Conception et implémentation de systèmes sûrs et sécurisés

Julien Delange <delange@enst.fr>

(06/03/2009)

Plan

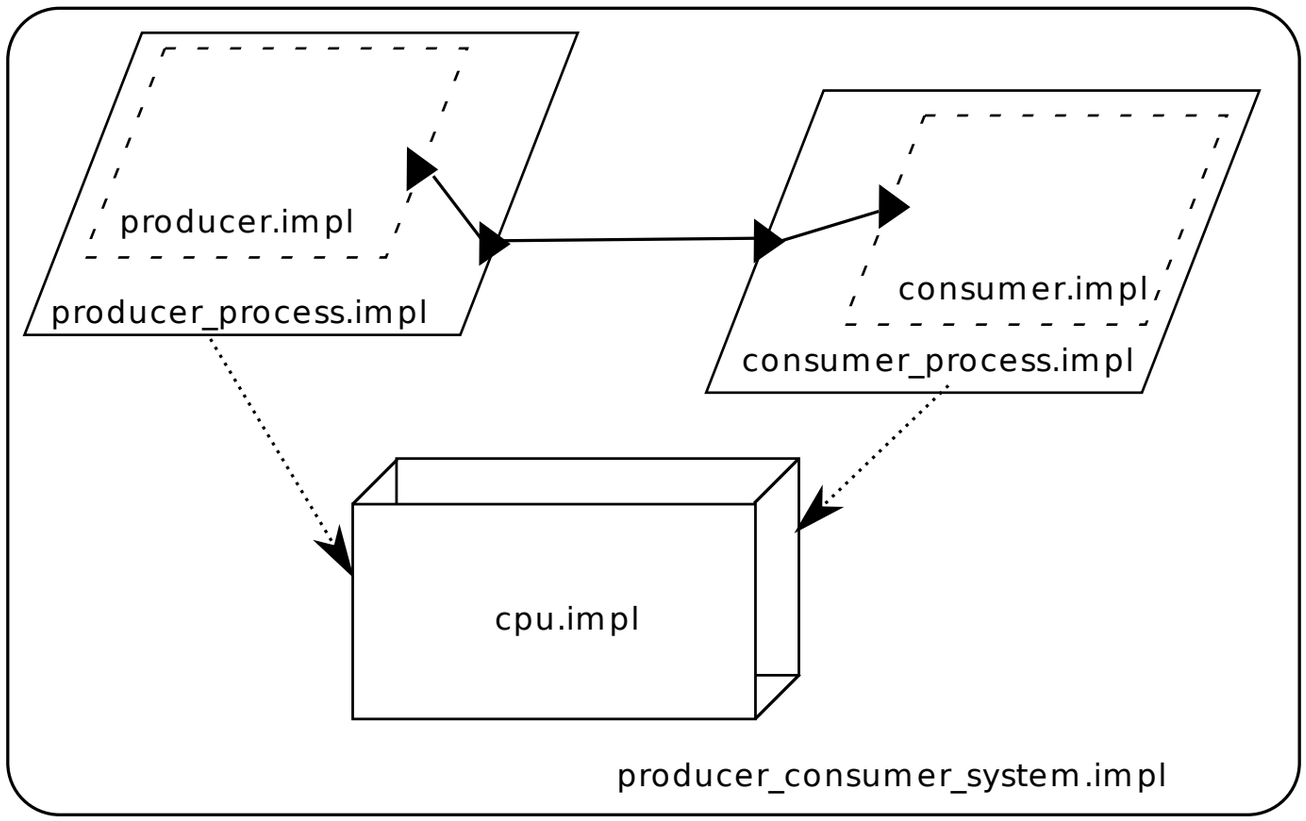
- Introduction
- Approches de sécurité et sûreté
- Application au langage AADL
- Implémentation de systèmes sûrs et sécurisés
- Conclusion

Plan

- Introduction
- Approches de sécurité et sûreté
- Application au langage AADL
- Implémentation de systèmes sûrs et sécurisés
- Conclusion

AADL

- Langage à base de composants
  - Emboîtement de composants
  - Plus haut composant : system
- Ajout de propriétés
  - Description fine du système
- Outillage complet
  - Atelier de modélisation
  - Validation de modèles (Cheddar, plug-ins, ...)



Retour vers le futur ...

- AADL, bon candidat pour conception/l'implémentation
  - cf. interventions de B. Zalila et P. Feiler
  - Projets Assert, Flex-eWare, ...
- Modélisation et validation du système
  - Description "bas-niveau" comportant toutes les propriétés du système
  - Etapes de validation (ordonnançabilité, ...)
- Génération automatique de code
  - Implémentation de l'application
  - Seule partie écrite : le code applicatif (partie dîte "fonctionnelle")

De PolyORB-HI à POK

- Faits interessants ...
  - Implémentation réellement minimale, faible complexité
  - cf. projets Flex-eWare
- ... manque des propriétés importantes
  - Assurer le fonctionnement du système (aspect sûreté)
  - Apporter des garanties sur la légitimité des échanges (aspect sécurité)
- Intégration des propriétés de sécurité/sûreté
  - Quels efforts de modélisation, validation ou implémentation ?
  - Création du projet POK

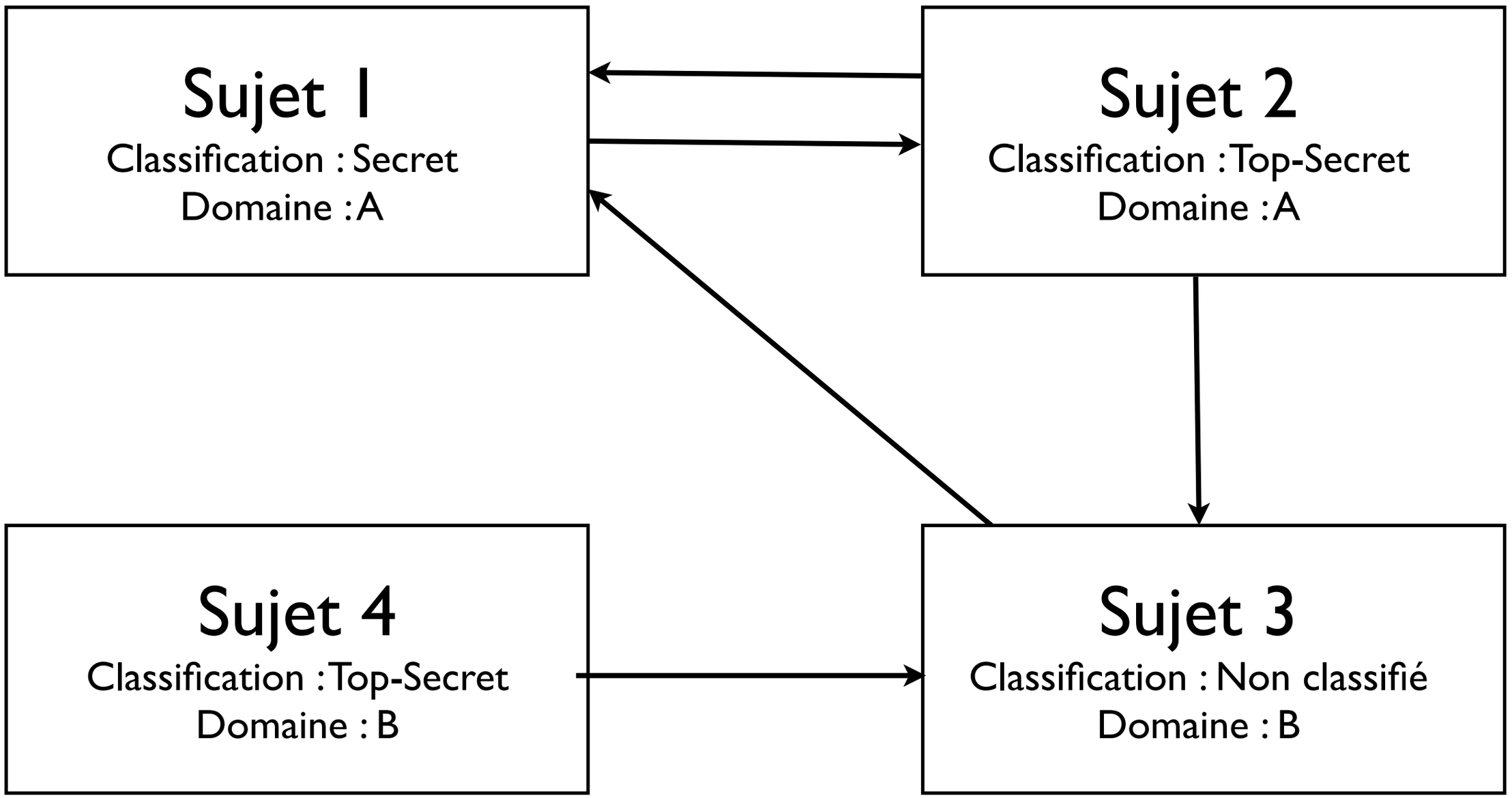
Plan

- Introduction
- Approches de sécurité et sûreté
- Application au langage AADL
- Implémentation de systèmes sûrs et sécurisés
- Conclusion

Politiques de sécurité (1)

- Que peut faire un sujet sur un objet ?
  - Objet = donnée ;
  - Sujet manipule les objets
  - Opérations = {lecture, écriture, éxécution}
- Niveaux et domaines de sécurité
  - Niveaux = {secret,top-secret,non classé,...}
  - Domaines = {militaire, civil, ...}
- Nombreuses politiques existantes
  - Bell-Lapadula : no read-up, no write-down
  - Biba : no read-down, no write-up

Politiques de sécurité (2)



Politiques de sécurité (2)

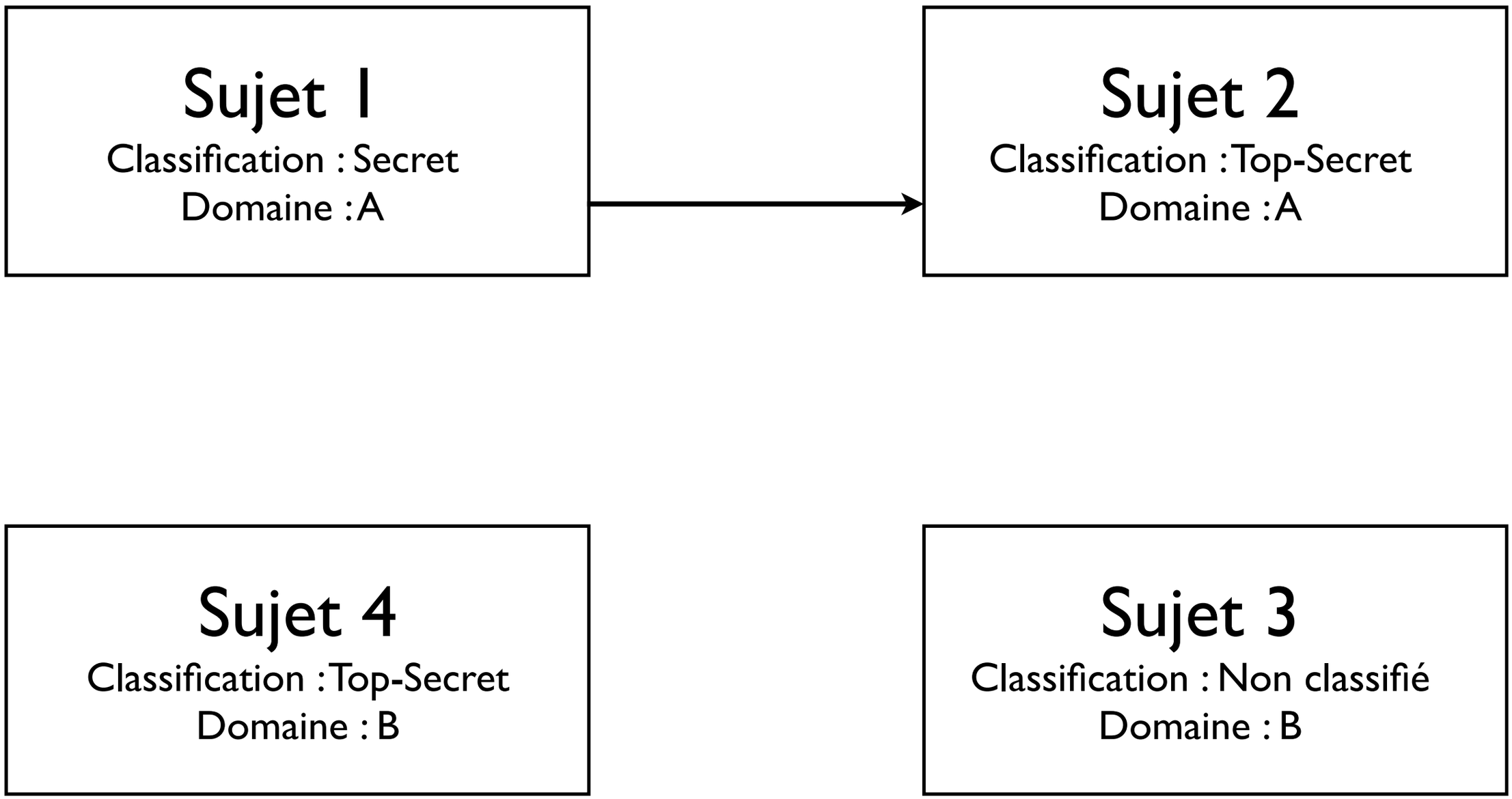
Sujet 1
Classification : Secret
Domaine : A

Sujet 2
Classification : Top-Secret
Domaine : A

Sujet 4
Classification : Top-Secret
Domaine : B

Sujet 3
Classification : Non classifié
Domaine : B

Politiques de sécurité (2)



Politiques de sécurité (2)

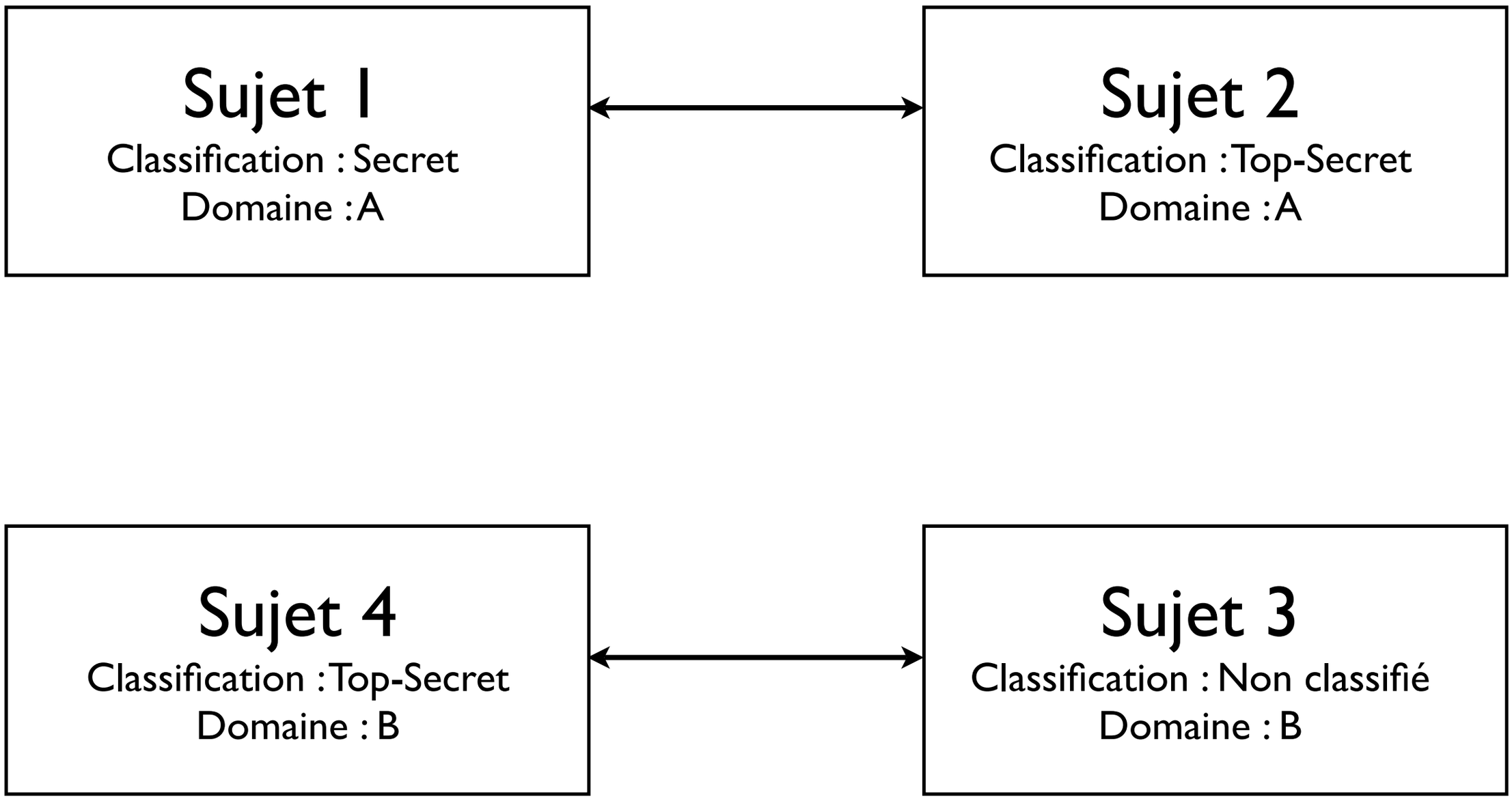
Sujet 1
Classification : Secret
Domaine : A

Sujet 2
Classification : Top-Secret
Domaine : A

Sujet 4
Classification : Top-Secret
Domaine : B

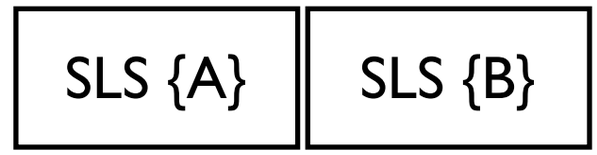
Sujet 3
Classification : Non classifié
Domaine : B

Politiques de sécurité (2)



MILS

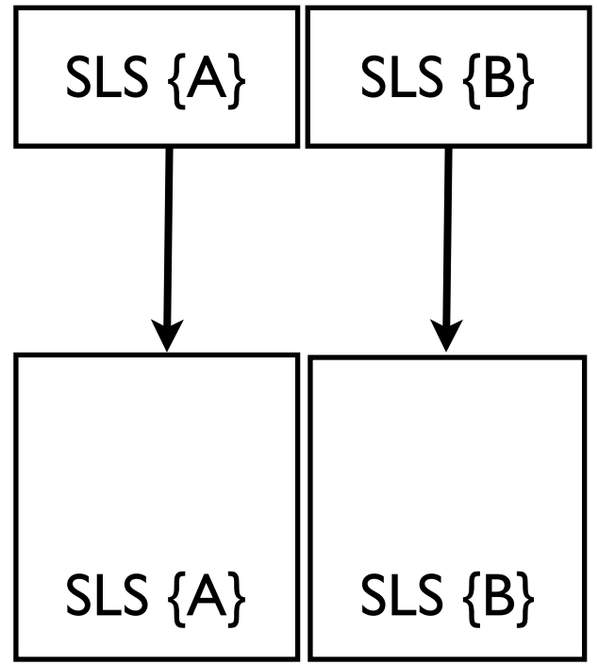
- **M**ultiple **I**ndependent **L**evel of **S**ecurity (J. Rushby)
  - Plusieurs niveaux de sécurité
  - Maintient d'une séparation entre les niveaux



- Classification des composants
  - **S**ingle **L**evel of **S**ecurity
  - **M**ultiple **S**ingle **L**evel of **S**ecurity
  - **M**ultiple **L**evel of **S**ecurity (à éviter)
- Supervision des échanges
  - Introduction d'un noyau/machine virtuelle de supervision
- Abstrait ? Oui, ce n'est pas un standard ...

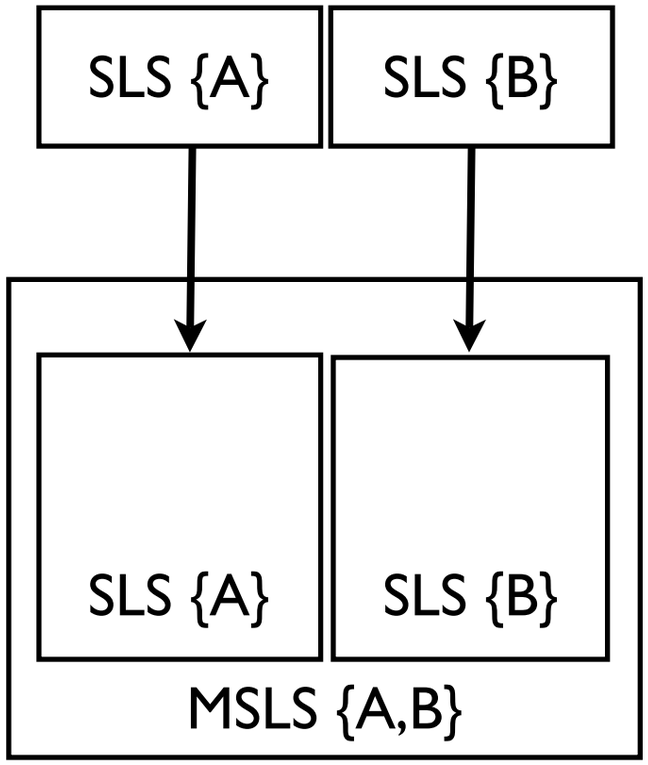
MILS

- **M**ultiple **I**ndependent **L**evel of **S**ecurity (J. Rushby)
  - Plusieurs niveaux de sécurité
  - Maintient d'une séparation entre les niveaux
- Classification des composants
  - **S**ingle **L**evel of **S**ecurity
  - **M**ultiple **S**ingle **L**evel of **S**ecurity
  - **M**ultiple **L**evel of **S**ecurity (à éviter)
- Supervision des échanges
  - Introduction d'un noyau/machine virtuelle de supervision
- Abstrait ? Oui, ce n'est pas un standard ...



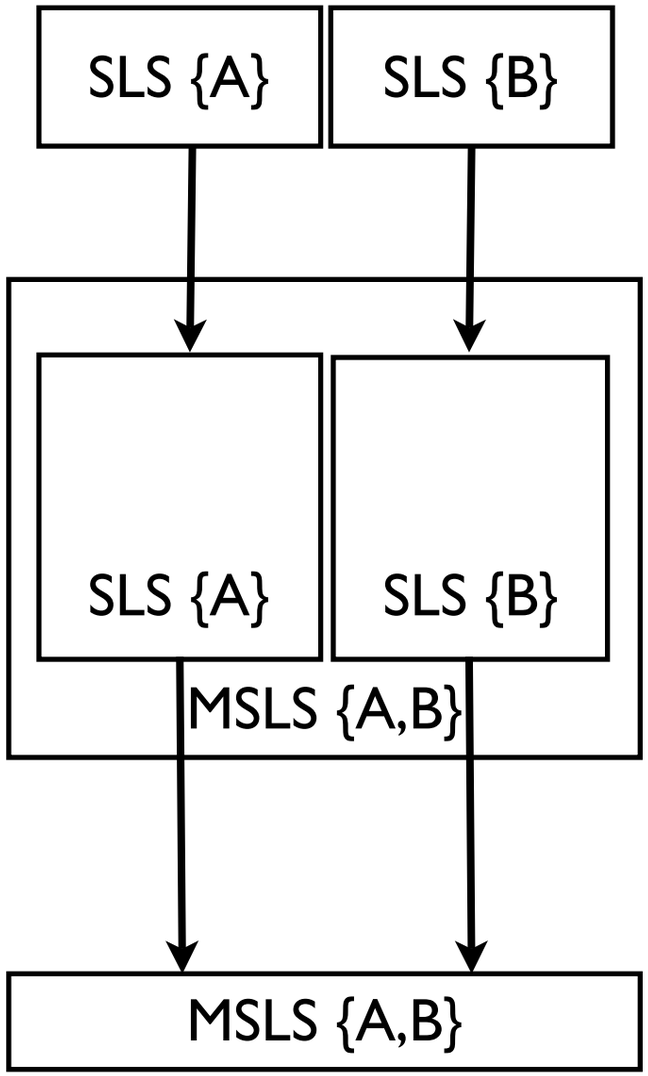
MILS

- **M**ultiple **I**ndependent **L**evel of **S**ecurity (J. Rushby)
  - Plusieurs niveaux de sécurité
  - Maintient d'une séparation entre les niveaux
- Classification des composants
  - **S**ingle **L**evel of **S**ecurity
  - **M**ultiple **S**ingle **L**evel of **S**ecurity
  - **M**ultiple **L**evel of **S**ecurity (à éviter)
- Supervision des échanges
  - Introduction d'un noyau/machine virtuelle de supervision
- Abstrait ? Oui, ce n'est pas un standard ...



MILS

- **M**ultiple **I**ndependent **L**evel of **S**ecurity (J. Rushby)
  - Plusieurs niveaux de sécurité
  - Maintient d'une séparation entre les niveaux
- Classification des composants
  - **S**ingle **L**evel of **S**ecurity
  - **M**ultiple **S**ingle **L**evel of **S**ecurity
  - **M**ultiple **L**evel of **S**ecurity (à éviter)
- Supervision des échanges
  - Introduction d'un noyau/machine virtuelle de supervision
- Abstrait ? Oui, ce n'est pas un standard ...

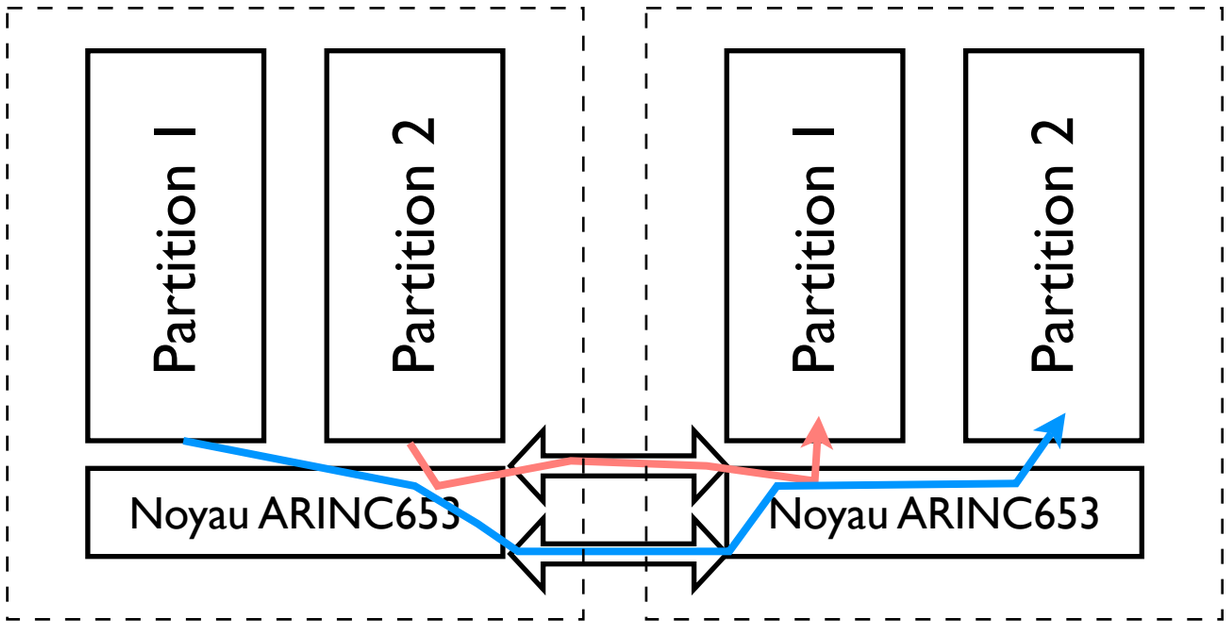


Apport de la sûreté

- Assurer le fonctionnement du système
  - Eviter toute perturbation, maximiser la disponibilité
  - Nécessaire dans de nombreux domaines (ex : DO178B)
- Identification et recouvrement des fautes
  - Quelles fautes, à quel niveau peut-on les détecter ?
  - Quelles procédures exécuter ?
- Limiter la propagation
  - Restreindre les potentiels impacts des fautes
  - Nécessite une architecture dédiée

ARINC653

- Standard avionique
  - Concerne davantage la sûreté
- Notion de partitionnement
  - Noyau/intergiciel dédié
  - Chaque partition est indépendante (ressources, runtime, ...)
- Identification et traîtement des fautes
  - Niveaux de traîtement des fautes (noyau, partition, tâche)
  - Confinement des fautes, réduction de l'impact

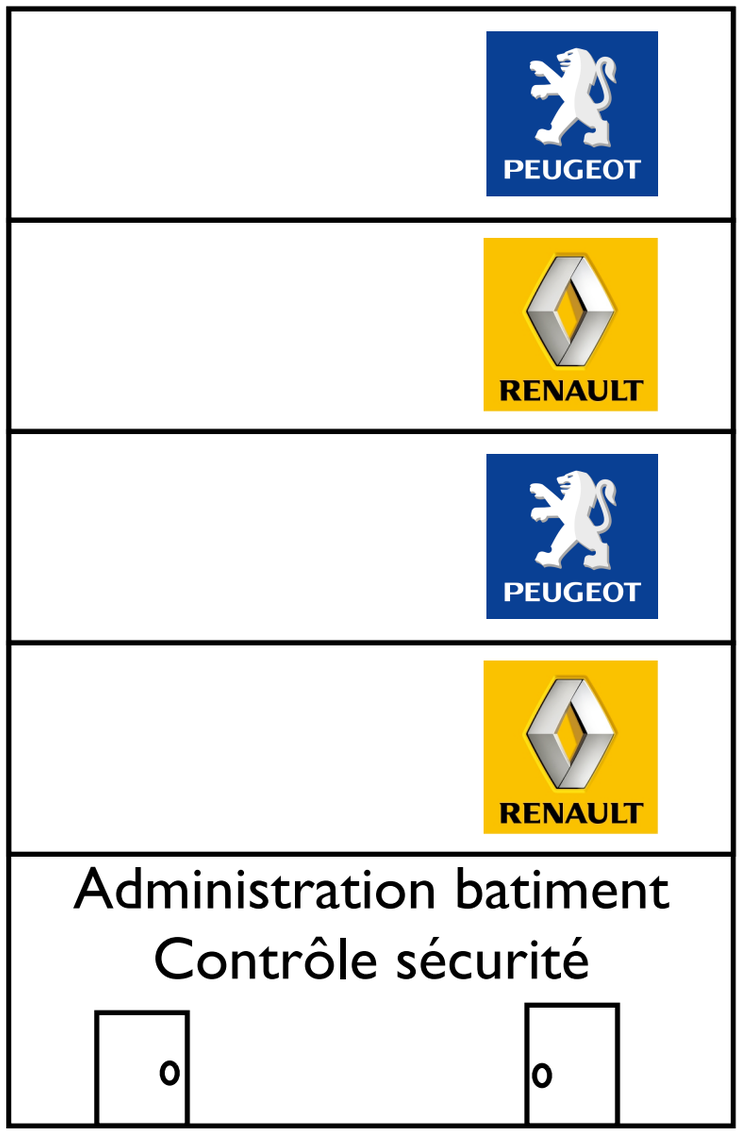


Bilan

- Notion de partitionnement
  - Entité d'exécution indépendante
  - Isolation des ressources partagée
- Politique de sécurité
  - Classification sujet/objet, autorisation de la communication
  - Supervision des échanges
- Politique de sûreté
  - Quelles fautes, quelle procédure de recouvrement ?
  - Limitation des effets causés par une faute/erreur

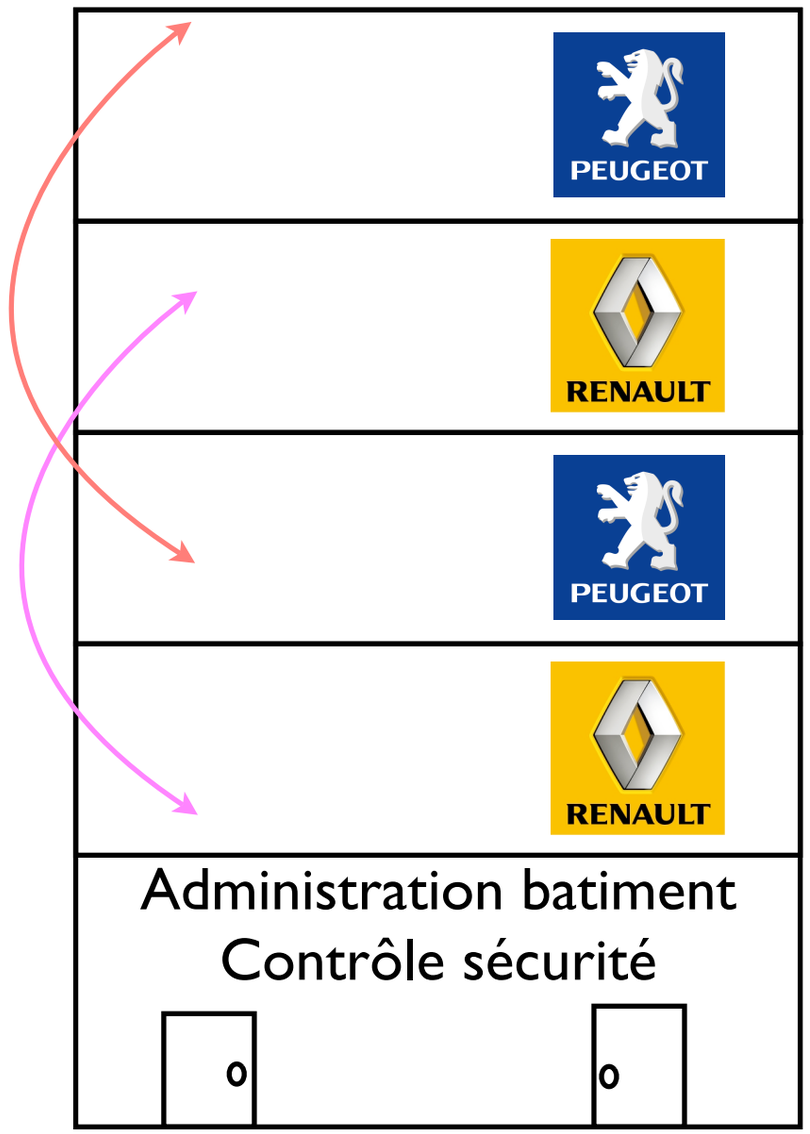
Bilan

- Notion de partitionnement
  - Entité d'exécution indépendante
  - Isolation des ressources partagée
- Politique de sécurité
  - Classification sujet/objet, autorisation de la communication
  - Supervision des échanges
- Politique de sûreté
  - Quelles fautes, quelle procédure de recouvrement ?
  - Limitation des effets causés par une faute/erreur



Bilan

- Notion de partitionnement
  - Entité d'exécution indépendante
  - Isolation des ressources partagée
- Politique de sécurité
  - Classification sujet/objet, autorisation de la communication
  - Supervision des échanges
- Politique de sûreté
  - Quelles fautes, quelle procédure de recouvrement ?
  - Limitation des effets causés par une faute/erreur



Plan

- Introduction
- Approches de sécurité et sûreté
- Application au langage AADL
- Implémentation de systèmes sûrs et sécurisés
- Conclusion

Modifications niveau modèle

- Utilisation des spécificités AADLv2
- Expression de la notion de partitionnement
  - Noyau dédié, runtime spécifique à chaque partition
  - Propriétés d'exécution (empreinte mémoire, ordonnancement)
- Spécification des propriétés de sécurité
  - Niveaux de sécurité, domaines
  - Comment spécifier ? Quels composants ?

Expression du partitionnement

- Utilisation combinée de plusieurs composants
  - Aspect runtime (`virtual processor`)
  - Aspect isolation mémoire (`process`)
  - Association 1-1 `process` et `virtual processor`
- Ajout des spécificités à chaque composant
  - Ordonnancement (`processor` et `virtual processor`)
  - Ressources (taille des threads de chaque partition)

Modélisation des niveaux de sécurité

- Modélisation de niveaux de sécurité `(virtual bus)`
  - Possibilité de réutilisation, d'héritage
- Classification des composants
  - Ajout d'une propriété pour chaque partition
  - Spécification du nombre de niveaux et de leur isolement (SLS, MSLS, MLS)
- Association composants et niveaux de sécurité
  - Composants et/ou ports

Validation sécurité

- Inspection des connections inter-partitions
  - Quels sont les niveaux de sécurité ?
  - Respectent-ils la politique de sécurité ?
- Inspection intra-partition inutile
  - Niveau de sécurité d'une partition = U niveaux de sécurité de ses tâches
- Méthode extensible
  - Vérification de plusieurs politiques de sécurité

Modélisation des aspects “sûreté”

- Modélisation des potentielles fautes
  - A chaque niveau du système (noyau, partition, tâche)
  - Fautes différentes pour chaque niveau
- Modélisation des procédures de recouvrement
  - Association d’une procédure pour chaque faute
  - Annotation du modèle (utilisation des *properties* AADL)
- Non-propagation implicite (via le *virtual processor*)

Validation sûreté

- Parcours du tryptique tâche-partition-noyau
  - Pour chaque tâche, calcul de la partition exécutée et du noyau utilisé
  - Calcul du recouvrement des fautes
- Vérification que chaque cas est traîté
  - Aucune faute ne devrait impacter le système
  - Détection des erreurs non-traîtées

Plan

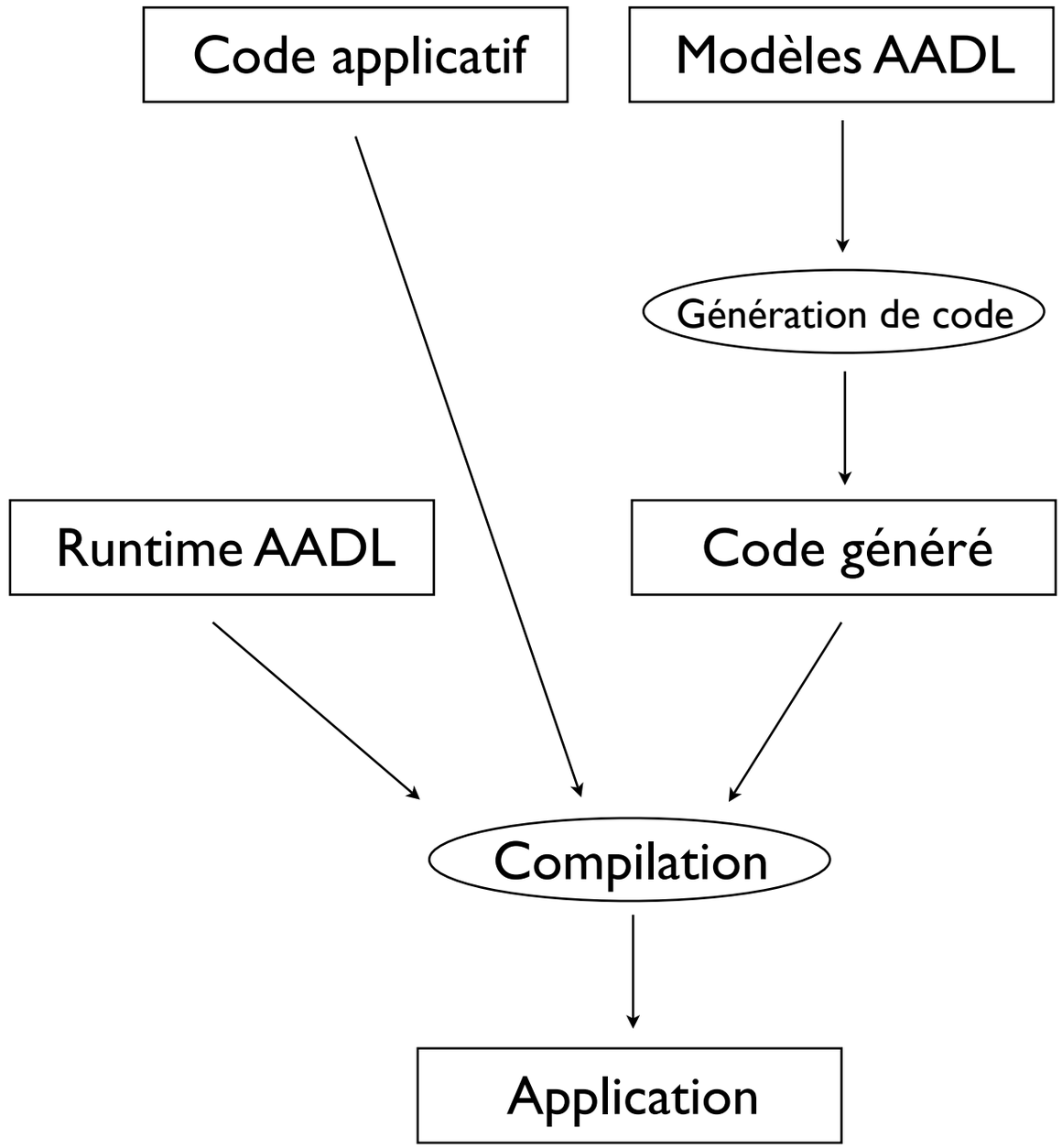
- Introduction
- Approches de sécurité et sûreté
- Application au langage AADL
- Implémentation de systèmes sûrs et sécurisés
- Conclusion

Ajout supplémentaires

- Configuration des partitions
  - Ressources, code de l'application
  - Erreurs gérées
- Configuration du noyau
  - Communication inter-partitions
  - Protocole d'ordonnancement
  - Erreurs
- Runtime dédiée
  - Gestion du partitionnement
  - Détection et reprise d'erreur

Ajout supplémentaires

- Configuration des partitions
  - Ressources, code de l'application
  - Erreurs gérées
- Configuration du noyau
  - Communication inter-partitions
  - Protocole d'ordonnancement
  - Erreurs
- Runtime dédiée
  - Gestion du partitionnement
  - Détection et reprise d'erreur



Sécurité et sûreté à l'exécution

- Aucune vérification des échanges à l'exécution
  - Le système est valide par construction
  - Supervision des échanges par le noyau

- Détection et reprise d'erreurs
  - Conforme au modèle
  - Confinement des erreurs non traîtées au runtime

Plan

- Introduction
- Approches de sécurité et sûreté
- Application au langage AADL
- Implémentation de systèmes sûrs et sécurisés
- Conclusion

Conclusion

- AADL : langage pivot d'implémentation de système
  - Spécification des propriétés non fonctionnelles
  - Implémentation "concrète" (pas de stub/skels à compléter)
- Apport de la sécurité/sûreté
  - Point de vue "bas-niveau"
  - Conformité avec les standards actuels

Questions ?

(merci de votre attention)

Note : ces travaux ont été en partie publiés dans l'annexe ARINC653 du standard AADL