

- École Nationale Supérieure
des Télécommunications de Bretagne



Disponibilité

Ahmed Bouabdallah, Nora Cuppens et Frédéric Cuppens

Rennes le 18 Novembre 2003

www.enst-bretagne.fr



■ Plan

- Etat de l'art
 - Modèle de Yu et Gligor
 - Modèle de Millen
 - Modèle FCCS
- Expression d'une politique de disponibilité dans Or-BAC
- Cahier des charges pour une étude de cas
- Cadre compositionnel pour la disponibilité

■ Modèle de Yu et Gligor

■ Disponibilité

- Garantir qu'un service est fourni en respectant un temps d'attente maximum, et cela malgré des accès concurrents réalisés par divers groupes d'utilisateurs

■ Concepts du modèle

- Politique de temps d'attente fini et de temps d'attente borné
- Notion de contrat d'utilisateur
- Politique de loyauté
- Politique de simultanéité
- Spécification de services
- Modèle d'allocation de ressources

■ Modèle de Yu et Gligor (suite)

■ Contrat d'utilisateurs

- Contraintes imposées aux utilisateurs de services

■ Exemples :

- ➔ Obligation pour un utilisateur de libérer une ressource dès lors qu'il n'en a plus besoin, à la fin de la tâche qui l'utilisait
- ➔ Interdiction de demander une ressource qui ne serait pas utile pour réaliser une tâche

■ Les utilisateurs peuvent violer leur contrat d'utilisateur

- ➔ Besoin de contrôles externes aux services effectués avant d'accorder l'accès au service et pendant l'utilisation du service

■ Modèle de Yu et Gligor (suite)

- Spécification des services
 - Politique de loyauté
 - Un utilisateur ne restera pas bloqué dans un service si cet utilisateur a la possibilité de poursuivre son exécution
 - Politique de simultanéité
 - Un utilisateur a la possibilité de poursuivre son exécution dès lors que les contrats d'utilisateur de ce service sont satisfaits
- Les différents concepts sont exprimés en logique temporelle
 - Pas de logique temporisée
 - Modélisation trop « grossière » pour permettre un contrôle fin de la disponibilité

■ **Modèle de Millen**

- Proche du modèle de Yu et Gligor
- Basé sur un allocateur de ressources
- Pas de logique temporelle
 - Représentation explicite du temps en logique de 1er ordre
- Politique en temps fini, borné et *probabiliste*

■ Modèle de Millen (suite)

- Concept de TCB pour la disponibilité
 - Base de protection contre le déni de service (DPB)
 - Incontournable et exhaustif
 - Incorruptible
 - La DPB doit pouvoir annuler l'accès à une ressource
 - ➔ Pour limiter le temps d'accès aux ressources

■ **Modèle de Millen (suite)**

■ Réalisation d'une tâche

- La garantie d'accéder à une ressource n'est pas suffisante pour empêcher le déni de service
- Le processus doit pouvoir conserver la ressource un minimum de temps pour réaliser une certaine tâche
- Il y a déni de service si l'accès à la ressource est annulée immédiatement après que le processus l'ait obtenu

■ Conséquence

- Besoin de modéliser la notion de réalisation de tâche

■ Modèle de Millen (suite)

- Modélisation des processus
 - A chaque processus
 - Vecteur des besoins en ressources
 - Représente le nombre d'unités de chaque ressource que le processus souhaite détenir en plus de celles qu'il a déjà
 - Vecteur des besoins en temps
 - Représente la durée pendant laquelle le processus souhaite avoir un accès exclusif à chaque ressource
 - Un processus ne peut être actif que si toutes ses demandes ont été satisfaites
- Processus bienveillant
 - Processus qui respecte son contrat d'utilisateur

■ **Modèle de Millen (suite)**

- **Modélisation d'une DPB**
 - Système d'allocation de ressources
 - Algorithme du contrôleur de ressources
 - Politique de temps d'attente
 - Contrats d'utilisateur

- **Spécification de contraintes de faisabilité**
 - Faisabilité du vecteur des besoins en ressources
 - Faisabilité du vecteur des besoins en temps

- **Expression des politiques de temps d'attente**
 - Limitée à la vérification d'un allocateur de ressources
 - Pas de gestion des priorités
 - N'explique pas comment automatiser la vérification

- École Nationale Supérieure des Télécommunications de Bretagne



Le modèle FCCS

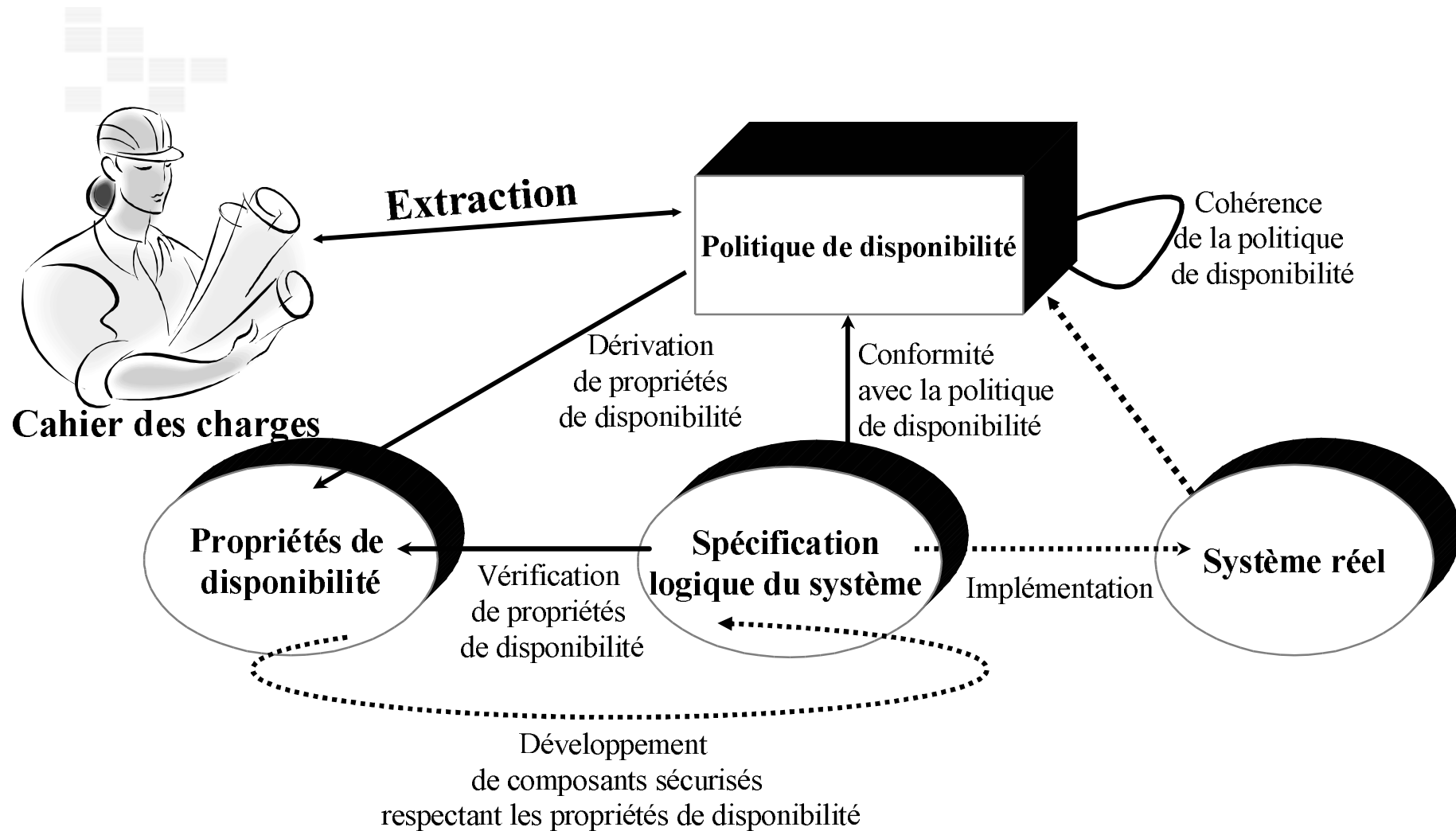
■ Base d'investigation

- Le déni de service
 - *Service **non rendu** alors qu'il **est du** au moment où il est demandé*
- La disponibilité
 - *Aptitude à **répondre à une demande** d'un service, d'une ressource en **garantissant des contraintes** d'horaires, de délai et de performances*

▪ Objectifs

- Définir un langage formel
 - Expression d'une politique de sécurité
 - Expression des propriétés de sécurité
- Effectuer des contrôles
 - L'analyse de la politique de sécurité
 - La vérification des propriétés de sécurité

Le chantier



■ Exemple

- Politique de disponibilité
 - R1 : Le sujet S doit pouvoir commencer à réaliser la tâche T au plus tard 6 unités de temps après sa requête
 - R2 : Le sujet S doit pouvoir disposer de la ressource nécessaire à T au plus tard 4 unités de temps après l'avoir demandée
 - R2' : Le sujet S doit pouvoir disposer de la ressource nécessaire à T au plus tard 9 unités de temps après l'avoir demandée
 - R4 : La durée maximale de réalisation de la tâche T est de 3 unités de temps
- Cohérence d'une politique de disponibilité
 - (R1, R2, R4) : contraintes **satisfaisables**
 - (R1, R2', R4) : incohérence
- Propriété de disponibilité
 - La requête du sujet S doit être **satisfaite** au plus tard 9 unités de temps après qu'elle ait été émise

■ Les abstractions

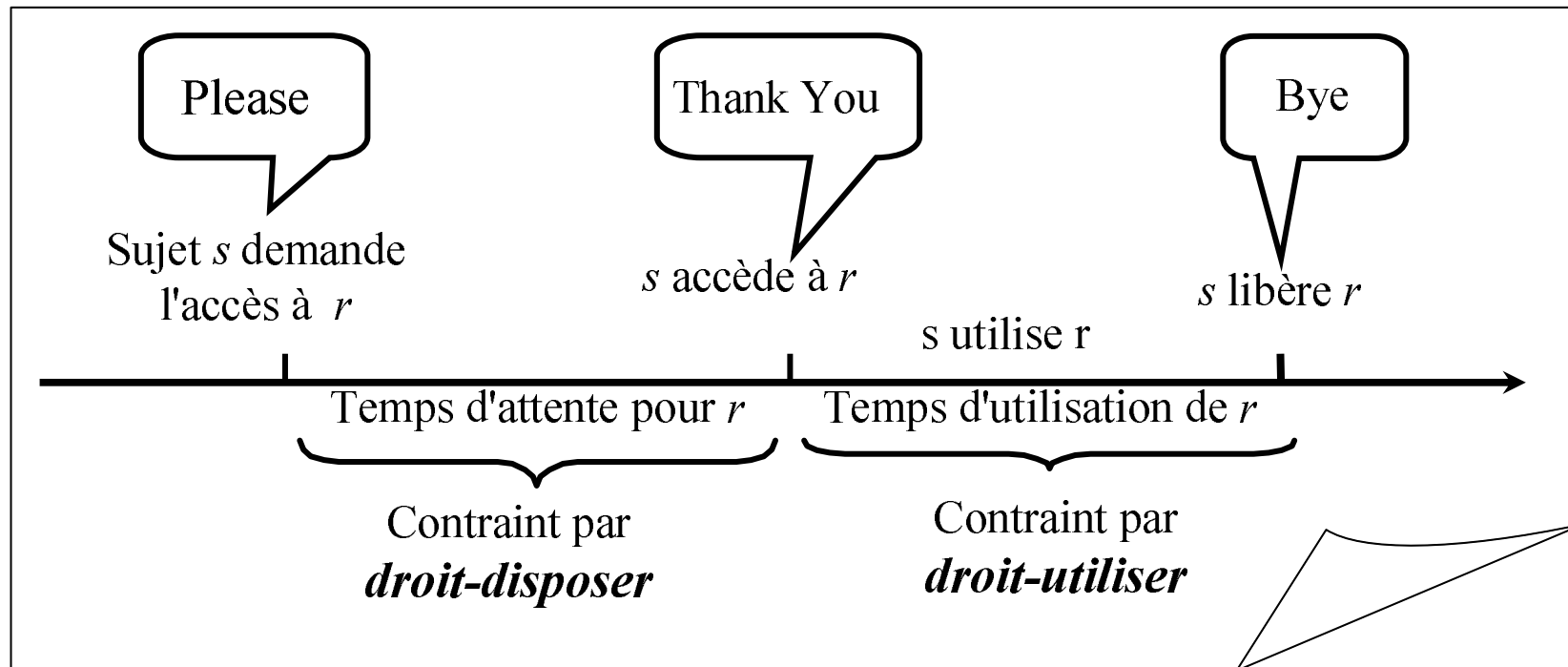
- **Ressource** : type (mémoire, CPU...), attributs (taille...), état (occupé, libre), structure,...
- **Tâche** : ressources nécessaires, durée nécessaire d'utilisation de chaque ressource, ordre d'utilisation,...
- **Sujet** : ressources consommées, état (attente, élu, en cours de réalisation...) actif ou pas (CPU)
- **Droit** : des sujets sur les ressources ou les tâches
 - *Concepts déontiques* (obligations, permissions et interdictions)
- **Notions temporelles** : événements (réalisation d'une tâche), intervalles (délais d'attente), durées, dates (points temporels)

■ **Éléments de formalisation**

- Logique du premier ordre avec égalité
- Logique temporelle (temps discret)
- Modalités déontiques
 - les privilèges
- Prédicats dédiés pour représenter
 - les concepts de base
 - ceux relatifs à la disponibilité
- Expression d'axiomes et de contraintes d'intégrité concernant
 - Les ressources
 - Les tâches

■ Spécification d'une politique de sécurité

- Elle doit exprimer des droits
 - Accès aux ressources
 - Disposition des ressources



■ Droits et contrats

- Droits associés aux tâches
 - Contraindre le temps de réalisation
 - Contraindre la durée
- Gestion des priorités
 - Entre les sujets
 - Conditions sur les tâches et les ressources
- Contrats d'utilisateurs
 - Libération de la ressource une fois utilisée
 - Interdiction de demander une ressource non utile
 - Demande d'une ressource préalablement à son utilisation

■ Cohérence d'une politique de disponibilité (Pdisp.)



- Principe
 - Lorsqu'une Pdisp. spécifie qu'un sujet a l'autorisation / l'obligation de faire quelque chose, alors elle doit lui fournir les moyens de s'y conformer
- Définition d'un ensemble de contraintes devant être vérifiées par la Pdisp.
 - Au moyen de prédicats dédiés
- Une Pdisp. est cohérente s'il est possible de montrer que ces contraintes sont des théorèmes dans cette politique

■ Conformité d'une Pdisp. avec le système

■ Principes

- Toute opération réalisée dans le système est **autorisée** dans Pdisp.
- Toute opération **obligatoire** dans Pdisp. est réalisée dans le système
- Dérivation de **contraintes de conformité** entre le système et la politique

■ Approche

- Une spécification logique du système
- Un système est conforme à une Pdisp. s'il est possible de montrer que ces contraintes de conformité sont des théorèmes de cette spécification logique

■ Vérification d'une propriété de disponibilité

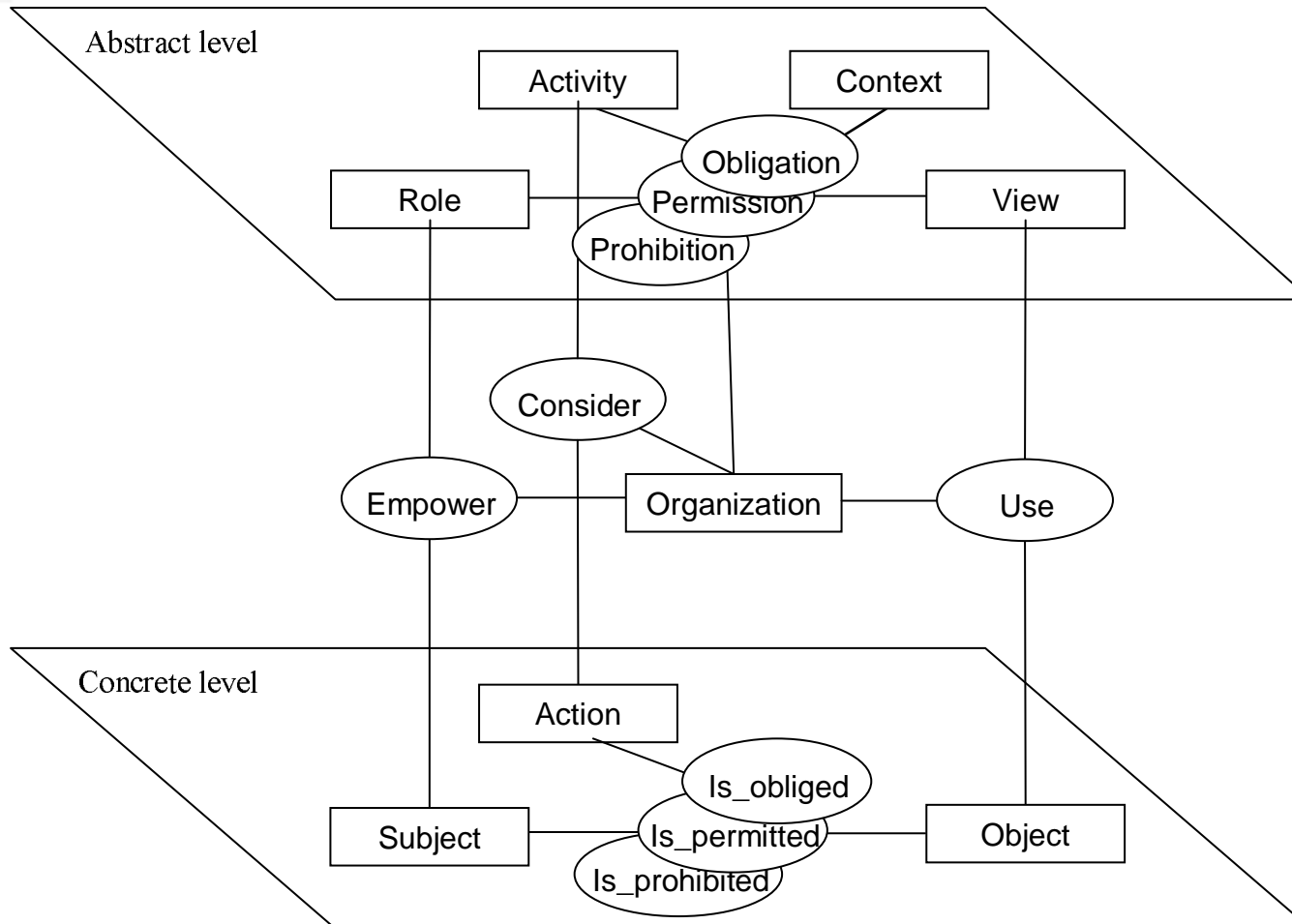
- Contraintes de contrôle de la disponibilité des ressources et de la réalisation des tâches
- Contraintes de conformité entre le système d'allocation des ressources et la politique
- Propriétés de disponibilité simple (à temps borné)
 - Toute requête sur une ressource sera satisfaite en un temps fini
 - Tout sujet demandant la réalisation d'une tâche la verra se réaliser en un temps fini
- Vérification
 - Montrer que ces contraintes sont des théorèmes de la spécification logique du système

■ **Modèle FCCS : Perspectives**

- Obtenir le service ou la ressource demandée mais altérée
 - Pb de disponibilité ?
- Obtenir le service ou la ressource demandée mais pour une durée ne permettant pas d'en disposer correctement
 - Pb de disponibilité ?
- Obtenir le service ou la ressource demandée mais avec déclassification
 - Pb de disponibilité ?
- Obtention d'un service ou d'une ressource dépendant de l'obtention d'autres services ou ressources
 - Composabilité de la disponibilité ?
- Introduction du mode dégradé (malveillance)

■ Expression d'une PDisp. dans le modèle Or-BAC

■ Or-BAC : Organization Based Access Control



■ Expression d'une PDisp. dans le modèle Or-BAC

■ Partie 1

- Permission pour un rôle de demander un service nécessitant un ensemble de ressources
 - Demander un service = activité
 - Ensemble de ressources = vue

■ Partie 2

- Obligation pour le système de satisfaire la requête en respectant les contraintes associées à la requête
- Modélisation en Or-BAC : utilisation du contexte ?

■ Cahier des charges pour une étude de cas

- Processus réalisant des services / tâches pour le compte d'utilisateurs
- Besoin d'accéder à des ressources pour réaliser les services
- La réalisation des services doit satisfaire des contraintes temps réel
- Les contraintes dépendent
 - Du rôle de l'utilisateur
 - Du caractère prioritaire de la demande
- Prise en compte des caractéristiques du système
- Spécification d'un mode dégradé
- Etudes de cas envisageables
 - ARINC 629 ?, gestion de trafic ?, base de données médicales dans un service d'urgence ?

■ Un cadre compositionnelle pour l'expression de la disponibilité

- Approche Rely-Guarantee (Assume-Guarantee) :
 - Introduite par C.Jones (81-83), pour offrir un cadre compositionnelle à la preuve de programmes parallèles
 - S'étend "naturellement" à la vérification compositionnelle de systèmes ouverts
- Une spécification Rely-Guarantee (R,G) pour un module M articule/associe la "correction" de son propre comportement, à la "correction" du comportement de l'environnement :
 - R : la condition que M suppose satisfaite par l'environnement, lorsque ce dernier interagit avec lui
 - G : dans cette hypothèse, la condition que s'engage à satisfaire M lorsqu'il interagit avec l'environnement
 - $M \models (R,G)$: si l'environnement respecte R alors M satisfait G

■ Principe de composition

- Soit M la composition de M_i ($i \in I$) :
 - $\forall i . M_i \models (R_i, G_i)$
 - $\forall j . R \wedge \bigwedge G_i \Rightarrow R_j$
 - $\bigwedge G_i \Rightarrow G$
 - **Alors** $M \models (R, G)$
- Théorème de Abadi-Lamport (92) :
 - Si tous les R_i sont des sûretés alors la circularité dans le raisonnement est évitée
- Résultat de Stark (85) :
 - Condition suffisante pour dériver compositionnellement une vivacité globale à partir de vivacités locales
- **Perspective** : exprimer une politique de disponibilité dans le format Rely-Guarantee et analyser l'application du principe de composition dans ce contexte