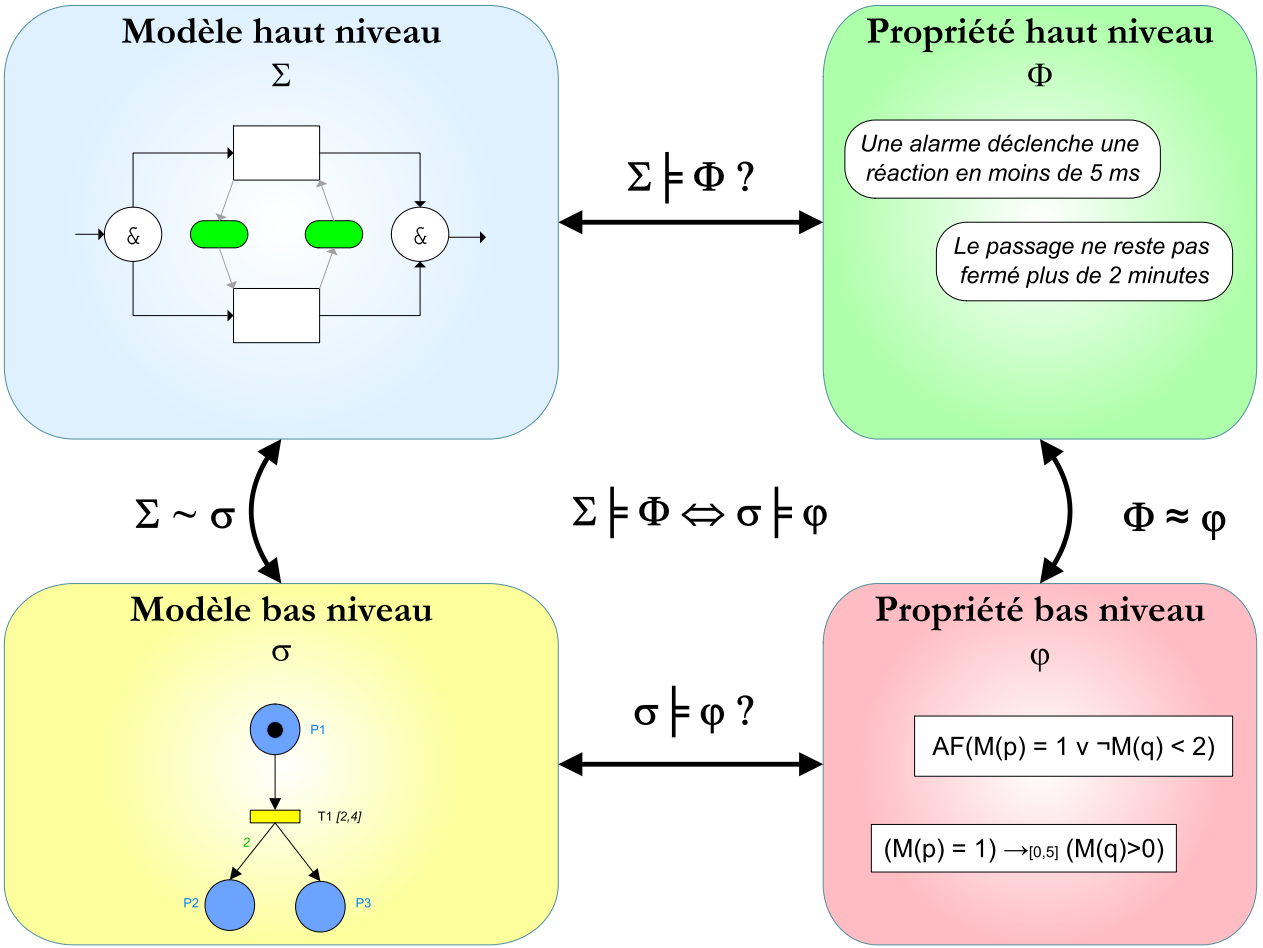
Démarche générale



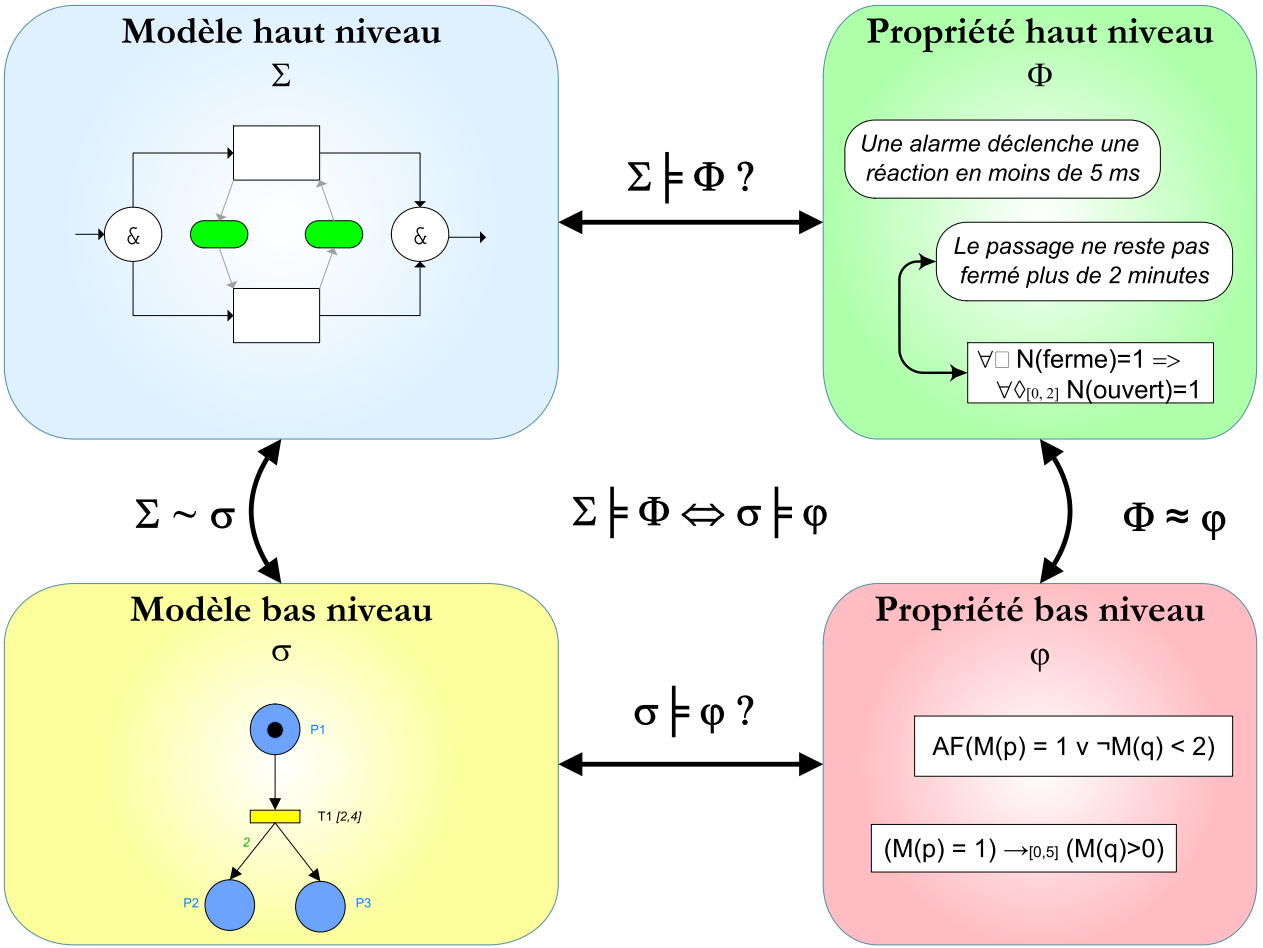
Démarche générale



Démarche générale



Démarche générale



Plan de la soutenance

Les diagrammes EFFBDs : généralités

- Enhanced Function Flow Block Diagram [Long 1995]
- Langage essentiellement graphique, non formalisé mais intuitif
- Représentation du comportement dynamique et fonctionnel de systèmes
  - complexes ;
  - distribués ;
  - hiérarchiques ;
  - concurrents ;
  - communicants. . .

Les diagrammes EFFBDs : généralités

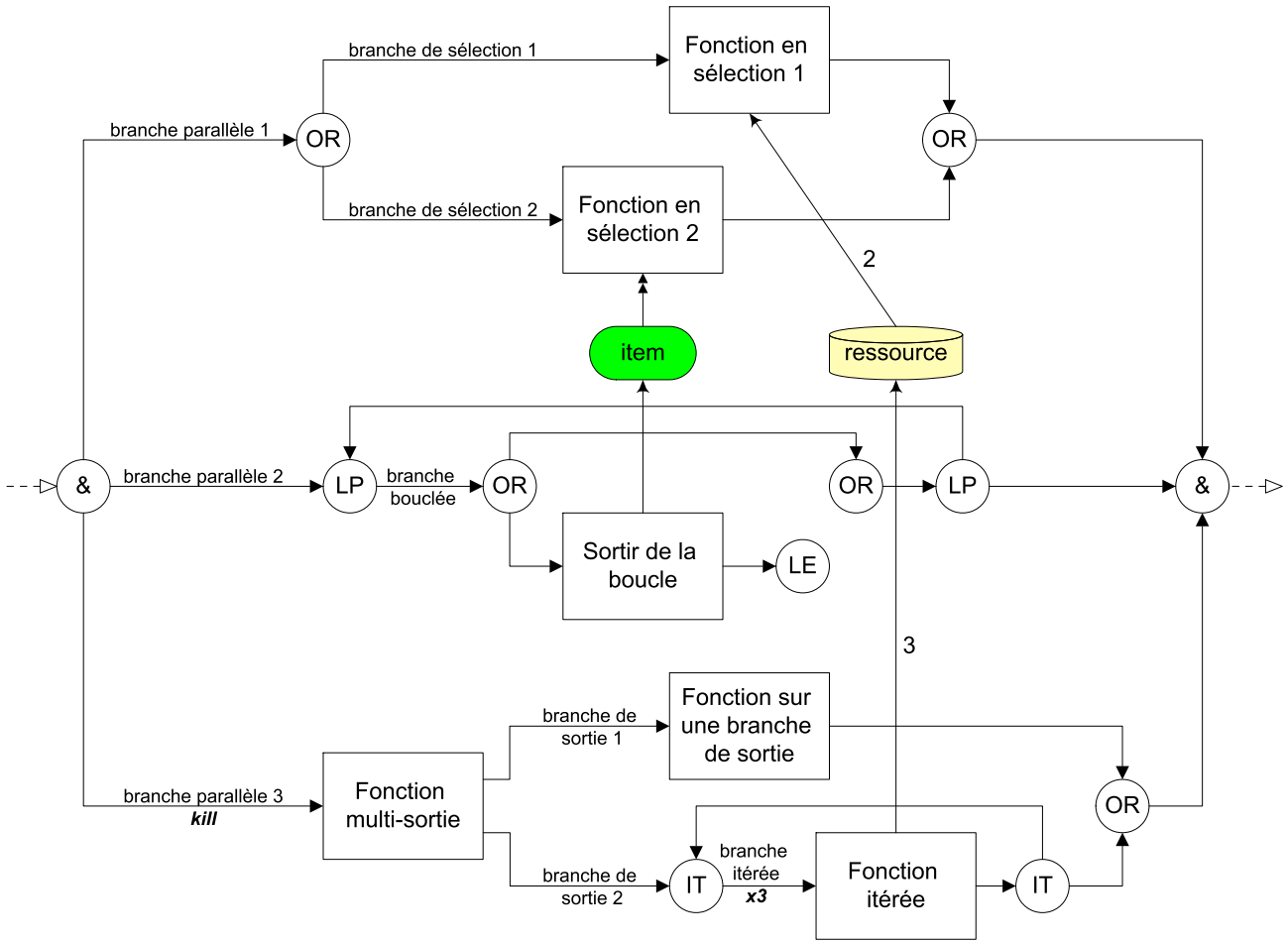
- Enhanced Function Flow Block Diagram [Long 1995]
- Langage essentiellement graphique, non formalisé mais intuitif
- Représentation du comportement dynamique et fonctionnel de systèmes
  - complexes ;
  - distribués ;
  - hiérarchiques ;
  - concurrents ;
  - communicants. . .

Les diagrammes EFFBDs : généralités

- Enhanced Function Flow Block Diagram [Long 1995]
- Langage essentiellement graphique, non formalisé mais intuitif
- Représentation du comportement dynamique et fonctionnel de systèmes
  - complexes ;
  - distribués ;
  - hiérarchiques ;
  - concurrents ;
  - communicants. . .

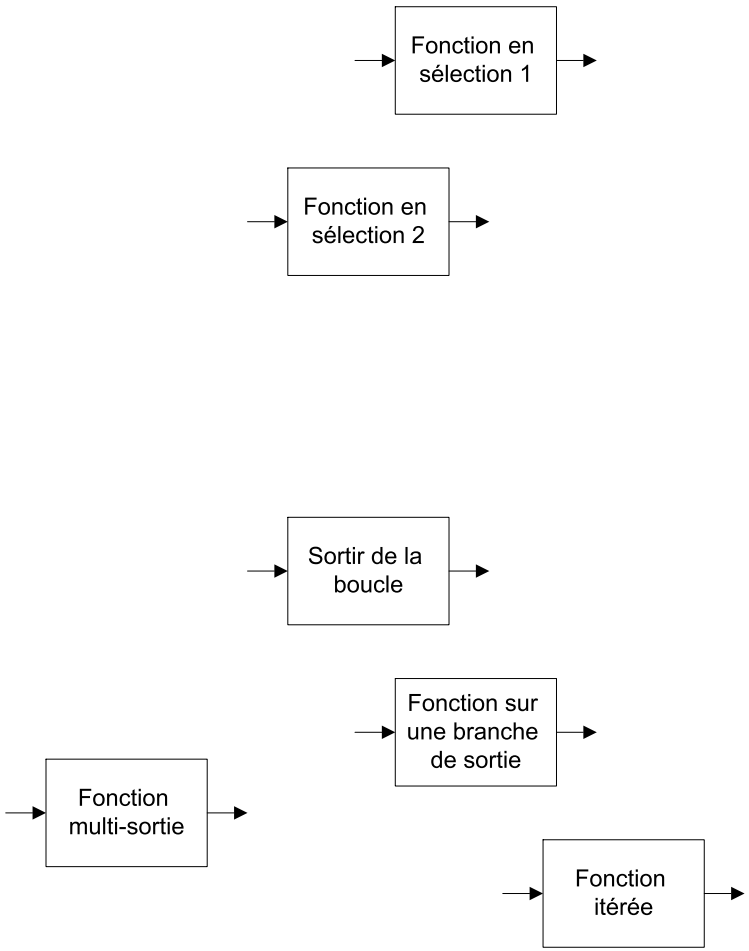
Aperçu d'un EFFBD (d'après [Long, 1995])

Ensemble de fonctions, d'items et de ressources, de structures de contrôle imbriquées



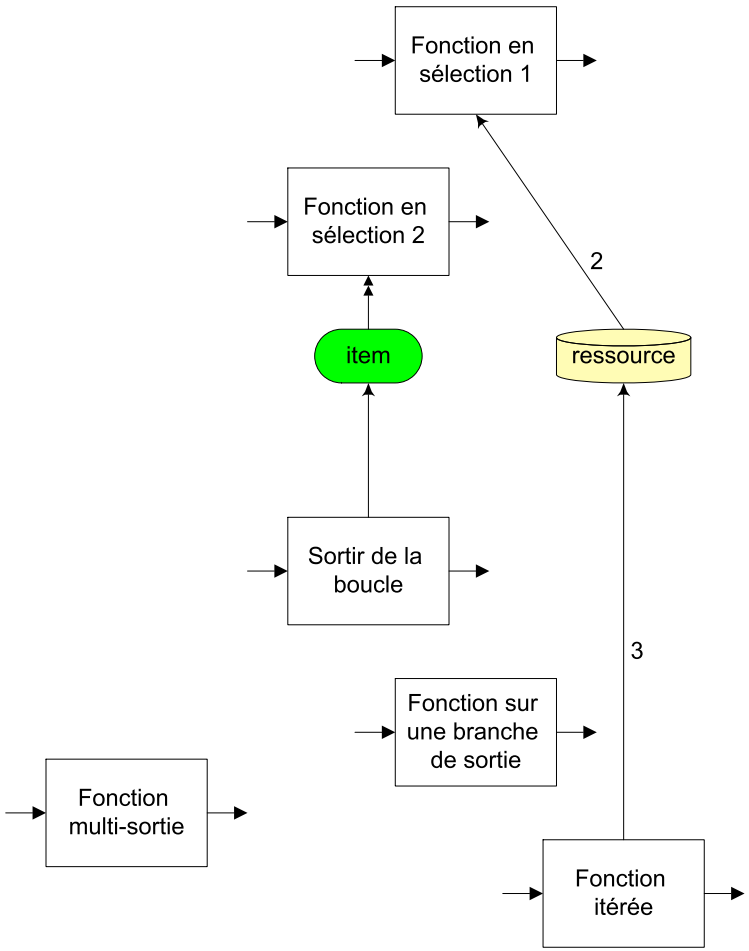
Aperçu d'un EFFBD (d'après [Long, 1995])

Ensemble de fonctions, d'items et de ressources, de structures de contrôle imbriquées



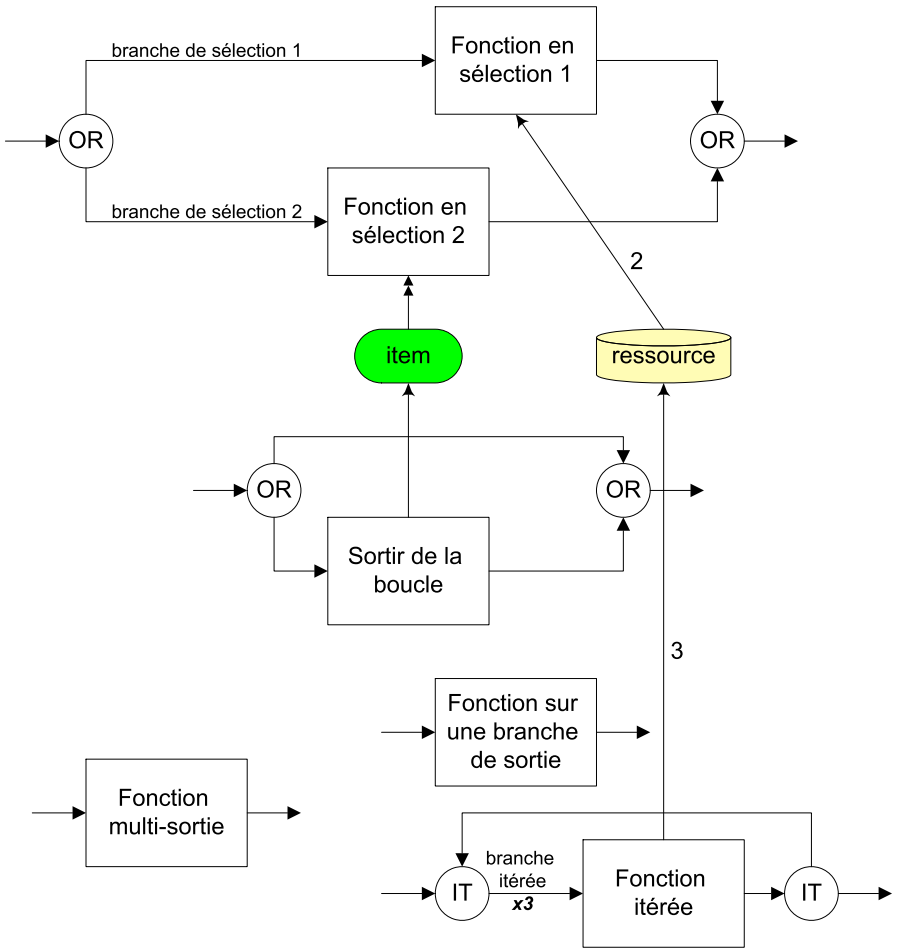
Aperçu d'un EFFBD (d'après [Long, 1995])

Ensemble de fonctions, d'items et de ressources, de structures de contrôle imbriquées



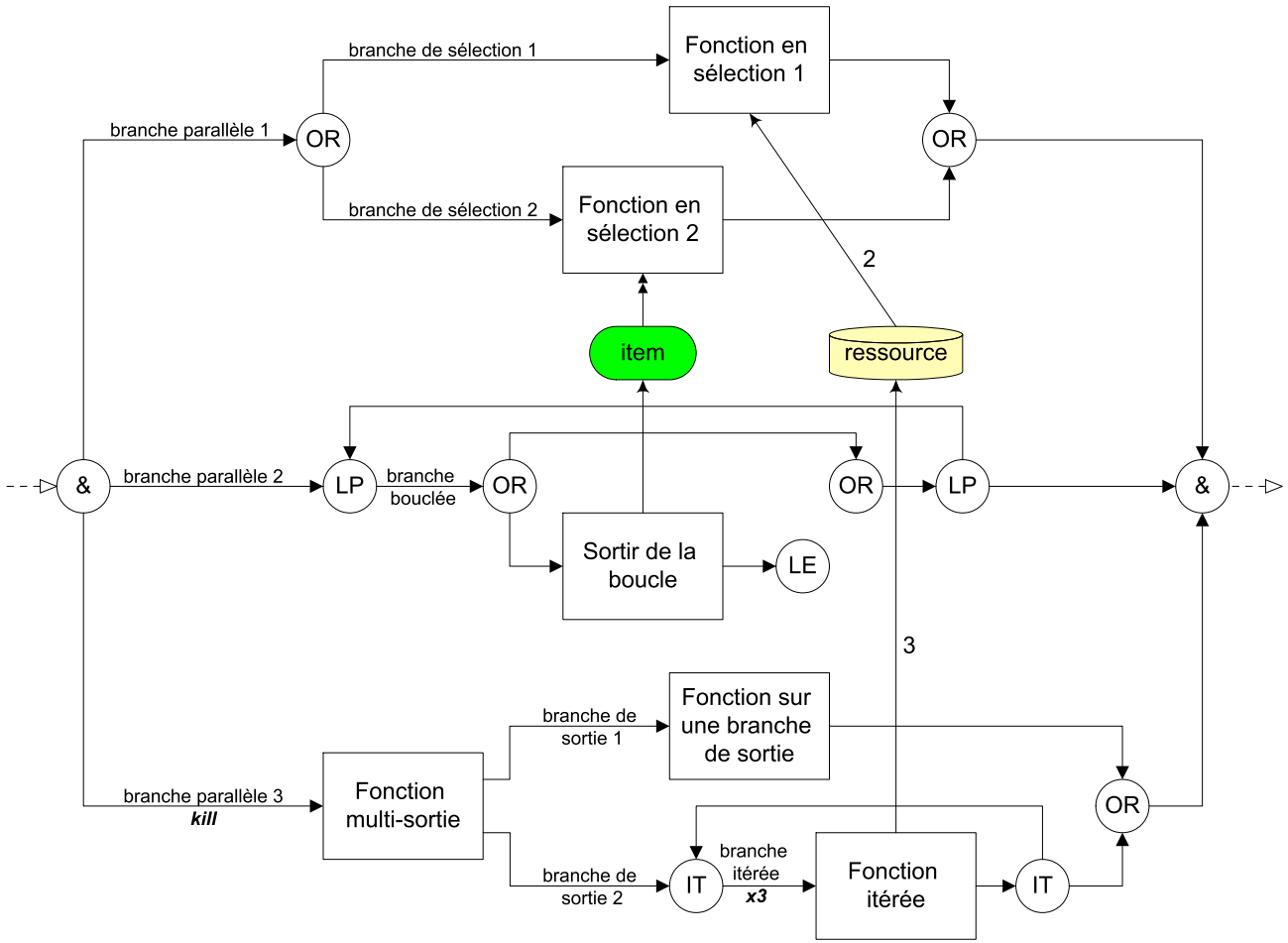
Aperçu d'un EFFBD (d'après [Long, 1995])

Ensemble de fonctions, d'items et de ressources, de structures de contrôle imbriquées



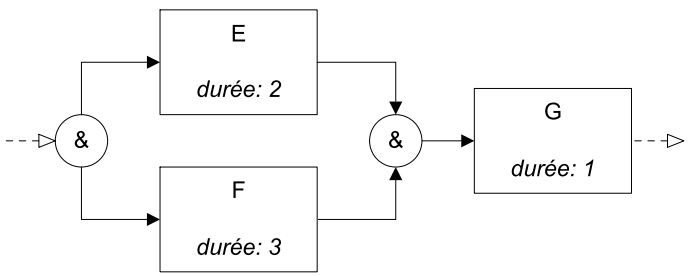
Aperçu d'un EFFBD (d'après [Long, 1995])

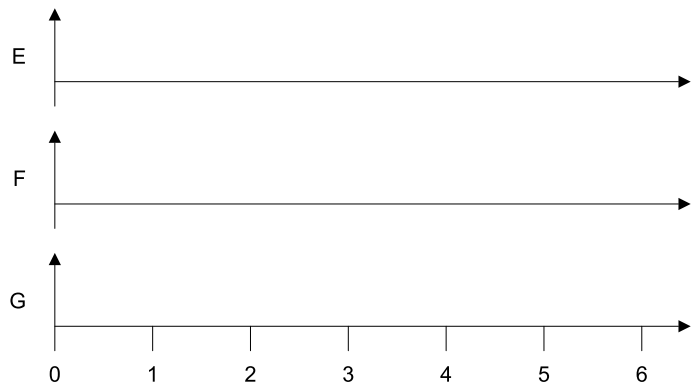
Ensemble de fonctions, d'items et de ressources, de structures de contrôle imbriquées



Sémantique informelle (1/2) : branches parallèles

Structure parallèle (AND)

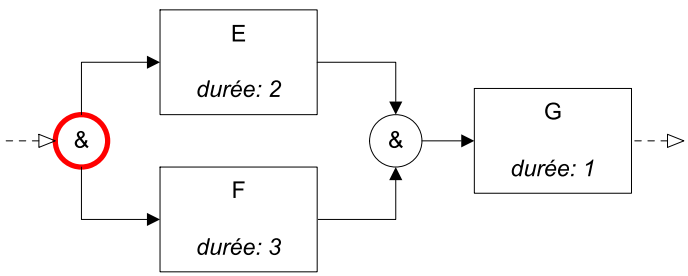


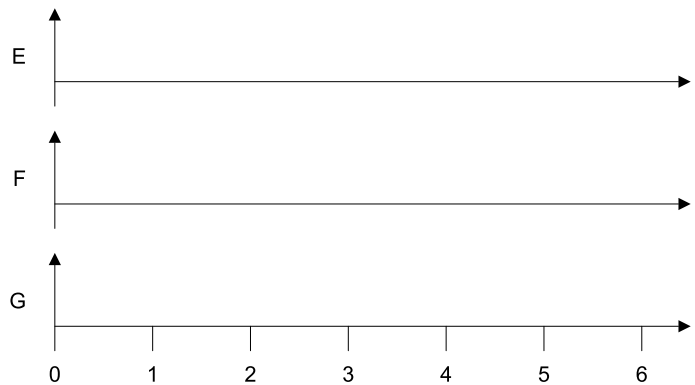


→ G attend la fin de E et F pour s'exécuter

Sémantique informelle (1/2) : branches parallèles

Structure parallèle (AND) : synchronisation initiale et finale

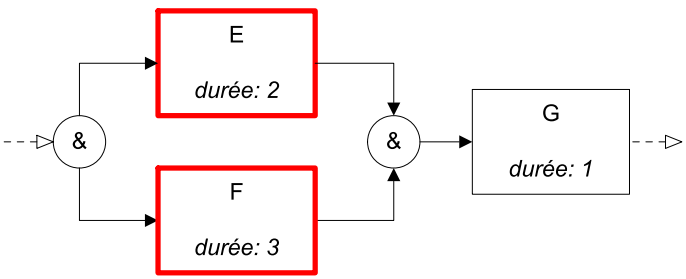


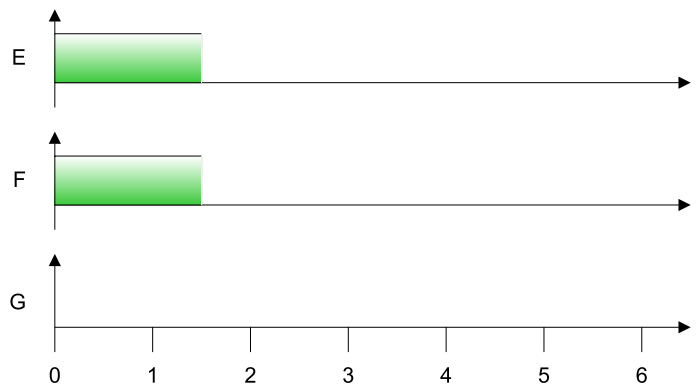


→ G attend la fin de E et F pour s'exécuter

Sémantique informelle (1/2) : branches parallèles

Structure parallèle (AND) : synchronisation initiale et finale

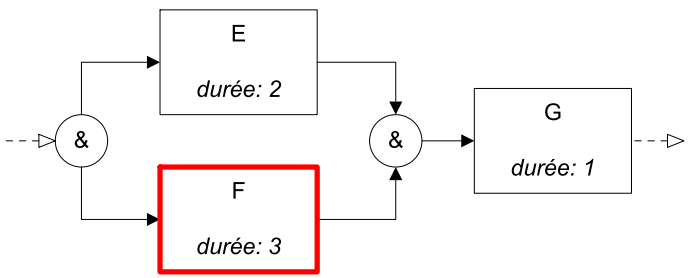


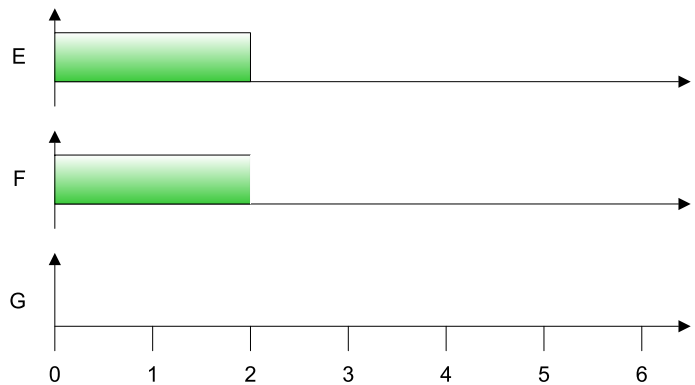


→ G attend la fin de E et F pour s'exécuter

Sémantique informelle (1/2) : branches parallèles

Structure parallèle (AND) : synchronisation initiale et finale

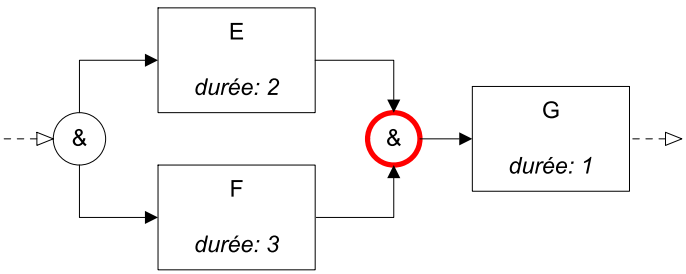


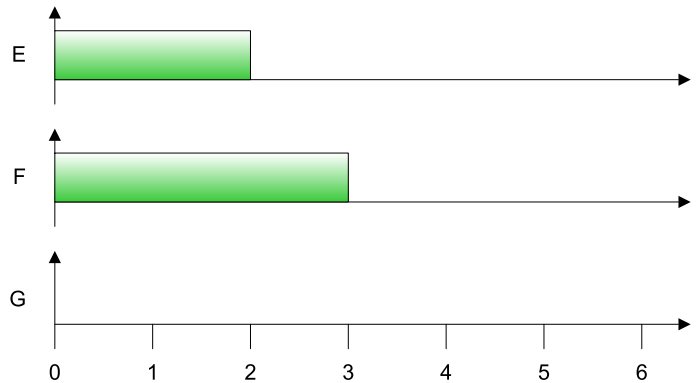


→ G attend la fin de E et F pour s'exécuter

Sémantique informelle (1/2) : branches parallèles

Structure parallèle (AND) : synchronisation initiale et finale

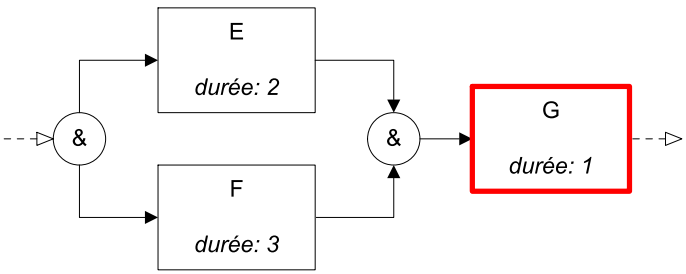


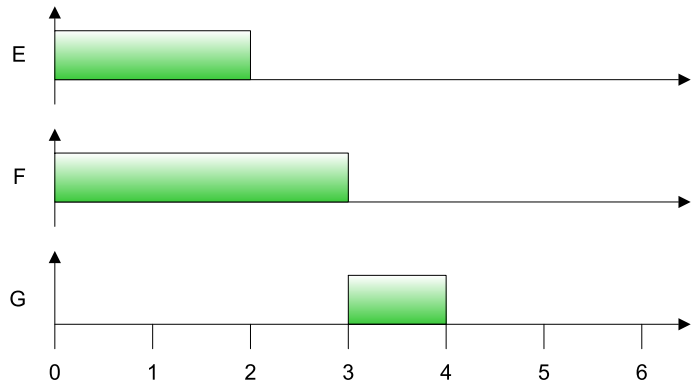


→ G attend la fin de E et F pour s'exécuter

Sémantique informelle (1/2) : branches parallèles

Structure parallèle (AND) : synchronisation initiale et finale

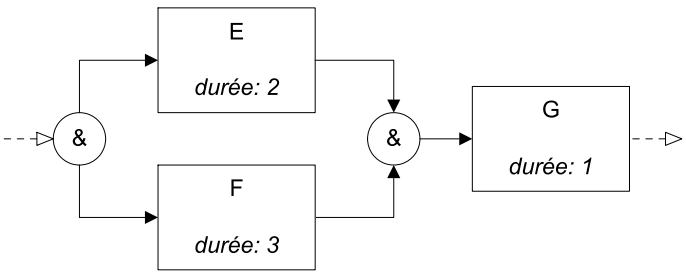


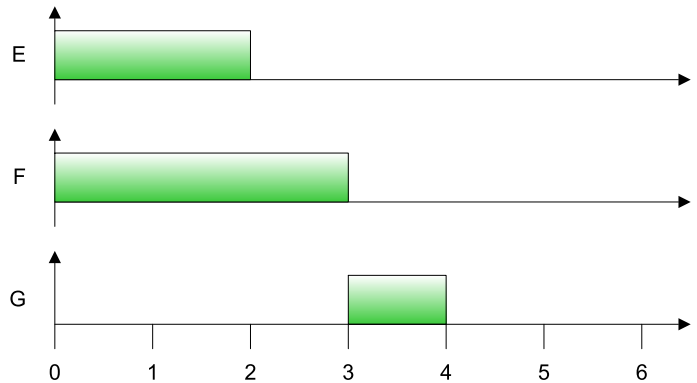


→ G attend la fin de E et F pour s'exécuter

Sémantique informelle (1/2) : branches parallèles

Structure parallèle (AND) : synchronisation initiale et finale

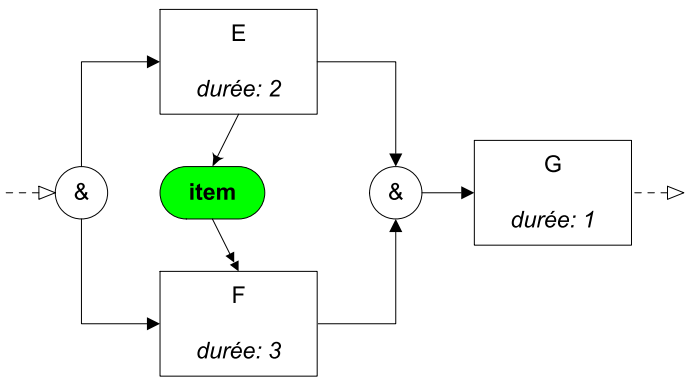


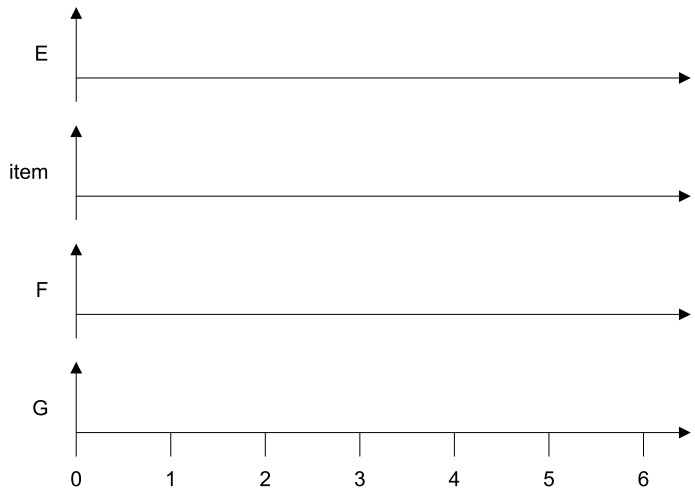


→ G attend la fin de E et F pour s'exécuter

Sémantique informelle (2/2) : synchronisation par flux

- Consommation des flux : avant l'exécution (rôle déclencheur)
- Production des flux : à l'issue de l'exécution

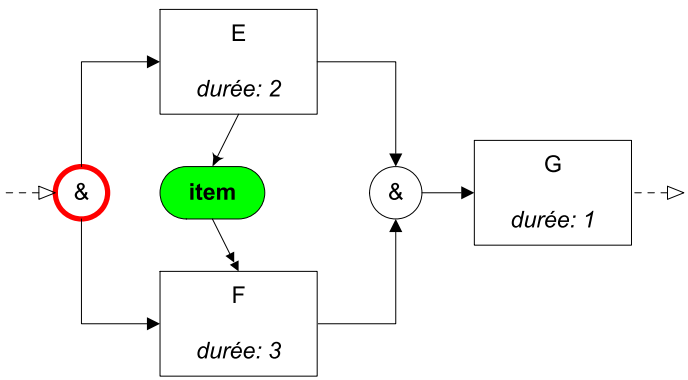


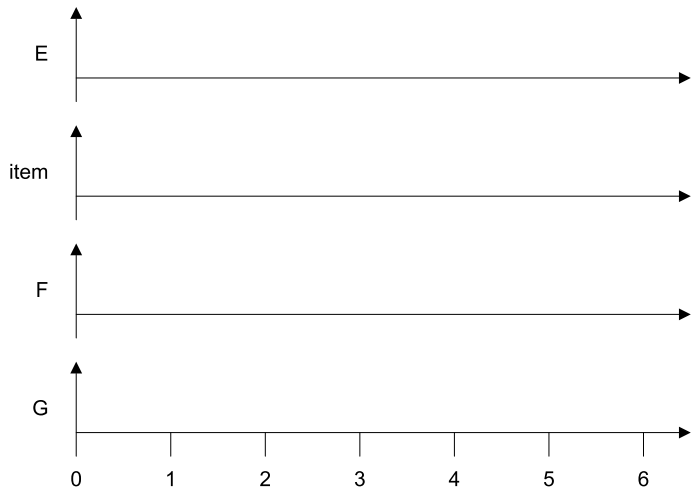


→ F doit attendre la fin de E pour s'exécuter

Sémantique informelle (2/2) : synchronisation par flux

- Consommation des flux : avant l'exécution (rôle déclencheur)
- Production des flux : à l'issue de l'exécution

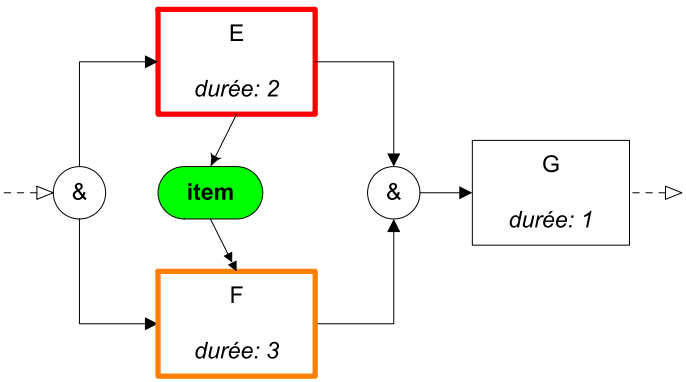


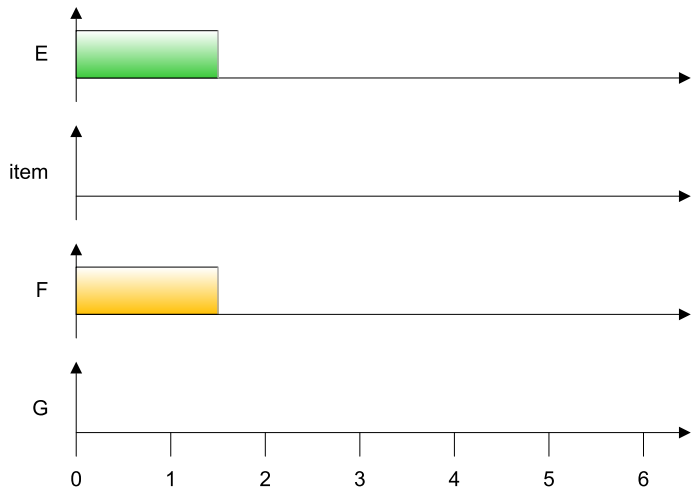


→ F doit attendre la fin de E pour s'exécuter

Sémantique informelle (2/2) : synchronisation par flux

- Consommation des flux : avant l'exécution (rôle déclencheur)
- Production des flux : à l'issue de l'exécution

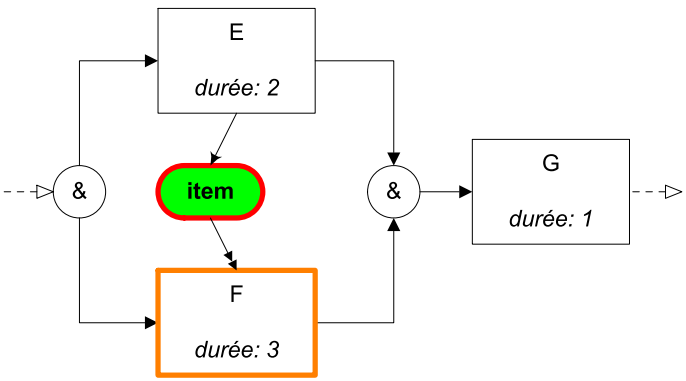


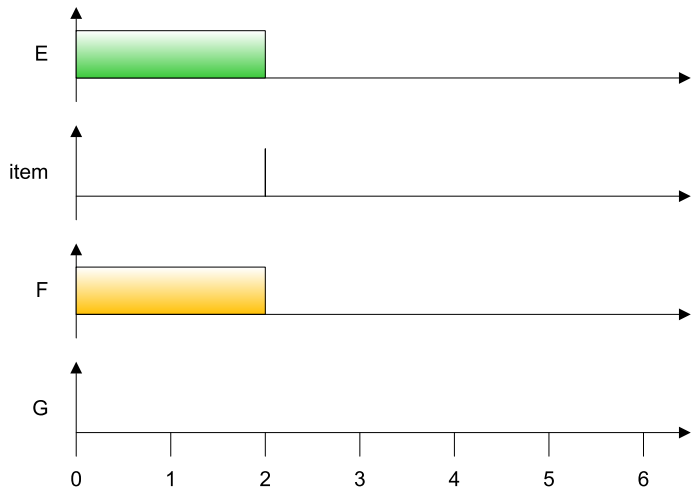


→ F doit attendre la fin de E pour s'exécuter

Sémantique informelle (2/2) : synchronisation par flux

- Consommation des flux : avant l'exécution (rôle déclencheur)
- Production des flux : à l'issue de l'exécution

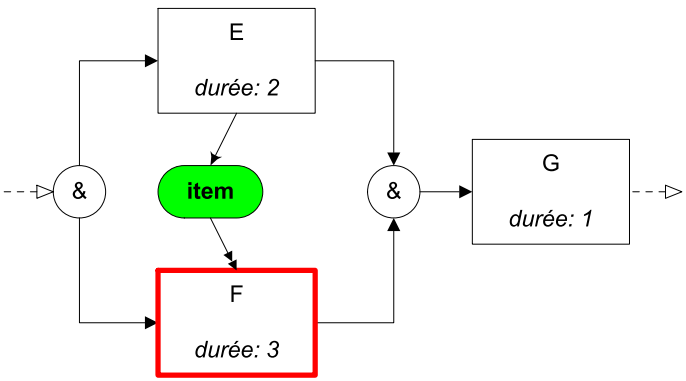


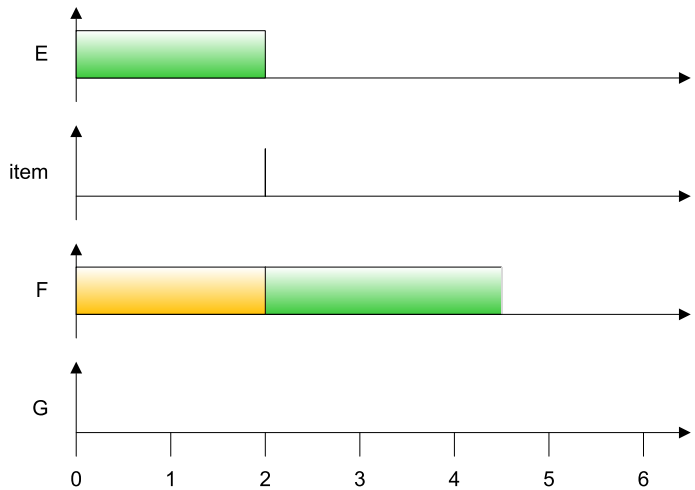


→ F doit attendre la fin de E pour s'exécuter

Sémantique informelle (2/2) : synchronisation par flux

- Consommation des flux : avant l'exécution (rôle déclencheur)
- Production des flux : à l'issue de l'exécution

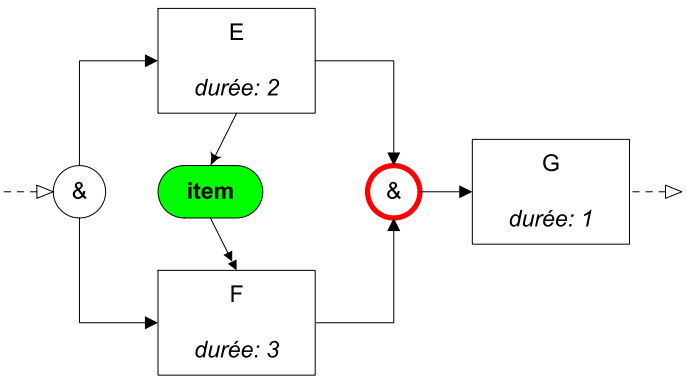


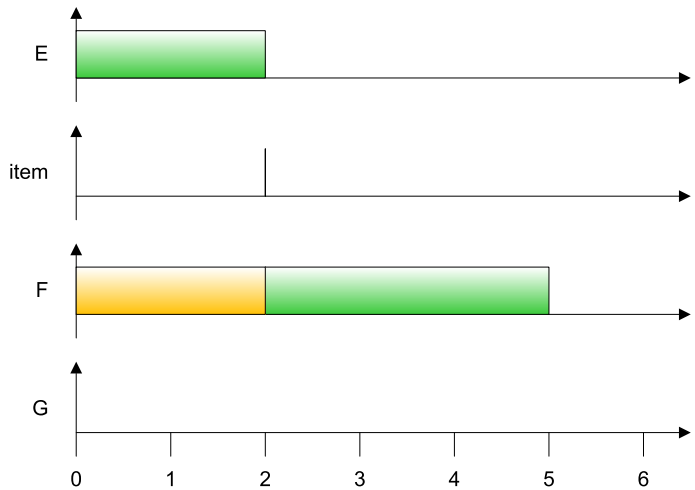


→ F doit attendre la fin de E pour s'exécuter

Sémantique informelle (2/2) : synchronisation par flux

- Consommation des flux : avant l'exécution (rôle déclencheur)
- Production des flux : à l'issue de l'exécution

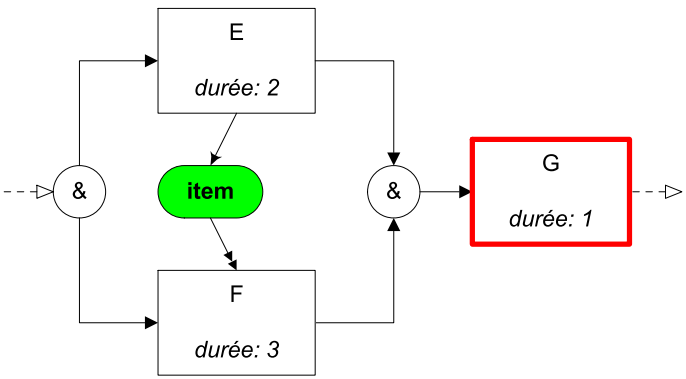


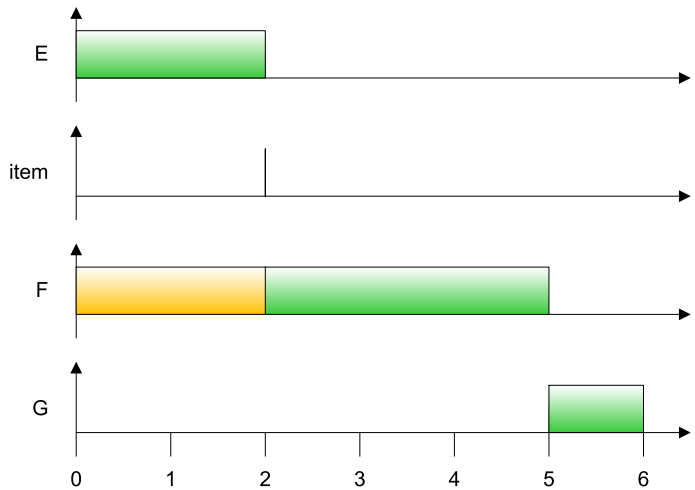


→ F doit attendre la fin de E pour s'exécuter

Sémantique informelle (2/2) : synchronisation par flux

- Consommation des flux : avant l'exécution (rôle déclencheur)
- Production des flux : à l'issue de l'exécution

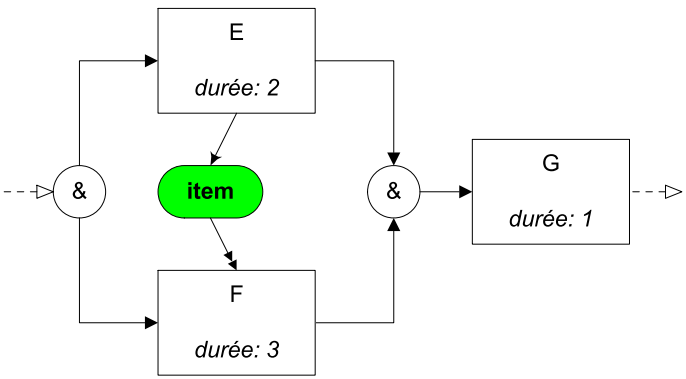


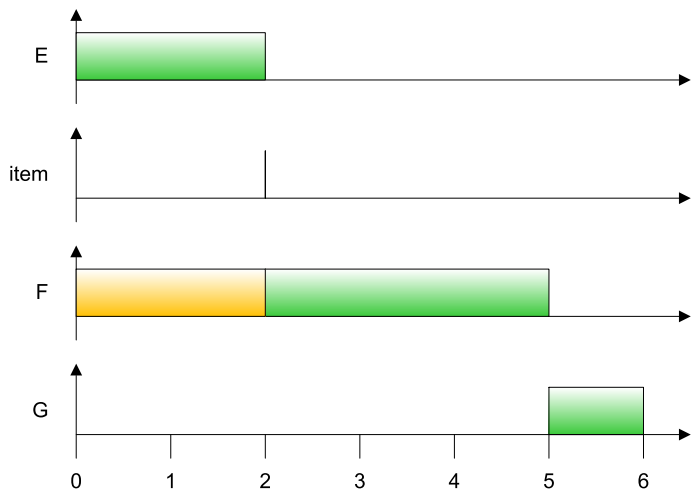


→ F doit attendre la fin de E pour s'exécuter

Sémantique informelle (2/2) : synchronisation par flux

- Consommation des flux : avant l'exécution (rôle déclencheur)
- Production des flux : à l'issue de l'exécution





→ F doit attendre la fin de E pour s'exécuter

Syntaxe des EFFBDs

- Un EFFBD est composé :
  - d'un ensemble $\mathcal{N}$ de nœuds ouvrants ou fermants *(fonctions et structures de contrôle)* reliés par des arcs ;
  - d'un ensemble $\mathcal{F}$ de flux *(items et ressources)* ayant un niveau initial et reliés aux fonctions par des arcs pondérés ;
  - de fonctions donnant les durées d'exécution minimales et maximales.
- Définition de contraintes syntaxiques : description formelle des EFFBDs bien formés :
  - imbrication des structures ;
  - unicité du nœud initial...

Syntaxe des EFFBDs

- Un EFFBD est composé :
  - d'un ensemble $\mathcal{N}$ de nœuds ouvrants ou fermants *(fonctions et structures de contrôle)* reliés par des arcs ;
  - d'un ensemble $\mathcal{F}$ de flux *(items et ressources)* ayant un niveau initial et reliés aux fonctions par des arcs pondérés ;
  - de fonctions donnant les durées d'exécution minimales et maximales.
- Définition de contraintes syntaxiques : description formelle des EFFBDs bien formés :
  - imbrication des structures ;
  - unicité du nœud initial...

Syntaxe des EFFBDs

- Un EFFBD est composé :
  - d'un ensemble $\mathcal{N}$ de nœuds ouvrants ou fermants *(fonctions et structures de contrôle)* reliés par des arcs ;
  - d'un ensemble $\mathcal{F}$ de flux *(items et ressources)* ayant un niveau initial et reliés aux fonctions par des arcs pondérés ;
  - de fonctions donnant les durées d'exécution minimales et maximales.
- Définition de contraintes syntaxiques : description formelle des EFFBDs bien formés :
  - imbrication des structures ;
  - unicité du nœud initial...

Syntaxe des EFFBDs

- Un EFFBD est composé :
  - d'un ensemble $\mathcal{N}$ de nœuds ouvrants ou fermants *(fonctions et structures de contrôle)* reliés par des arcs ;
  - d'un ensemble $\mathcal{F}$ de flux *(items et ressources)* ayant un niveau initial et reliés aux fonctions par des arcs pondérés ;
  - de fonctions donnant les durées d'exécution minimales et maximales.
- Définition de contraintes syntaxiques : description formelle des EFFBDs bien formés :
  - imbrication des structures ;
  - unicité du nœud initial...

Syntaxe des EFFBDs

- Un EFFBD est composé :
  - d'un ensemble $\mathcal{N}$ de nœuds ouvrants ou fermants *(fonctions et structures de contrôle)* reliés par des arcs ;
  - d'un ensemble $\mathcal{F}$ de flux *(items et ressources)* ayant un niveau initial et reliés aux fonctions par des arcs pondérés ;
  - de fonctions donnant les durées d'exécution minimales et maximales.
- Définition de contraintes syntaxiques : description formelle des EFFBDs bien formés :
  - imbrication des structures ;
  - unicité du nœud initial...

Sémantique des EFFBDs [Seidner et Roux, 2008]

- Inclusion du temps : description de la sémantique sous la forme d'un système de transitions temporisé (STT) $(S, s_0, \mathcal{N}, \rightarrow)$
- Un état $s = (Act, N, \nu) \in S$ représente :
  - l'activité de chaque nœud (inactif, en attente de flux,...) ;
  - le niveau de chaque flux ;
  - pour chaque fonction, le temps écoulé depuis le début de son exécution.
- La relation de transition $\rightarrow$ du STT décrit :
  - le tir d'un nœud (transmission du flux de contrôle) [transition discrète]
  - le passage du temps [transition continue]

Sémantique des EFFBDs [Seidner et Roux, 2008]

- Inclusion du temps : description de la sémantique sous la forme d'un système de transitions temporisé (STT) $(S, s_0, \mathcal{N}, \rightarrow)$
- Un état $s = (Act, N, \nu) \in S$ représente :
  - l'activité de chaque nœud (inactif, en attente de flux,...) ;
  - le niveau de chaque flux ;
  - pour chaque fonction, le temps écoulé depuis le début de son exécution.
- La relation de transition $\rightarrow$ du STT décrit :
  - le tir d'un nœud (transmission du flux de contrôle) [transition discrète]
  - le passage du temps [transition continue]

Sémantique des EFFBDs [Seidner et Roux, 2008]

- Inclusion du temps : description de la sémantique sous la forme d'un système de transitions temporisé (STT) $(S, s_0, \mathcal{N}, \rightarrow)$
- Un état $s = (Act, N, \nu) \in S$ représente :
  - l'activité de chaque nœud (inactif, en attente de flux,...) ;
  - le niveau de chaque flux ;
  - pour chaque fonction, le temps écoulé depuis le début de son exécution.
- La relation de transition $\rightarrow$ du STT décrit :
  - le tir d'un nœud (transmission du flux de contrôle) [transition discrète]
  - le passage du temps [transition continue]

Sémantique des EFFBDs [Seidner et Roux, 2008]

- Inclusion du temps : description de la sémantique sous la forme d'un système de transitions temporisé (STT) $(S, s_0, \mathcal{N}, \rightarrow)$
- Un état $s = (Act, N, \nu) \in S$ représente :
  - l'activité de chaque nœud (inactif, en attente de flux,...) ;
  - le niveau de chaque flux ;
  - pour chaque fonction, le temps écoulé depuis le début de son exécution.
- La relation de transition $\rightarrow$ du STT décrit :
  - le tir d'un nœud (transmission du flux de contrôle) [transition discrète]
  - le passage du temps [transition continue]

Sémantique des EFFBDs [Seidner et Roux, 2008]

- Inclusion du temps : description de la sémantique sous la forme d'un système de transitions temporisé (STT) $(S, s_0, \mathcal{N}, \rightarrow)$
- Un état $s = (Act, N, \nu) \in S$ représente :
  - l'activité de chaque nœud (inactif, en attente de flux,...) ;
  - le niveau de chaque flux ;
  - pour chaque fonction, le temps écoulé depuis le début de son exécution.
- La relation de transition $\rightarrow$ du STT décrit :
  - le tir d'un nœud (transmission du flux de contrôle) [transition discrète]
  - le passage du temps [transition continue]

Sémantique des EFFBDs [Seidner et Roux, 2008]

- Inclusion du temps : description de la sémantique sous la forme d'un système de transitions temporisé (STT) $(S, s_0, \mathcal{N}, \rightarrow)$
- Un état $s = (Act, N, \nu) \in S$ représente :
  - l'activité de chaque nœud (inactif, en attente de flux,...) ;
  - le niveau de chaque flux ;
  - pour chaque fonction, le temps écoulé depuis le début de son exécution.
- La relation de transition $\rightarrow$ du STT décrit :
  - le tir d'un nœud (transmission du flux de contrôle) [transition discrète]
  - le passage du temps [transition continue]

Sémantique des EFFBDs [Seidner et Roux, 2008]

- Inclusion du temps : description de la sémantique sous la forme d'un système de transitions temporisé (STT) $(S, s_0, \mathcal{N}, \rightarrow)$
- Un état $s = (Act, N, \nu) \in S$ représente :
  - l'activité de chaque nœud (inactif, en attente de flux,...) ;
  - le niveau de chaque flux ;
  - pour chaque fonction, le temps écoulé depuis le début de son exécution.
- La relation de transition $\rightarrow$ du STT décrit :
  - le tir d'un nœud (transmission du flux de contrôle) [transition discrète]
  - le passage du temps [transition continue]

Sémantique des EFFBDs [Seidner et Roux, 2008]

- Inclusion du temps : description de la sémantique sous la forme d'un système de transitions temporisé (STT) $(S, s_0, \mathcal{N}, \rightarrow)$
- Un état $s = (Act, N, \nu) \in S$ représente :
  - l'activité de chaque nœud (inactif, en attente de flux,...) ;
  - le niveau de chaque flux ;
  - pour chaque fonction, le temps écoulé depuis le début de son exécution.
- La relation de transition $\rightarrow$ du STT décrit :
  - le tir d'un nœud (transmission du flux de contrôle) [transition discrète]
  - le passage du temps [transition continue]

Premiers résultats

- Définition de la classe des EFFBDs bien formés
- Définition de la classe des EFFBDs bornés (niveau des flux)
- Description de conditions suffisantes de bornitude
- Construction d'un outil de simulation des EFFBDs intégré à la plate-forme MDWorkbench :

Premiers résultats

- Définition de la classe des EFFBDs bien formés
- Définition de la classe des EFFBDs bornés (niveau des flux)
- Description de conditions suffisantes de bornitude
- Construction d'un outil de simulation des EFFBDs intégré à la plate-forme MDWorkbench :

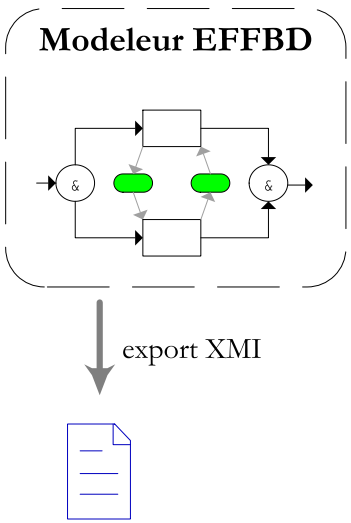
Premiers résultats

- Définition de la classe des EFFBDs bien formés
- Définition de la classe des EFFBDs bornés (niveau des flux)
- Description de conditions suffisantes de bornitude
- Construction d'un outil de simulation des EFFBDs intégré à la plate-forme MDWorkbench :

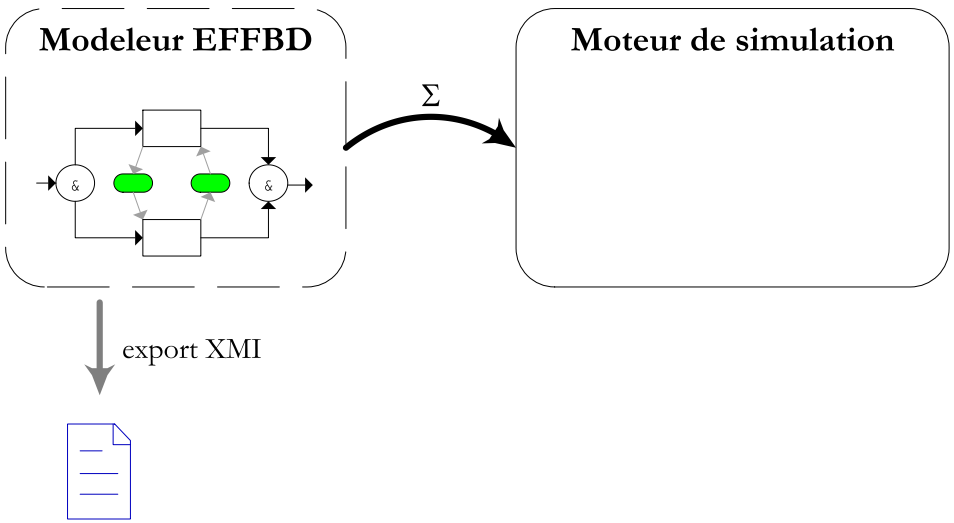
Premiers résultats

- Définition de la classe des EFFBDs bien formés
- Définition de la classe des EFFBDs bornés (niveau des flux)
- Description de conditions suffisantes de bornitude
- Construction d'un outil de simulation des EFFBDs intégré à la plate-forme MDWorkbench :

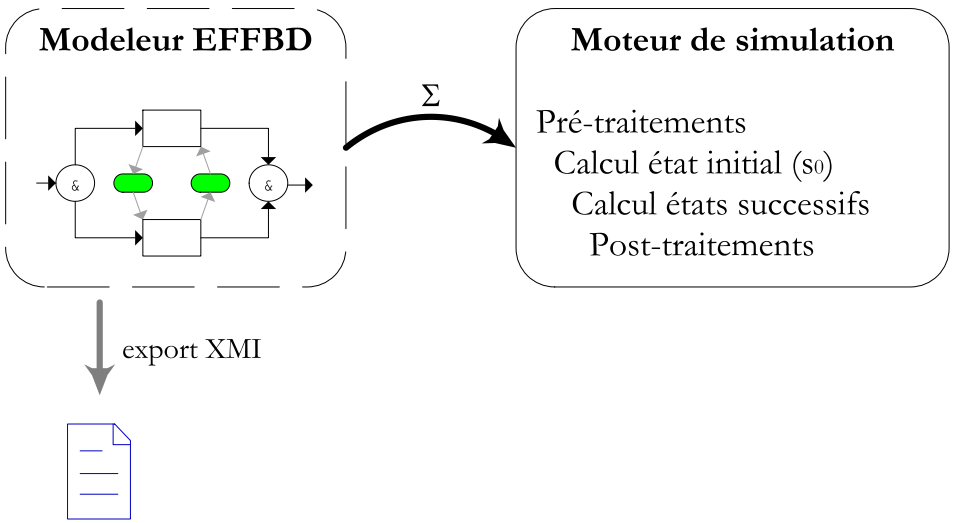
Premiers résultats



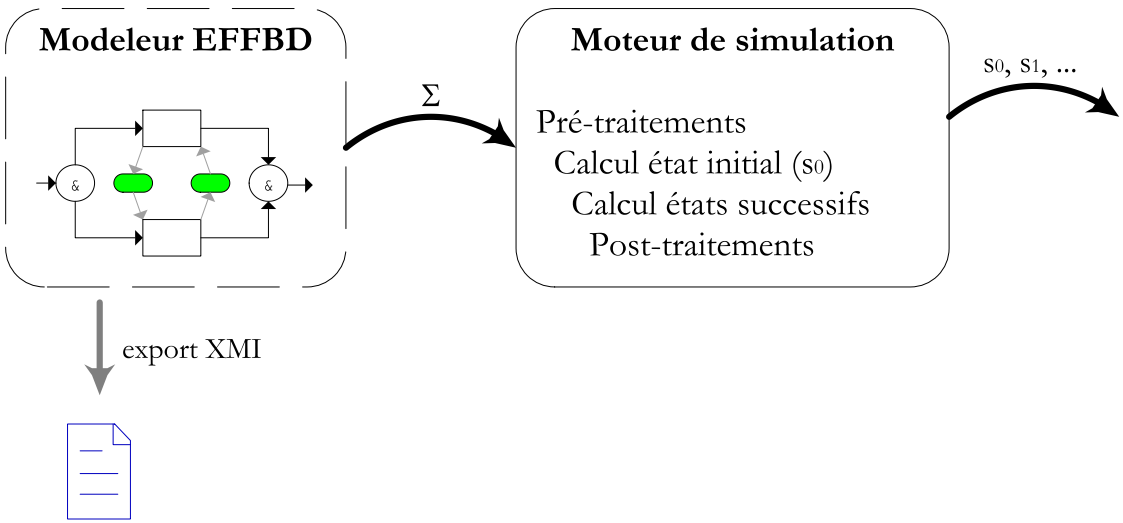
Premiers résultats



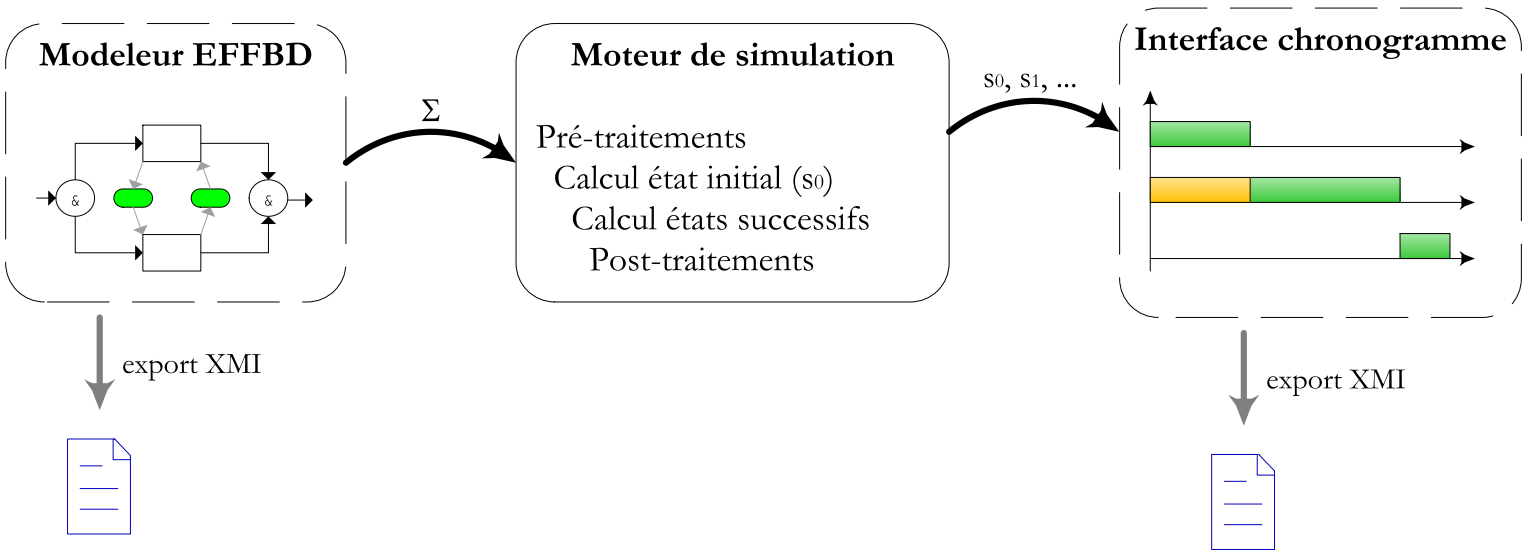
Premiers résultats



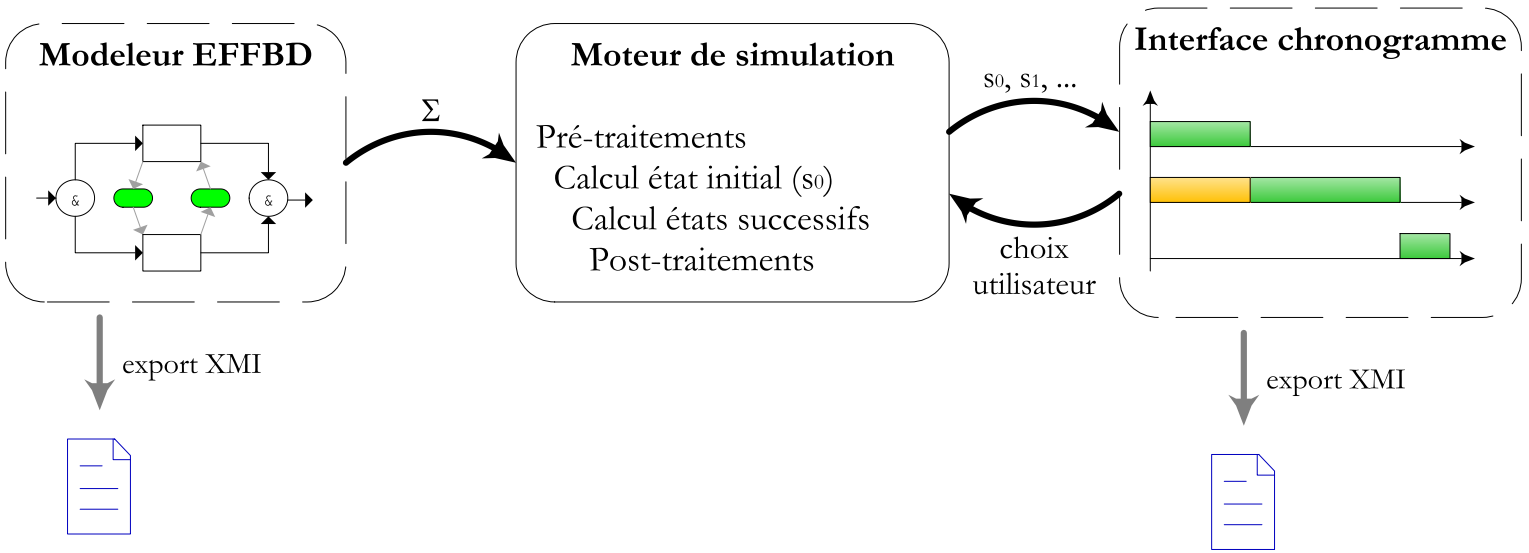
Premiers résultats



Premiers résultats

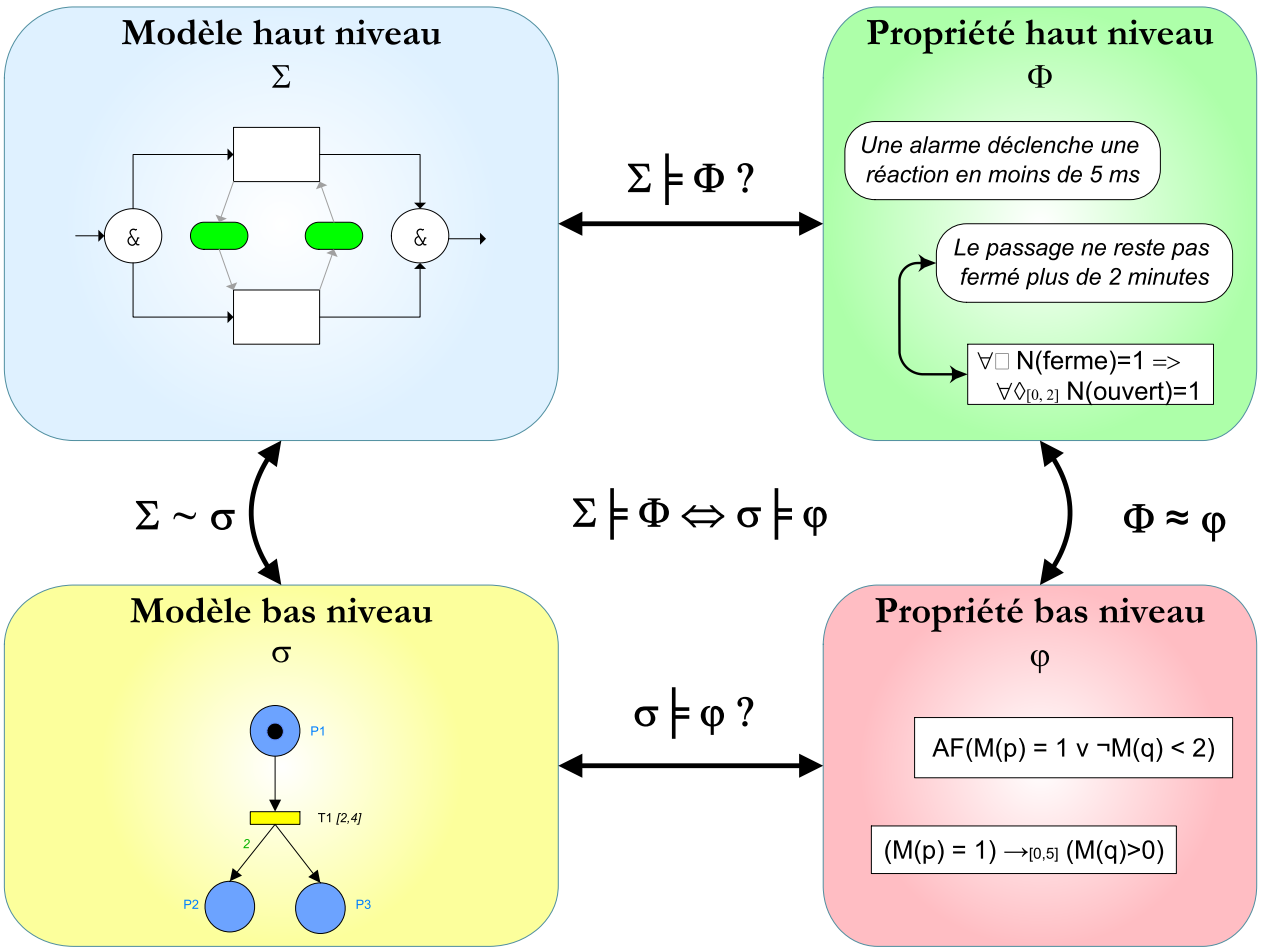


Premiers résultats



Plan de la soutenance

Retour sur la démarche générale



Les réseaux de Petri temporels (Time Petri Nets, TPNs)

- Formalisme constituant le langage « bas-niveau »
- Classe des réseaux T-temporels [Merlin 1974] :
  - étendent les réseaux de Petri classiques [Petri 1962] ;
  - les transitions portent un intervalle temporel ;
  - contrainte sur la date de tir après sensibilisation
- Modélisation de systèmes dynamiques, concurrents, communicants
- Sémantique formelle (STT), nombreux travaux et outils

Les réseaux de Petri temporels (Time Petri Nets, TPNs)

- Formalisme constituant le langage « bas-niveau »
- Classe des réseaux T-temporels [Merlin 1974] :
  - étendent les réseaux de Petri classiques [Petri 1962] ;
  - les transitions portent un intervalle temporel ;
  - contrainte sur la date de tir après sensibilisation
- Modélisation de systèmes dynamiques, concurrents, communicants
- Sémantique formelle (STT), nombreux travaux et outils

Les réseaux de Petri temporels (Time Petri Nets, TPNs)

- Formalisme constituant le langage « bas-niveau »
- Classe des réseaux T-temporels [Merlin 1974] :
  - étendent les réseaux de Petri classiques [Petri 1962] ;
  - les transitions portent un intervalle temporel ;
  - contrainte sur la date de tir après sensibilisation
- Modélisation de systèmes dynamiques, concurrents, communicants
- Sémantique formelle (STT), nombreux travaux et outils

Les réseaux de Petri temporels (Time Petri Nets, TPNs)

- Formalisme constituant le langage « bas-niveau »
- Classe des réseaux T-temporels [Merlin 1974] :
  - étendent les réseaux de Petri classiques [Petri 1962] ;
  - les transitions portent un intervalle temporel ;
  - contrainte sur la date de tir après sensibilisation
- Modélisation de systèmes dynamiques, concurrents, communicants
- Sémantique formelle (STT), nombreux travaux et outils

Les réseaux de Petri temporels (Time Petri Nets, TPNs)

- Formalisme constituant le langage « bas-niveau »
- Classe des réseaux T-temporels [Merlin 1974] :
  - étendent les réseaux de Petri classiques [Petri 1962] ;
  - les transitions portent un intervalle temporel ;
  - contrainte sur la date de tir après sensibilisation
- Modélisation de systèmes dynamiques, concurrents, communicants
- Sémantique formelle (STT), nombreux travaux et outils

Les réseaux de Petri temporels (Time Petri Nets, TPNs)

- Formalisme constituant le langage « bas-niveau »
- Classe des réseaux T-temporels [Merlin 1974] :
  - étendent les réseaux de Petri classiques [Petri 1962] ;
  - les transitions portent un intervalle temporel ;
  - contrainte sur la date de tir après sensibilisation
- Modélisation de systèmes dynamiques, concurrents, communicants
- Sémantique formelle (STT), nombreux travaux et outils

Les réseaux de Petri temporels (Time Petri Nets, TPNs)

- Formalisme constituant le langage « bas-niveau »
- Classe des réseaux T-temporels [Merlin 1974] :
  - étendent les réseaux de Petri classiques [Petri 1962] ;
  - les transitions portent un intervalle temporel ;
  - contrainte sur la date de tir après sensibilisation
- Modélisation de systèmes dynamiques, concurrents, communicants
- Sémantique formelle (STT), nombreux travaux et outils

Principes de la traduction des EFFBDs en TPNs

- Traduction des nœuds et des flux par des motifs élémentaires :
  - places $\leftrightarrow$ états (attente de synchronsation, fonction en exécution...)
  - transitions $\leftrightarrow$ événements (fin d'exécution, entrée dans une structure de contrôle...)
- Écriture de règles de connexion entre motifs
- Obtention d'une structure globale similaire à celle de l'EFFBD source

Principes de la traduction des EFFBDs en TPNs

- Traduction des nœuds et des flux par des motifs élémentaires :
  - places $\leftrightarrow$ états (attente de synchronsation, fonction en exécution...)
  - transitions $\leftrightarrow$ événements (fin d'exécution, entrée dans une structure de contrôle...)
- Écriture de règles de connexion entre motifs
- Obtention d'une structure globale similaire à celle de l'EFFBD source

Principes de la traduction des EFFBDs en TPNs

- Traduction des nœuds et des flux par des motifs élémentaires :
  - places $\leftrightarrow$ états (attente de synchronsation, fonction en exécution...)
  - transitions $\leftrightarrow$ événements (fin d'exécution, entrée dans une structure de contrôle...)
- Écriture de règles de connexion entre motifs
- Obtention d'une structure globale similaire à celle de l'EFFBD source

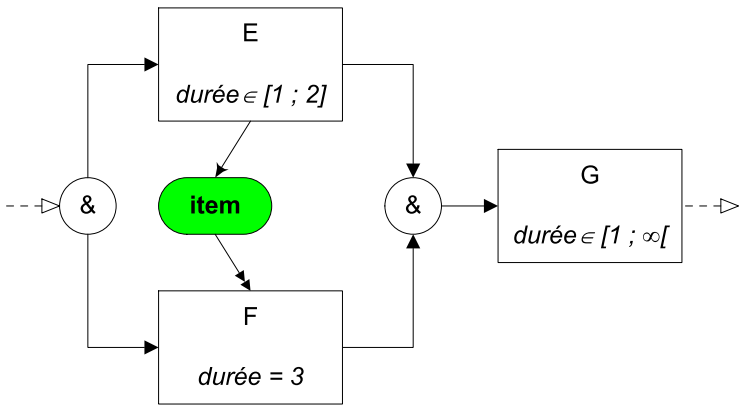
Principes de la traduction des EFFBDs en TPNs

- Traduction des nœuds et des flux par des motifs élémentaires :
  - places $\leftrightarrow$ états (attente de synchronsation, fonction en exécution...)
  - transitions $\leftrightarrow$ événements (fin d'exécution, entrée dans une structure de contrôle...)
- Écriture de règles de connexion entre motifs
- Obtention d'une structure globale similaire à celle de l'EFFBD source

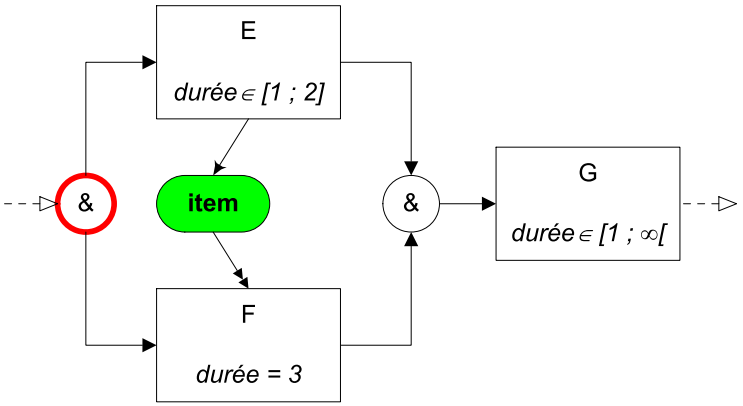
Principes de la traduction des EFFBDs en TPNs

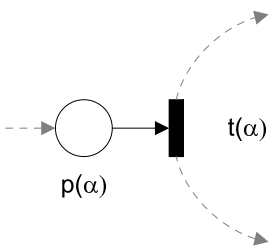
- Traduction des nœuds et des flux par des motifs élémentaires :
  - places $\leftrightarrow$ états (attente de synchronsation, fonction en exécution...)
  - transitions $\leftrightarrow$ événements (fin d'exécution, entrée dans une structure de contrôle...)
- Écriture de règles de connexion entre motifs
- Obtention d'une structure globale similaire à celle de l'EFFBD source

Exemple : traduction du motif de synchronisation par flux

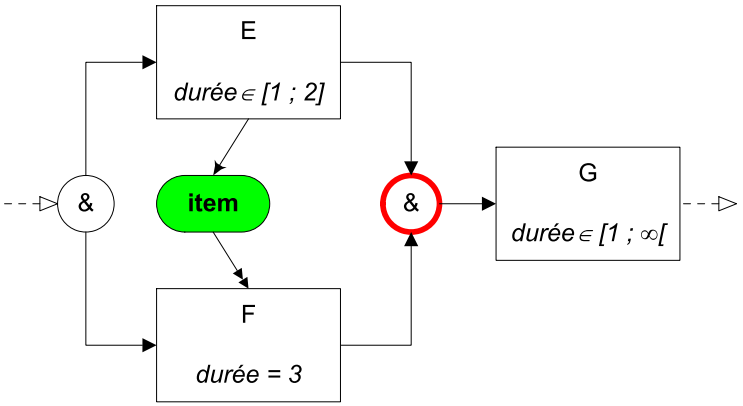


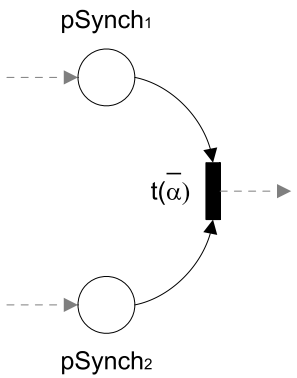
Exemple : traduction du motif de synchronisation par flux



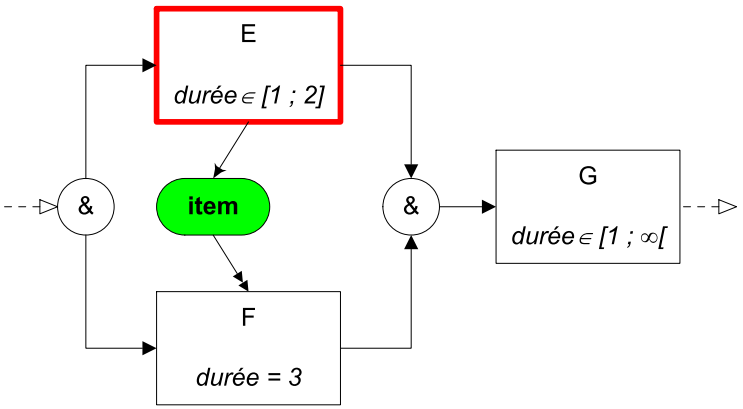


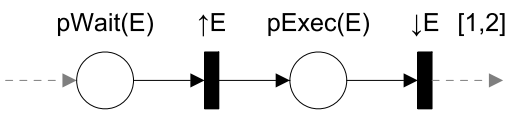
Exemple : traduction du motif de synchronisation par flux



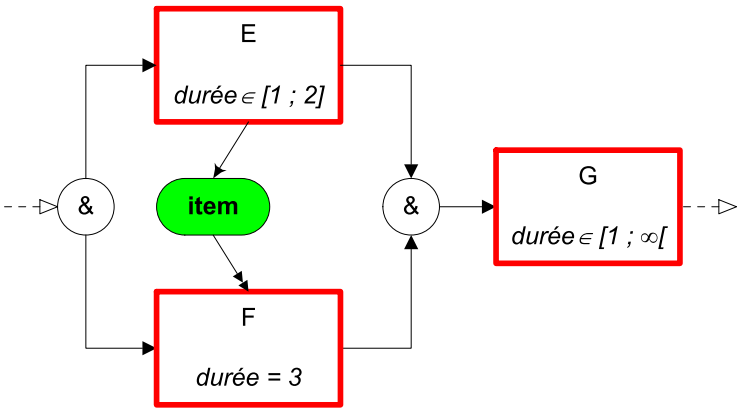


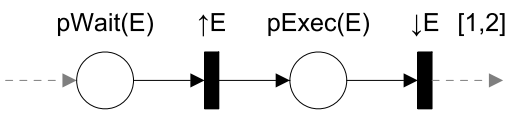
Exemple : traduction du motif de synchronisation par flux

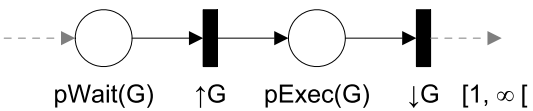


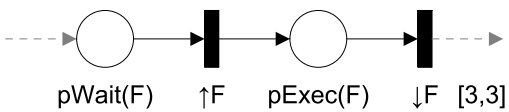


Exemple : traduction du motif de synchronisation par flux

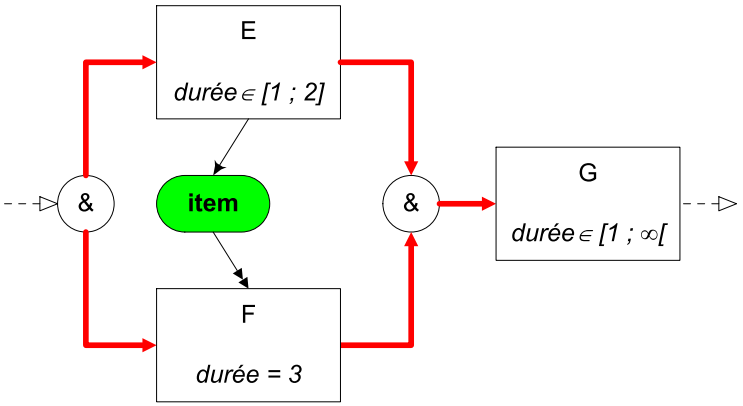


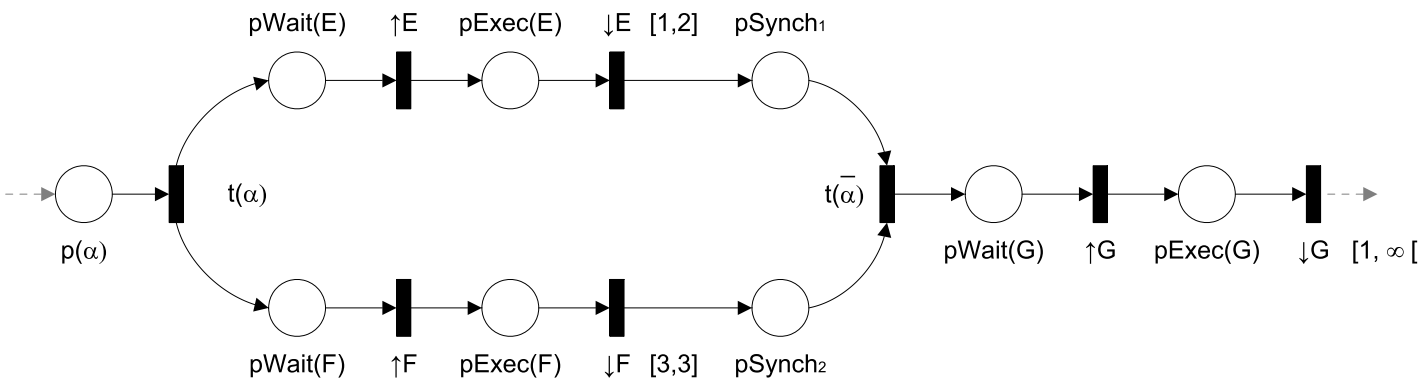




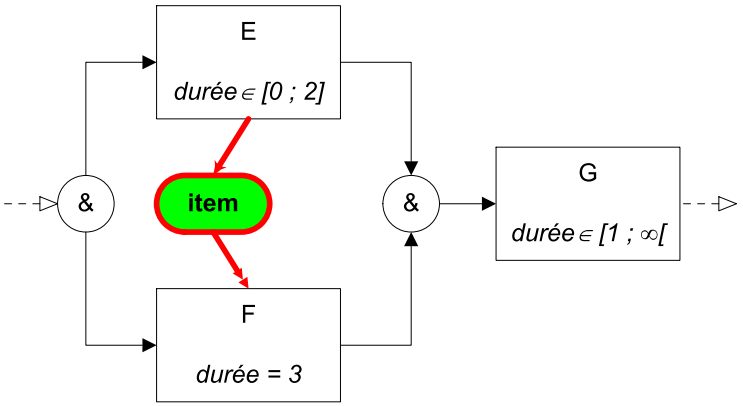


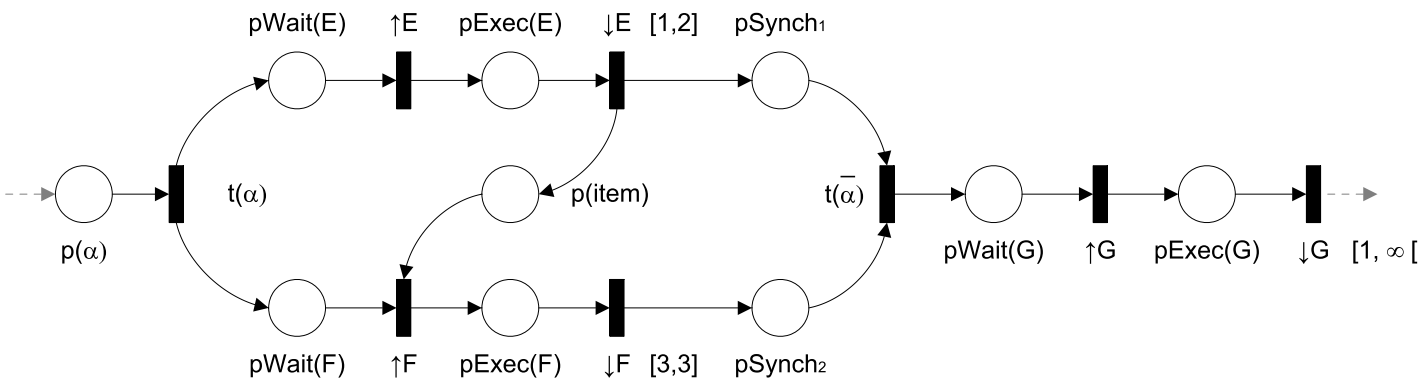
Exemple : traduction du motif de synchronisation par flux



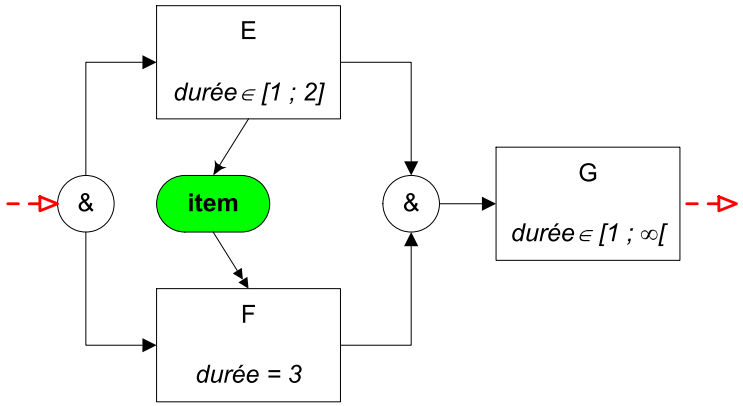


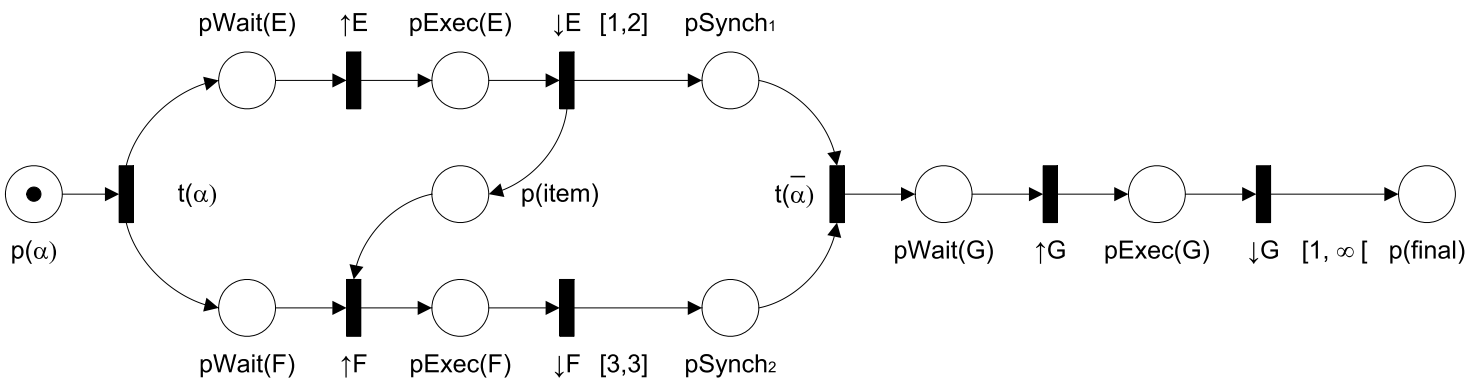
Exemple : traduction du motif de synchronisation par flux





Exemple : traduction du motif de synchronisation par flux





Résultats théoriques

Définition d'une relation $\sim$ entre les états d'un EFFBD et ceux de sa traduction

Proposition

La relation $\sim$ est une bisimulation temporelle forte

Conséquence : la traduction en TPN préserve les comportements

Corollaires

- La k-bornitude d'un EFFBD est décidable
- La bornitude d'un EFFBD est indécidable
- L'accessibilité d'un état d'un EFFBD est décidable
- Le TPN issu de la traduction d'un EFFBD borné est borné

Résultats théoriques

Définition d'une relation $\sim$ entre les états d'un EFFBD et ceux de sa traduction

Proposition

La relation $\sim$ est une bisimulation temporelle forte

Conséquence : la traduction en TPN préserve les comportements

Corollaires

- La k-bornitude d'un EFFBD est décidable
- La bornitude d'un EFFBD est indécidable
- L'accessibilité d'un état d'un EFFBD est décidable
- Le TPN issu de la traduction d'un EFFBD borné est borné

Résultats théoriques

Définition d'une relation $\sim$ entre les états d'un EFFBD et ceux de sa traduction

Proposition

La relation $\sim$ est une bisimulation temporelle forte

Conséquence : la traduction en TPN préserve les comportements

Corollaires

- La k-bornitude d'un EFFBD est décidable
- La bornitude d'un EFFBD est indécidable
- L'accessibilité d'un état d'un EFFBD est décidable
- Le TPN issu de la traduction d'un EFFBD borné est borné

Résultats théoriques

Définition d'une relation $\sim$ entre les états d'un EFFBD et ceux de sa traduction

Proposition

La relation $\sim$ est une bisimulation temporelle forte

Conséquence : la traduction en TPN préserve les comportements

Corollaires

- La k-bornitude d'un EFFBD est décidable
- La bornitude d'un EFFBD est indécidable
- L'accessibilité d'un état d'un EFFBD est décidable
- Le TPN issu de la traduction d'un EFFBD borné est borné

Résultats théoriques

Définition d'une relation $\sim$ entre les états d'un EFFBD et ceux de sa traduction

Proposition

La relation $\sim$ est une bisimulation temporelle forte

Conséquence : la traduction en TPN préserve les comportements

Corollaires

- La k-bornitude d'un EFFBD est décidable
- La bornitude d'un EFFBD est indécidable
- L'accessibilité d'un état d'un EFFBD est décidable
- Le TPN issu de la traduction d'un EFFBD borné est borné

Résultats théoriques

Définition d'une relation $\sim$ entre les états d'un EFFBD et ceux de sa traduction

Proposition

La relation $\sim$ est une bisimulation temporelle forte

Conséquence : la traduction en TPN préserve les comportements

Corollaires

- La k-bornitude d'un EFFBD est décidable
- La bornitude d'un EFFBD est indécidable
- L'accessibilité d'un état d'un EFFBD est décidable
- Le TPN issu de la traduction d'un EFFBD borné est borné

Résultats théoriques

Définition d'une relation $\sim$ entre les états d'un EFFBD et ceux de sa traduction

Proposition

La relation $\sim$ est une bisimulation temporelle forte

Conséquence : la traduction en TPN préserve les comportements

Corollaires

- La k-bornitude d'un EFFBD est décidable
- La bornitude d'un EFFBD est indécidable
- L'accessibilité d'un état d'un EFFBD est indécidable
- Le TPN issu de la traduction d'un EFFBD borné est borné

Résultats théoriques

Définition d'une relation $\sim$ entre les états d'un EFFBD et ceux de sa traduction

Proposition

La relation $\sim$ est une bisimulation temporelle forte

Conséquence : la traduction en TPN préserve les comportements

Corollaires

- La k-bornitude d'un EFFBD est décidable
- La bornitude d'un EFFBD est indécidable
- L'accessibilité d'un état d'un EFFBD est indécidable
- Le TPN issu de la traduction d'un EFFBD borné est borné

Résultats théoriques

Définition d'une relation $\sim$ entre les états d'un EFFBD et ceux de sa traduction

Proposition

La relation $\sim$ est une bisimulation temporelle forte

Conséquence : la traduction en TPN préserve les comportements

Corollaires

- La k-bornitude d'un EFFBD est décidable
- La bornitude d'un EFFBD est indécidable
- L'accessibilité d'un état d'un EFFBD borné est décidable
- Le TPN issu de la traduction d'un EFFBD borné est borné

Expression de propriétés logiques

- Utilisation de la logique *TCTL* [Alur *et al.* 1990] : propriétés de sûreté et de vivacité temporelles quantitatives :

  « Un train à l'approche déclenche toujours la fermeture du passage à niveau en moins de 25 secondes »

  $$\forall\Box\ \textit{Train approchant} \Rightarrow (\forall\Diamond_{[0,25]}\ \textit{Passage fermé})$$

- Extension aux TPNs et aux EFFBDs
- Définition d'une relation $\approx$ entre une propriété $\varphi_{\mathcal{E}}$ et sa traduction $\varphi_{\mathcal{T}}$
- Définition de classes de propriétés « haut-niveau » (exclusion mutuelle, réponse bornée, non-blocage...) : formalisation de propriétés métiers

Expression de propriétés logiques

- Utilisation de la logique *TCTL* [Alur *et al.* 1990] : propriétés de sûreté et de vivacité temporelles quantitatives :

  « Un train à l'approche déclenche toujours la fermeture du passage à niveau en moins de 25 secondes »

  $$\forall\Box\ \mathit{Train\ approchant} \Rightarrow (\forall\Diamond_{[0,25]}\ \mathit{Passage\ fermé})$$

- Extension aux TPNs et aux EFFBDs
- Définition d'une relation $\approx$ entre une propriété $\varphi_{\mathcal{E}}$ et sa traduction $\varphi_{\mathcal{T}}$
- Définition de classes de propriétés « haut-niveau » (exclusion mutuelle, réponse bornée, non-blocage...) : formalisation de propriétés métiers

Expression de propriétés logiques

- Utilisation de la logique *TCTL* [Alur *et al.* 1990] : propriétés de sûreté et de vivacité temporelles quantitatives :

  « Un train à l'approche déclenche toujours la fermeture du passage à niveau en moins de 25 secondes »

  $$\forall\Box\ \textit{Train approchant} \Rightarrow (\forall\Diamond_{[0,25]}\ \textit{Passage fermé})$$

- Extension aux TPNs et aux EFFBDs
- Définition d'une relation $\approx$ entre une propriété $\varphi_{\mathcal{E}}$ et sa traduction $\varphi_{\mathcal{T}}$
- Définition de classes de propriétés « haut-niveau » (exclusion mutuelle, réponse bornée, non-blocage...) : formalisation de propriétés métiers

Expression de propriétés logiques

- Utilisation de la logique *TCTL* [Alur *et al.* 1990] : propriétés de sûreté et de vivacité temporelles quantitatives :

  « Un train à l'approche déclenche toujours la fermeture du passage à niveau en moins de 25 secondes »

  $$\forall\Box\ \textit{Train approchant} \Rightarrow (\forall\Diamond_{[0,25]}\ \textit{Passage fermé})$$

- Extension aux TPNs et aux EFFBDs
- Définition d'une relation $\approx$ entre une propriété $\varphi_{\mathcal{E}}$ et sa traduction $\varphi_{\mathcal{T}}$
- Définition de classes de propriétés « haut-niveau » (exclusion mutuelle, réponse bornée, non-blocage...) : formalisation de propriétés métiers

Expression de propriétés logiques

- Utilisation de la logique *TCTL* [Alur *et al.* 1990] : propriétés de sûreté et de vivacité temporelles quantitatives :

  « Un train à l'approche déclenche toujours la fermeture du passage à niveau en moins de 25 secondes »

  $$\forall\Box\ \textit{Train approchant} \Rightarrow (\forall\Diamond_{[0,25]}\ \textit{Passage fermé})$$

- Extension aux TPNs et aux EFFBDs
- Définition d'une relation $\approx$ entre une propriété $\varphi_{\mathcal{E}}$ et sa traduction $\varphi_{\mathcal{T}}$
- Définition de classes de propriétés « haut-niveau » (exclusion mutuelle, réponse bornée, non-blocage...) : formalisation de propriétés métiers

Expression de propriétés logiques

- Utilisation de la logique *TCTL* [Alur *et al.* 1990] : propriétés de sûreté et de vivacité temporelles quantitatives :

  « Un train à l'approche déclenche toujours la fermeture du passage à niveau en moins de 25 secondes »

  $$\forall\Box\ \textit{Train approchant} \Rightarrow (\forall\Diamond_{[0,25]}\ \textit{Passage fermé})$$

- Extension aux TPNs et aux EFFBDs
- Définition d'une relation $\approx$ entre une propriété $\varphi_{\mathcal{E}}$ et sa traduction $\varphi_{\mathcal{T}}$
- Définition de classes de propriétés « haut-niveau » (exclusion mutuelle, réponse bornée, non-blocage...) : formalisation de propriétés métiers

Résultats théoriques

Proposition

Soient $\mathcal{E}$ et $\mathcal{T}$ sa traduction tels que $\mathcal{E} \sim \mathcal{T}$.
Soient $\varphi_{\mathcal{E}}$ et $\varphi_{\mathcal{T}}$ sa traduction tels que $\varphi_{\mathcal{E}} \approx \varphi_{\mathcal{T}}$.
Alors $\mathcal{E} \models \varphi_{\mathcal{E}} \Leftrightarrow \mathcal{T} \models \varphi_{\mathcal{T}}$

Conséquence : notre démarche générale est correcte

Corollaires

La satisfaisabilité de la logique EFFBD-TCTL est décidable sur les EFFBDs

Proposition

Le model-checking d'EFFBD-TCTL sur les EFFBDs bornés est décidable

Résultats théoriques

Proposition

Soient $\mathcal{E}$ et $\mathcal{T}$ sa traduction tels que $\mathcal{E} \sim \mathcal{T}$.
Soient $\varphi_{\mathcal{E}}$ et $\varphi_{\mathcal{T}}$ sa traduction tels que $\varphi_{\mathcal{E}} \approx \varphi_{\mathcal{T}}$.
Alors $\mathcal{E} \models \varphi_{\mathcal{E}} \Leftrightarrow \mathcal{T} \models \varphi_{\mathcal{T}}$

Conséquence : notre démarche générale est correcte

Corollaires

La satisfaisabilité de la logique EFFBD-TCTL est décidable sur les EFFBDs

Proposition

Le model-checking d'EFFBD-TCTL sur les EFFBDs bornés est décidable

Résultats théoriques

Proposition

Soient $\mathcal{E}$ et $\mathcal{T}$ sa traduction tels que $\mathcal{E} \sim \mathcal{T}$.
Soient $\varphi_{\mathcal{E}}$ et $\varphi_{\mathcal{T}}$ sa traduction tels que $\varphi_{\mathcal{E}} \approx \varphi_{\mathcal{T}}$.
Alors $\mathcal{E} \models \varphi_{\mathcal{E}} \Leftrightarrow \mathcal{T} \models \varphi_{\mathcal{T}}$

Conséquence : notre démarche générale est correcte

Corollaires

La satisfaisabilité de la logique EFFBD-TCTL est indécidable sur les EFFBDs

Proposition

Le model-checking d'EFFBD-TCTL sur les EFFBDs bornés est décidable

Résultats théoriques

Proposition

Soient $\mathcal{E}$ et $\mathcal{T}$ sa traduction tels que $\mathcal{E} \sim \mathcal{T}$.
Soient $\varphi_{\mathcal{E}}$ et $\varphi_{\mathcal{T}}$ sa traduction tels que $\varphi_{\mathcal{E}} \approx \varphi_{\mathcal{T}}$.
Alors $\mathcal{E} \models \varphi_{\mathcal{E}} \Leftrightarrow \mathcal{T} \models \varphi_{\mathcal{T}}$

Conséquence : notre démarche générale est correcte

Corollaires

La satisfaisabilité de la logique EFFBD-TCTL est décidable sur les EFFBDs bornés

Proposition

Le model-checking d'EFFBD-TCTL sur les EFFBDs bornés est décidable

Résultats théoriques

Proposition

Soient $\mathcal{E}$ et $\mathcal{T}$ sa traduction tels que $\mathcal{E} \sim \mathcal{T}$.
Soient $\varphi_{\mathcal{E}}$ et $\varphi_{\mathcal{T}}$ sa traduction tels que $\varphi_{\mathcal{E}} \approx \varphi_{\mathcal{T}}$.
Alors $\mathcal{E} \models \varphi_{\mathcal{E}} \Leftrightarrow \mathcal{T} \models \varphi_{\mathcal{T}}$

Conséquence : notre démarche générale est correcte

Corollaires

La satisfaisabilité de la logique EFFBD-TCTL est décidable sur les EFFBDs bornés

Proposition

Le model-checking d'EFFBD-TCTL sur les EFFBDs bornés est décidable

Résultats théoriques

Proposition

Soient $\mathcal{E}$ et $\mathcal{T}$ sa traduction tels que $\mathcal{E} \sim \mathcal{T}$.
Soient $\varphi_{\mathcal{E}}$ et $\varphi_{\mathcal{T}}$ sa traduction tels que $\varphi_{\mathcal{E}} \approx \varphi_{\mathcal{T}}$.
Alors $\mathcal{E} \models \varphi_{\mathcal{E}} \Leftrightarrow \mathcal{T} \models \varphi_{\mathcal{T}}$

Conséquence : notre démarche générale est correcte

Corollaires

La satisfaisabilité de la logique EFFBD-TCTL est décidable sur les EFFBDs bornés

Proposition

Le model-checking d'EFFBD-TCTL sur les EFFBDs bornés est décidable et PSPACE-complet

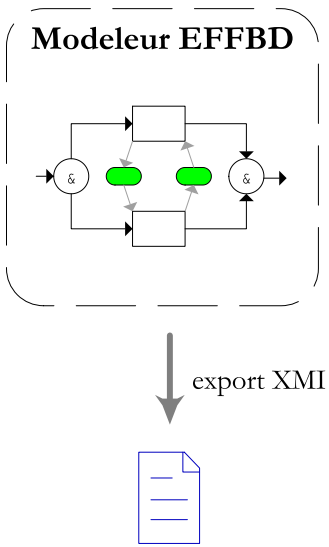
Résultats pratiques

Développement d'un outil de vérification intégré à MDWorkbench

Résultats d'analyse : traces de haut-niveau (sur les fonctions du modèle)

Résultats pratiques

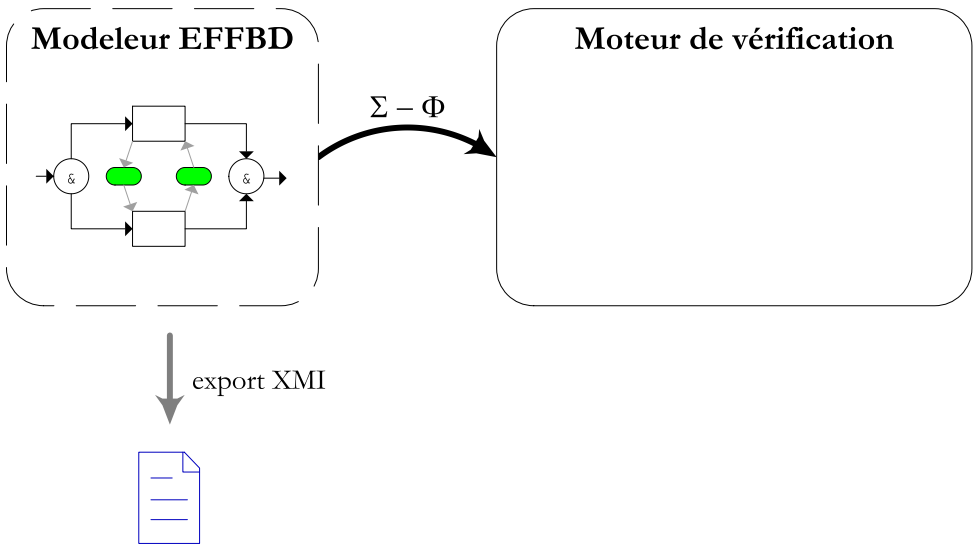
Développement d'un outil de vérification intégré à MDWorkbench



Résultats d'analyse : traces de haut-niveau (sur les fonctions du modèle)

Résultats pratiques

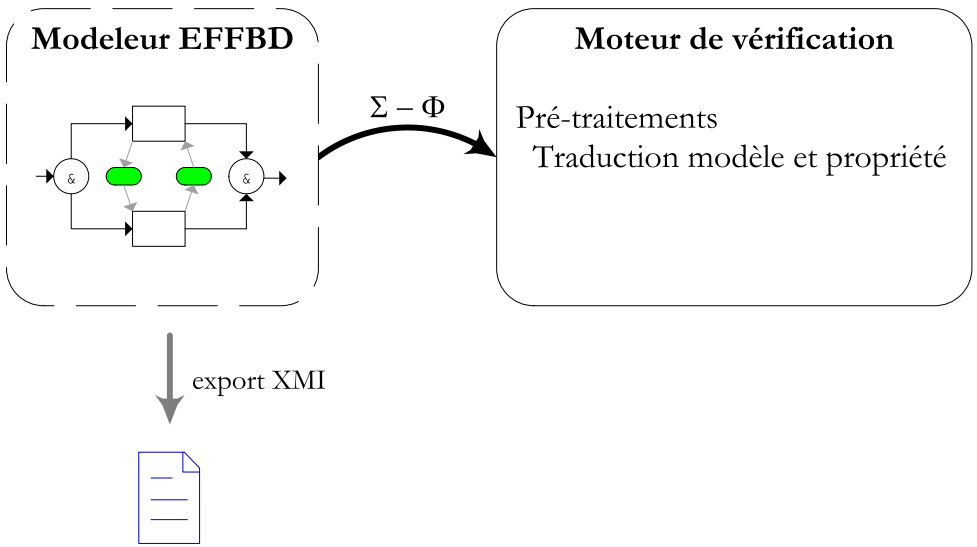
Développement d'un outil de vérification intégré à MDWorkbench



Résultats d'analyse : traces de haut-niveau (sur les fonctions du modèle)

Résultats pratiques

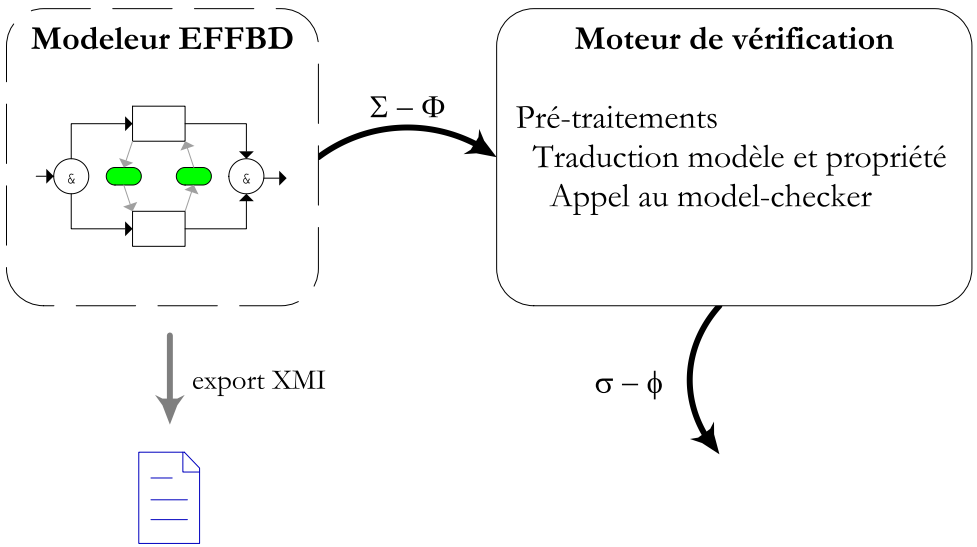
Développement d'un outil de vérification intégré à MDWorkbench



Résultats d'analyse : traces de haut-niveau (sur les fonctions du modèle)

Résultats pratiques

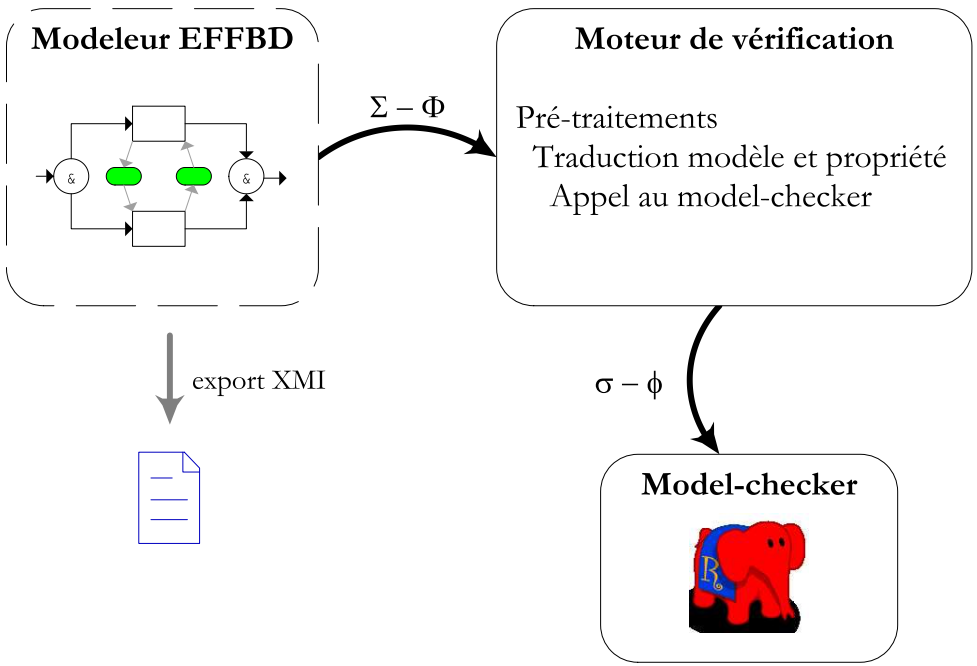
Développement d'un outil de vérification intégré à MDWorkbench



Résultats d'analyse : traces de haut-niveau (sur les fonctions du modèle)

Résultats pratiques

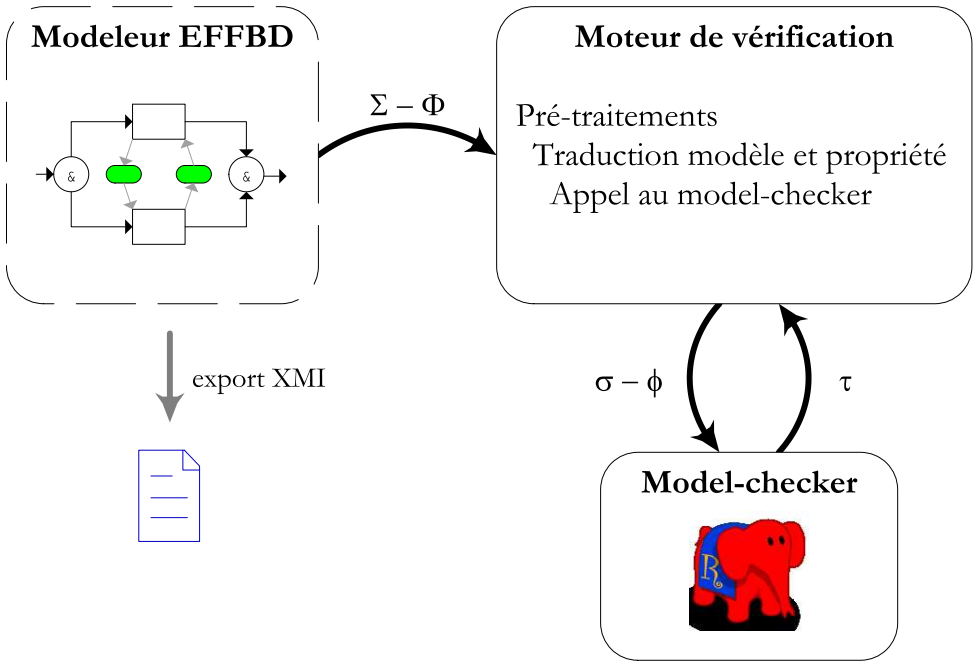
Développement d'un outil de vérification intégré à MDWorkbench



Résultats d'analyse : traces de haut-niveau (sur les fonctions du modèle)

Résultats pratiques

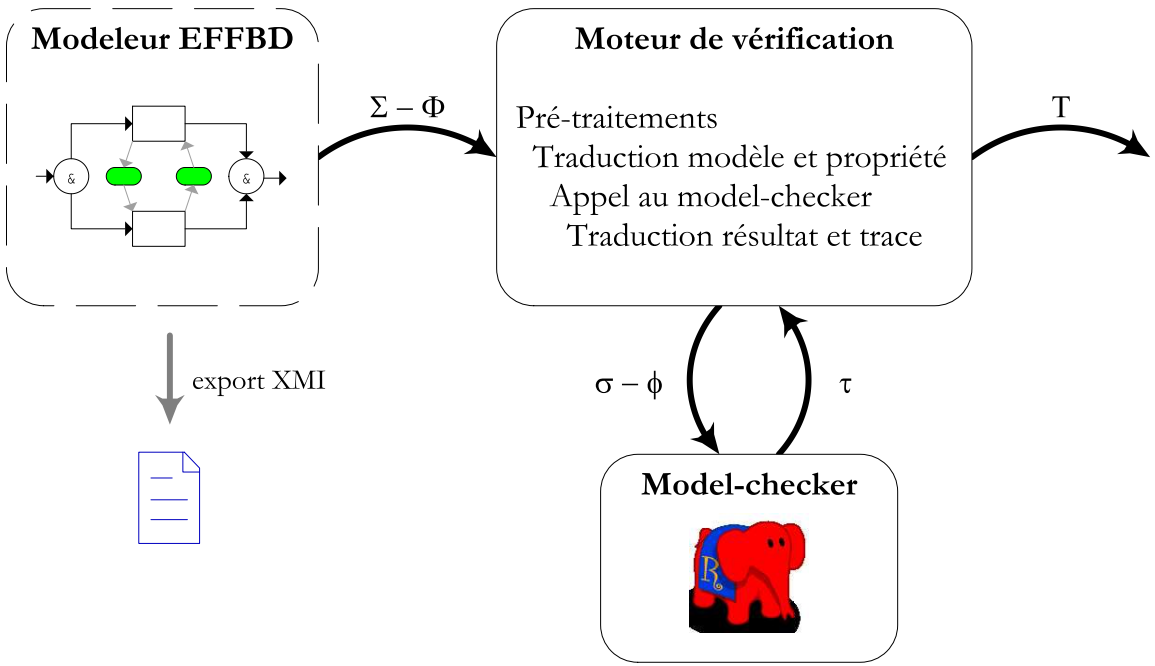
Développement d'un outil de vérification intégré à MDWorkbench



Résultats d'analyse : traces de haut-niveau (sur les fonctions du modèle)

Résultats pratiques

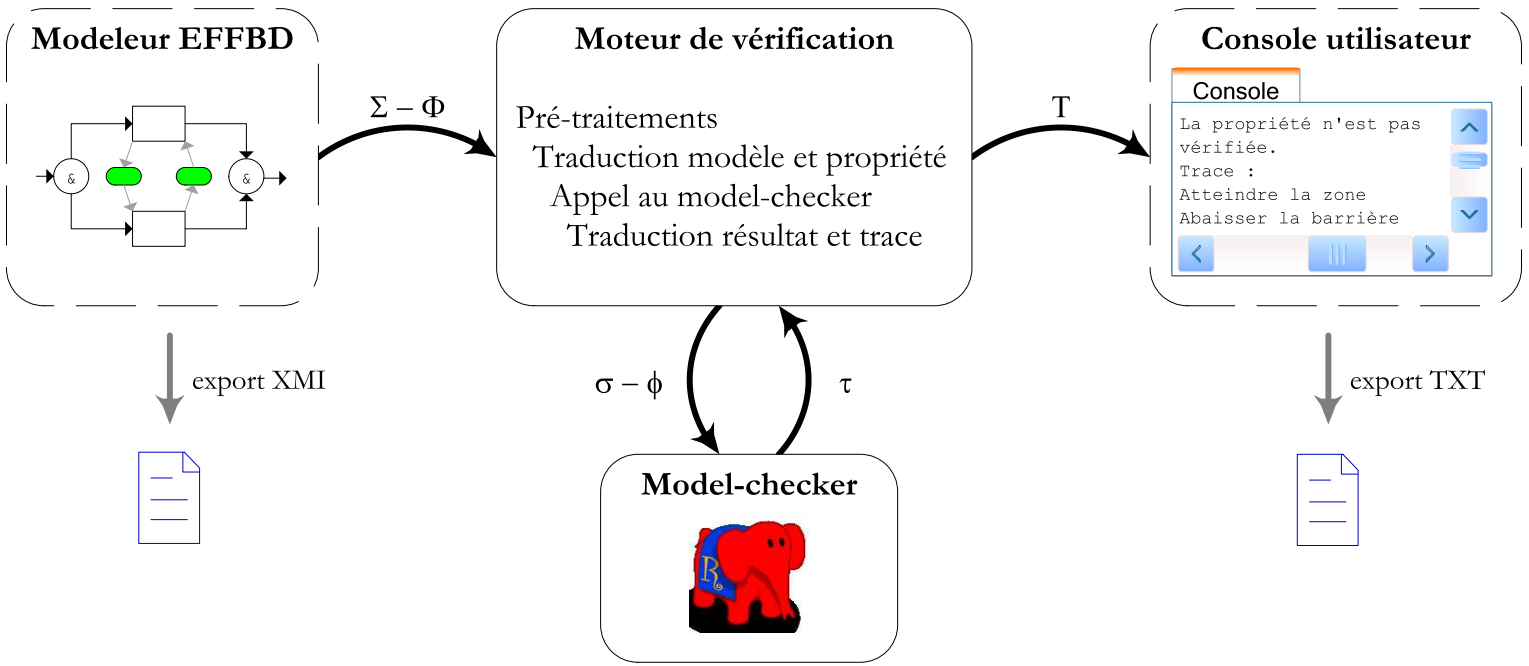
Développement d'un outil de vérification intégré à MDWorkbench



Résultats d'analyse : traces de haut-niveau (sur les fonctions du modèle)

Résultats pratiques

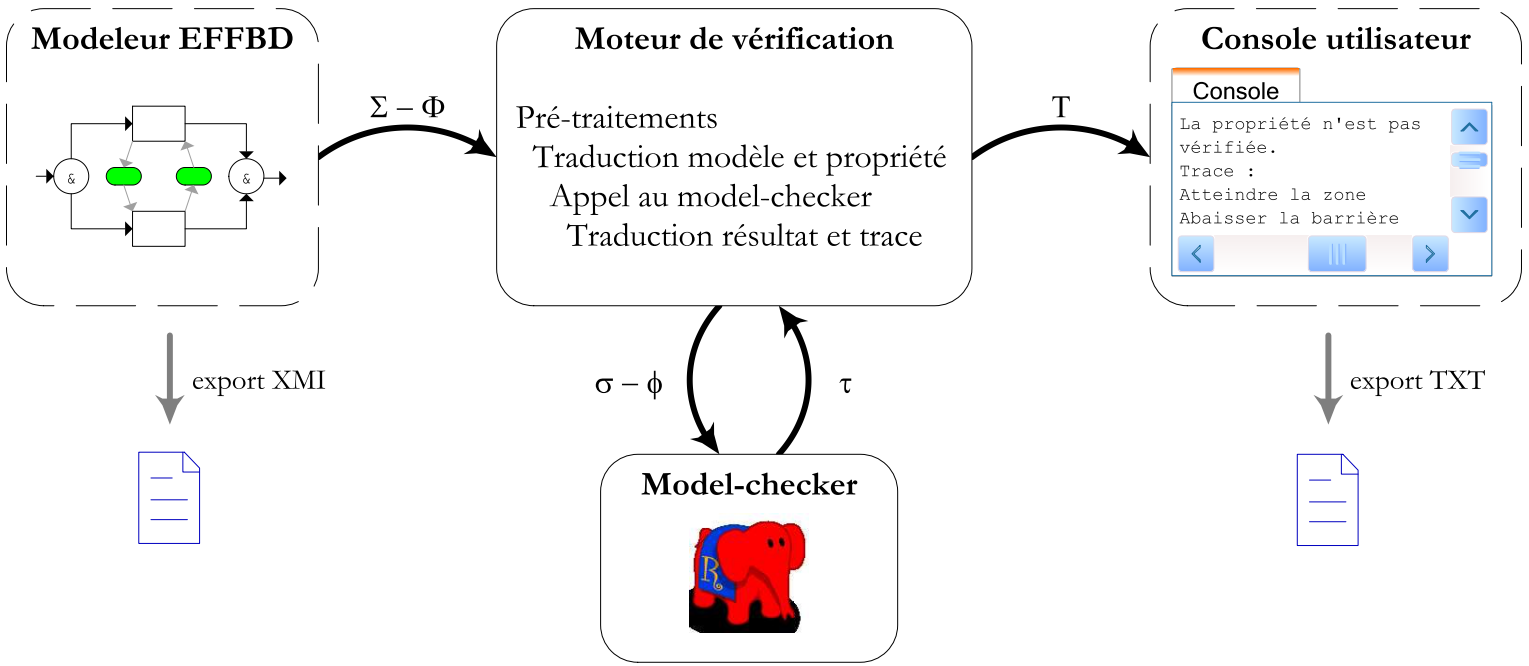
Développement d'un outil de vérification intégré à MDWorkbench



Résultats d'analyse : traces de haut-niveau (sur les fonctions du modèle)

Résultats pratiques

Développement d'un outil de vérification intégré à MDWorkbench



Résultats d'analyse : traces de haut-niveau (sur les fonctions du modèle)

Extension aux modèles défaillants

- Définition de cas de pannes sur les flux et les fonctions
- Extension :
  - de la syntaxe et de la sémantique des EFFBDs ;
  - des motifs de traduction ;
  - de la logique EFFBD-TCTL ;
  - des propriétés-types offertes à la vérification ;
  - des outils de simulation et de vérification.

Extension aux modèles défaillants

- Définition de cas de pannes sur les flux et les fonctions
- Extension :
  - de la syntaxe et de la sémantique des EFFBDs ;
  - des motifs de traduction ;
  - de la logique EFFBD-TCTL ;
  - des propriétés-types offertes à la vérification ;
  - des outils de simulation et de vérification.

Extension aux modèles défaillants

- Définition de cas de pannes sur les flux et les fonctions
- Extension :
  - de la syntaxe et de la sémantique des EFFBDs ;
  - des motifs de traduction ;
  - de la logique EFFBD-TCTL ;
  - des propriétés-types offertes à la vérification ;
  - des outils de simulation et de vérification.

Extension aux modèles défaillants

- Définition de cas de pannes sur les flux et les fonctions
- Extension :
  - de la syntaxe et de la sémantique des EFFBDs ;
  - des motifs de traduction ;
  - de la logique EFFBD-TCTL ;
  - des propriétés-types offertes à la vérification ;
  - des outils de simulation et de vérification.

Extension aux modèles défaillants

- Définition de cas de pannes sur les flux et les fonctions
- Extension :
  - de la syntaxe et de la sémantique des EFFBDs ;
  - des motifs de traduction ;
  - de la logique EFFBD-TCTL ;
  - des propriétés-types offertes à la vérification ;
  - des outils de simulation et de vérification.

Extension aux modèles défaillants

- Définition de cas de pannes sur les flux et les fonctions
- Extension :
  - de la syntaxe et de la sémantique des EFFBDs ;
  - des motifs de traduction ;
  - de la logique EFFBD-TCTL ;
  - des propriétés-types offertes à la vérification ;
  - des outils de simulation et de vérification.

Extension aux modèles défaillants

- Définition de cas de pannes sur les flux et les fonctions
- Extension :
  - de la syntaxe et de la sémantique des EFFBDs ;
  - des motifs de traduction ;
  - de la logique EFFBD-TCTL ;
  - des propriétés-types offertes à la vérification ;
  - des outils de simulation et de vérification.

Plan de la soutenance

Conclusion

En réponse à nos problématiques, nous avons proposé :

- la formalisation d'un langage haut-niveau, intuitif, riche et évolutif ;
- sa transformation vers un langage bas-niveau permettant l'analyse de propriétés complexes ;
- l'intégration dans un outil de conception d'IS, cachant la complexité à l'utilisateur

Conclusion

En réponse à nos problématiques, nous avons proposé :

- la formalisation d'un langage haut-niveau, intuitif, riche et évolutif ;
- sa transformation vers un langage bas-niveau permettant l'analyse de propriétés complexes ;
- l'intégration dans un outil de conception d'IS, cachant la complexité à l'utilisateur

Conclusion

En réponse à nos problématiques, nous avons proposé :

- la formalisation d'un langage haut-niveau, intuitif, riche et évolutif ;
- sa transformation vers un langage bas-niveau permettant l'analyse de propriétés complexes ;
- l'intégration dans un outil de conception d'IS, cachant la complexité à l'utilisateur

Conclusion

En réponse à nos problématiques, nous avons proposé :

- la formalisation d'un langage haut-niveau, intuitif, riche et évolutif ;
- sa transformation vers un langage bas-niveau permettant l'analyse de propriétés complexes ;
- l'intégration dans un outil de conception d'IS, cachant la complexité à l'utilisateur

Principales publications

- C. Seidner et O. (H.) Roux
  Formal Methods for Systems Engineering Behavior Models
  IEEE Transactions on Industrial Informatics, nov. 2008

- C. Seidner, J.P. Lerat et O. (H.) Roux
  Usability of formal verification on EFFBD models: Applying Petri nets to Systems Engineering issues
  17th International Symposium of the INCOSE, juin 2007

- C. Seidner, J.P. Lerat et O. (H.) Roux
  Behavior Diagrams Model-Checking: Formal Methods Applied to Systems Engineering and Design
  6th Annual Conference on Systems Engineering Research, avril 2008

- C. Seidner, J.P. Lerat et O. (H.) Roux
  Formal Verification in System Design Process: from EFFBDs to Petri nets
  18th International Symposium of the INCOSE, juin 2008

Principales publications

- C. Seidner et O. (H.) Roux
  Formal Methods for Systems Engineering Behavior Models
  IEEE Transactions on Industrial Informatics, nov. 2008

- C. Seidner, J.P. Lerat et O. (H.) Roux
  Usability of formal verification on EFFBD models: Applying Petri nets to Systems Engineering issues
  17th International Symposium of the INCOSE, juin 2007

- C. Seidner, J.P. Lerat et O. (H.) Roux
  Behavior Diagrams Model-Checking: Formal Methods Applied to Systems Engineering and Design
  6th Annual Conference on Systems Engineering Research, avril 2008

- C. Seidner, J.P. Lerat et O. (H.) Roux
  Formal Verification in System Design Process: from EFFBDs to Petri nets
  18th International Symposium of the INCOSE, juin 2008

Perspectives

- Poursuite des développements (retour d'expérience)
- Extension de la sémantique des EFFBDs (quantités « hybrides »)
- Conception et développement d'un model-checker basé sur les EFFBDs

Perspectives

- Poursuite des développements (retour d'expérience)
- Extension de la sémantique des EFFBDs (quantités « hybrides »)
- Conception et développement d'un model-checker basé sur les EFFBDs

Perspectives

- Poursuite des développements (retour d'expérience)
- Extension de la sémantique des EFFBDs (quantités « hybrides »)
- Conception et développement d'un model-checker basé sur les EFFBDs

Merci de votre attention !

Bibliographie

- J.-P. Meinadier
  Ingénierie et intégration des systèmes
  Études et logiciels informatiques. Hermès, Paris, 1998

- J. Long
  Relationships between common graphical representations in Systems Engineering
  5th International Symposium of the INCOSE, St. Louis, MO, juil. 1995

- P. M. Merlin
  A study of recoverability of computing systems
  Thèse de doctorat, University of California, Irvine, CA, 1974

- C. A. Petri
  Kommunikation mit Automaten
  Thèse de doctorat, Institut für instrumentelle Mathematik, Bonn, 1962

- R. Alur, C. A. Courcoubetis et D. L. Dill
  Model-checking for real-time systems
  5th IEEE Symposium on Logic in Computer Science, p. 414–425, Philadelphia, PA, juin 1990