Systerel

Solutions temps réel sécurisées

# Retour d'expérience sur l'usage du formel dans le ferroviaire

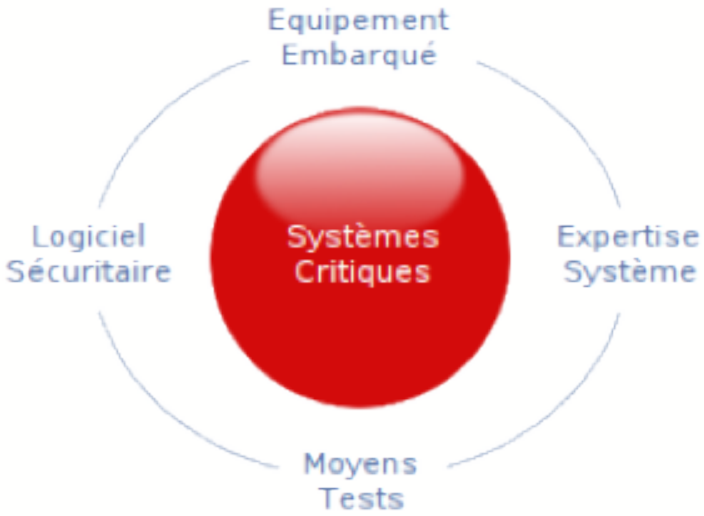
aerospace valley

SYSTEMATIC

MIPTRAIL innovation

# Carte de visite

## Spécialiste des systèmes critiques



- Expertise & Conseil
- Développement électronique & logiciel
- Méthodes formelles
- Evaluation & sûreté de fonctionnement
- Production de petites séries
- MCO

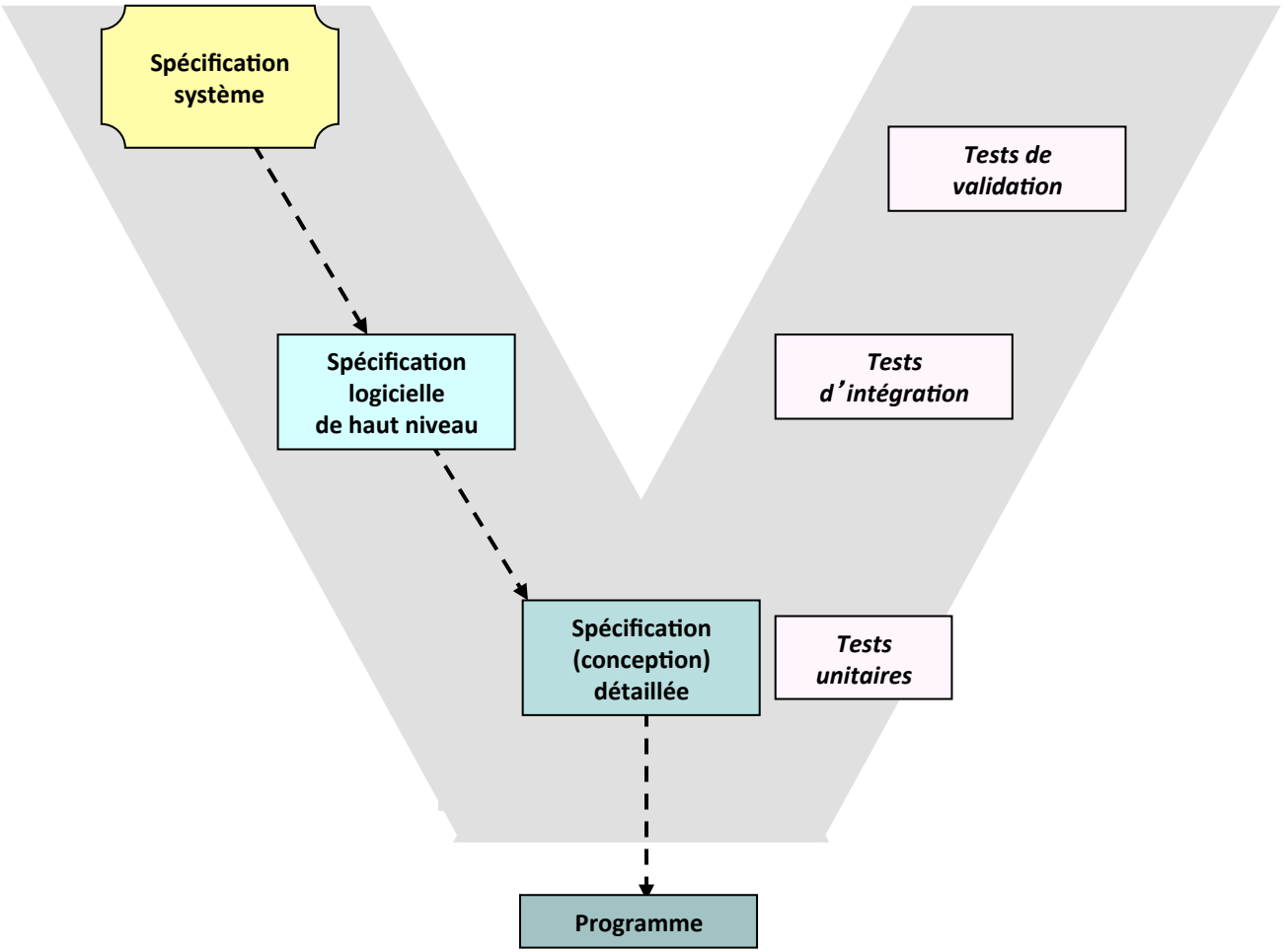
PME créée en 2002
+ 70 ingénieurs
CA : 5 M€
+ 20 % R&D

# Centre d'expertise en techniques formelles

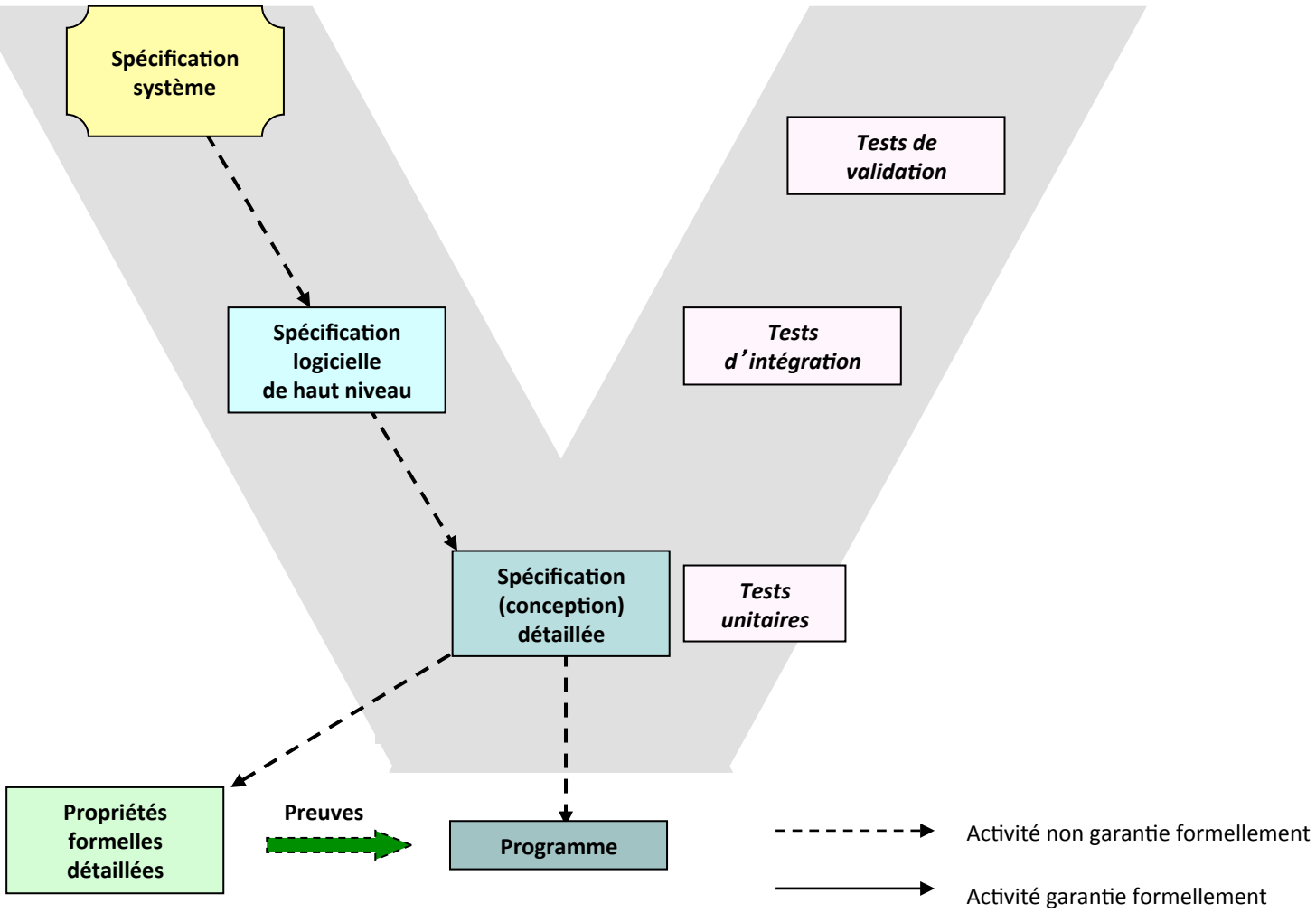
- Equipe formelle pluridisciplinaire
  - > 20 ingénieurs et experts
  - > 100 h.ans d'expérience cumulée dans la mise en œuvre industrielle de techniques formelles
  - 15 ans d'activité formelle continue sur des projets industriels et le développement d'outils

- Expertise sur des technologies formelles très variées
  - Event B et la plateforme Rodin
  - Simulink + Design Verifier
  - Méthode B et l'Atelier B
  - SCADE + Design Verifier
  - Frama-C et Polyspace
  - Outils de Prover Technology
  - GATeL
  - Différents Model Checkers, SMT and SAT Solvers, provers (COQ, ISABELLE…)
  - OVADO, PRoB… (vérification formelle de données de configuration)

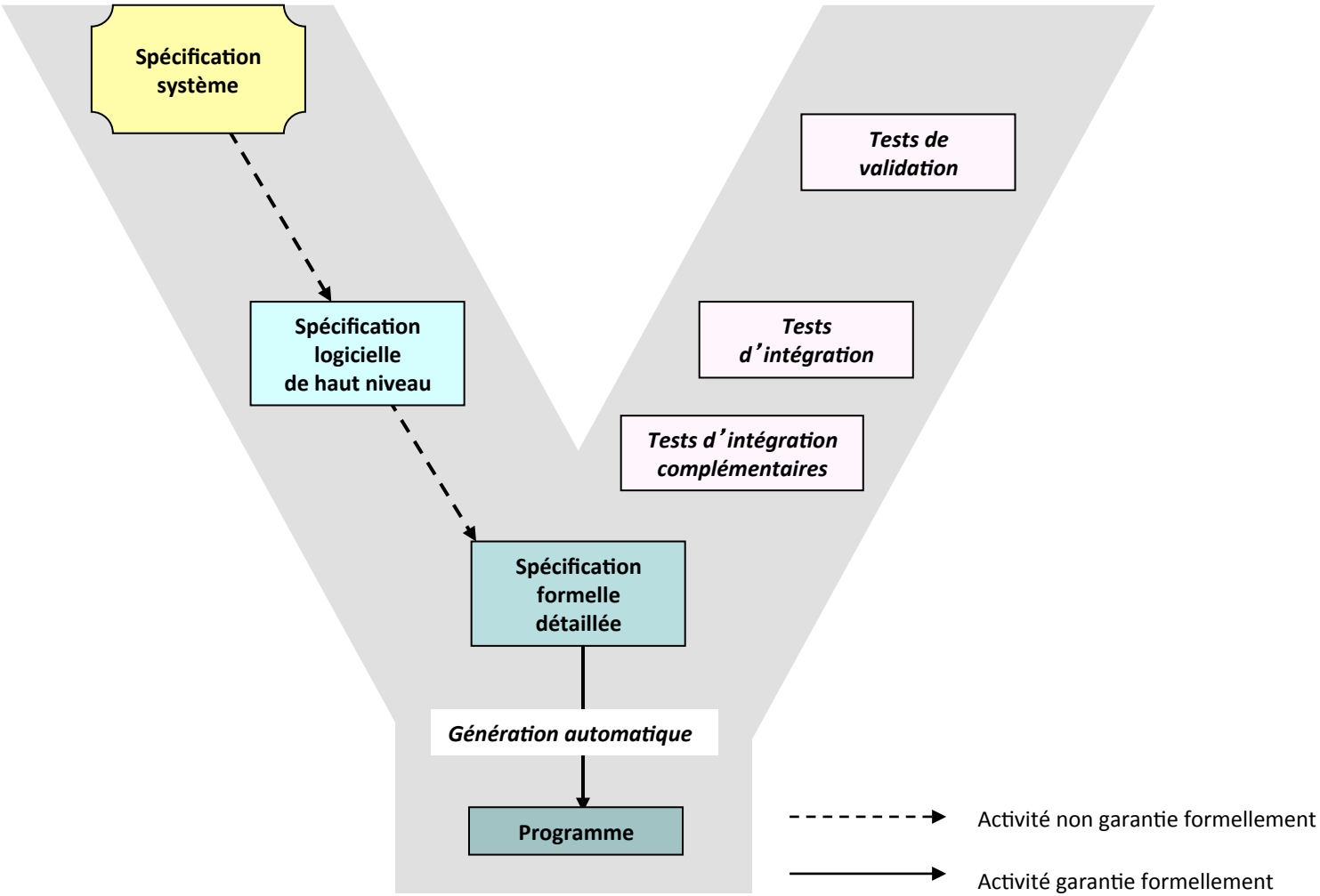
# Apport du « formel » au cycle en V



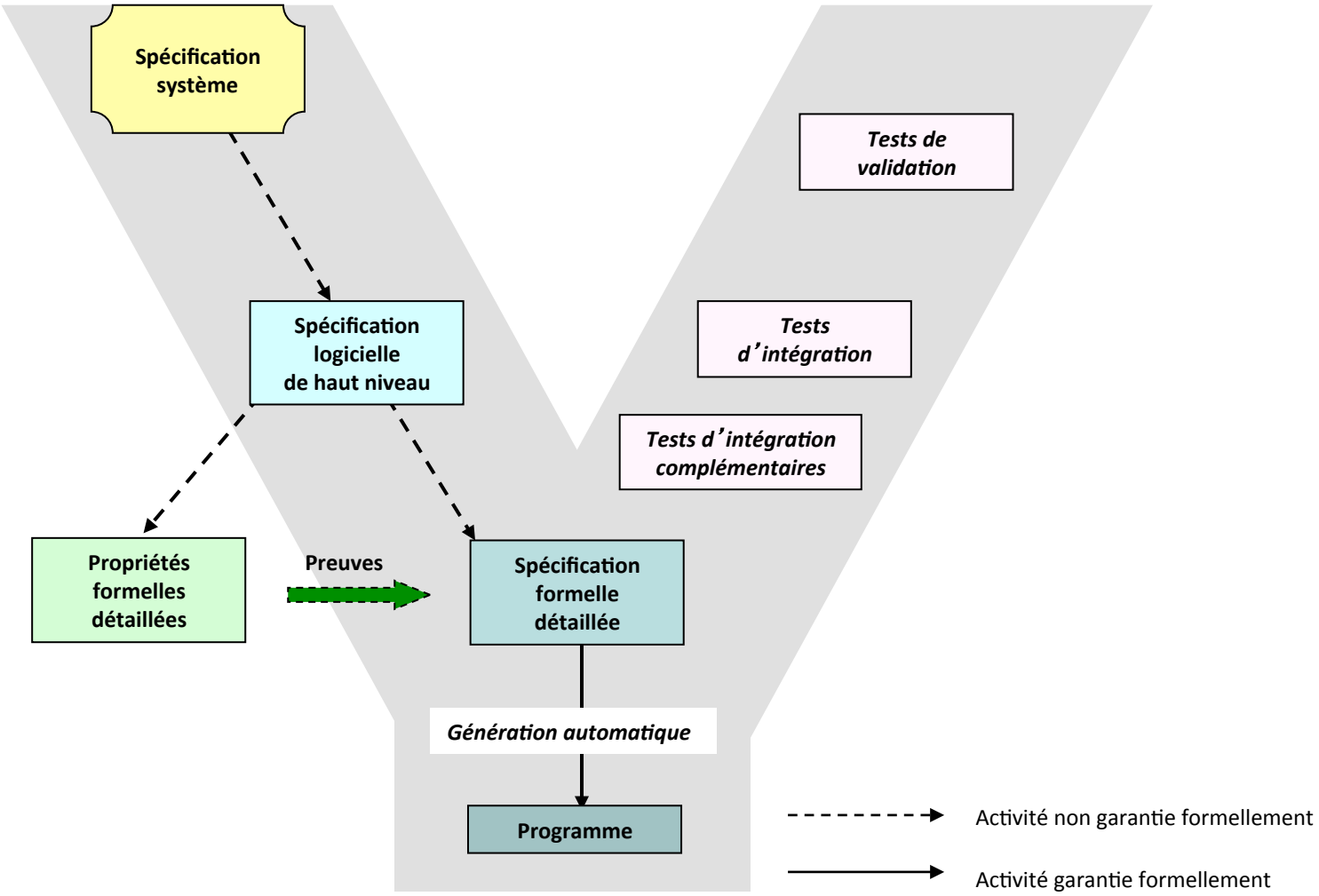
# Apport du « formel » au cycle en V



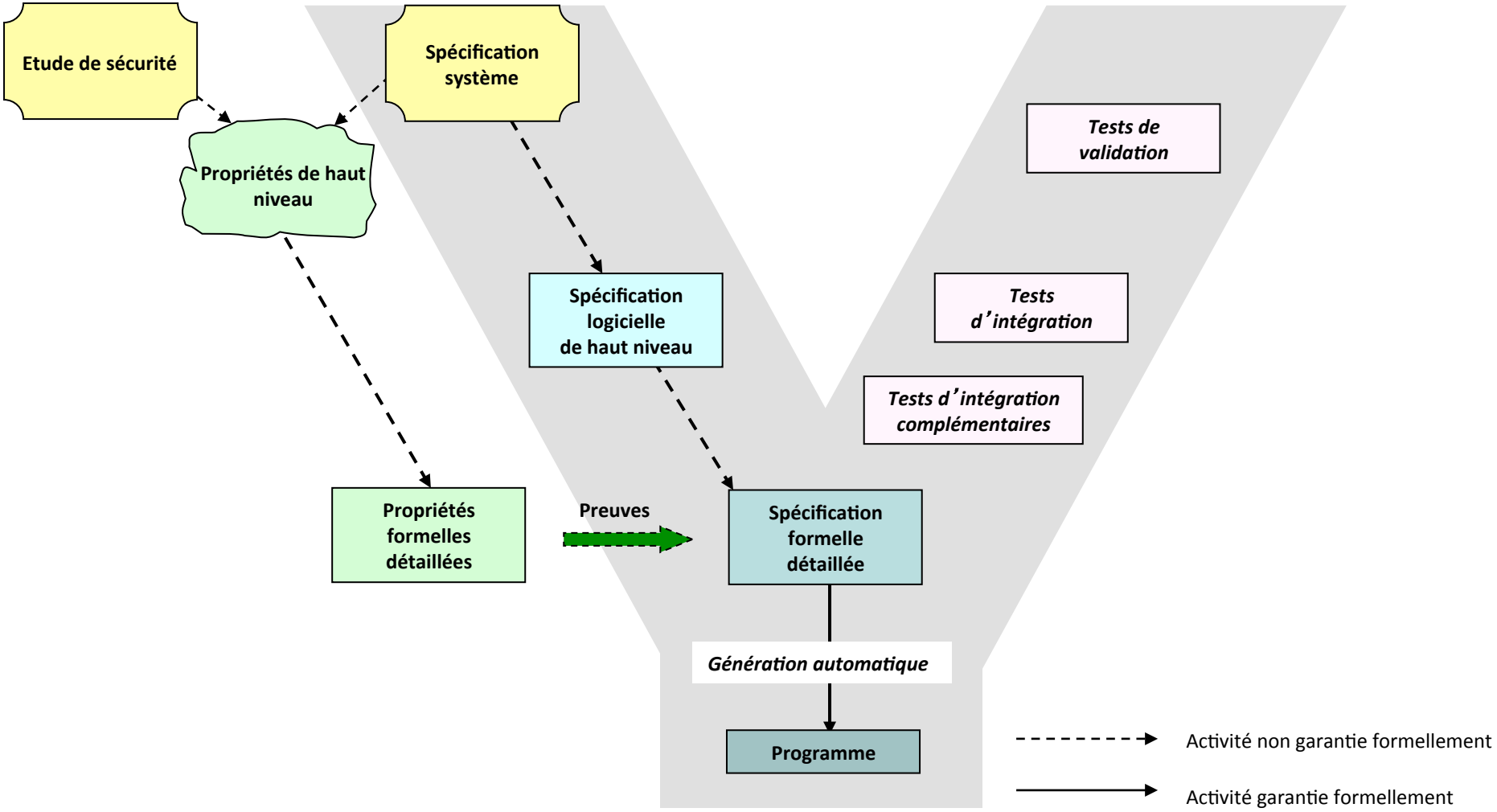
# Apport du « formel » au cycle en V



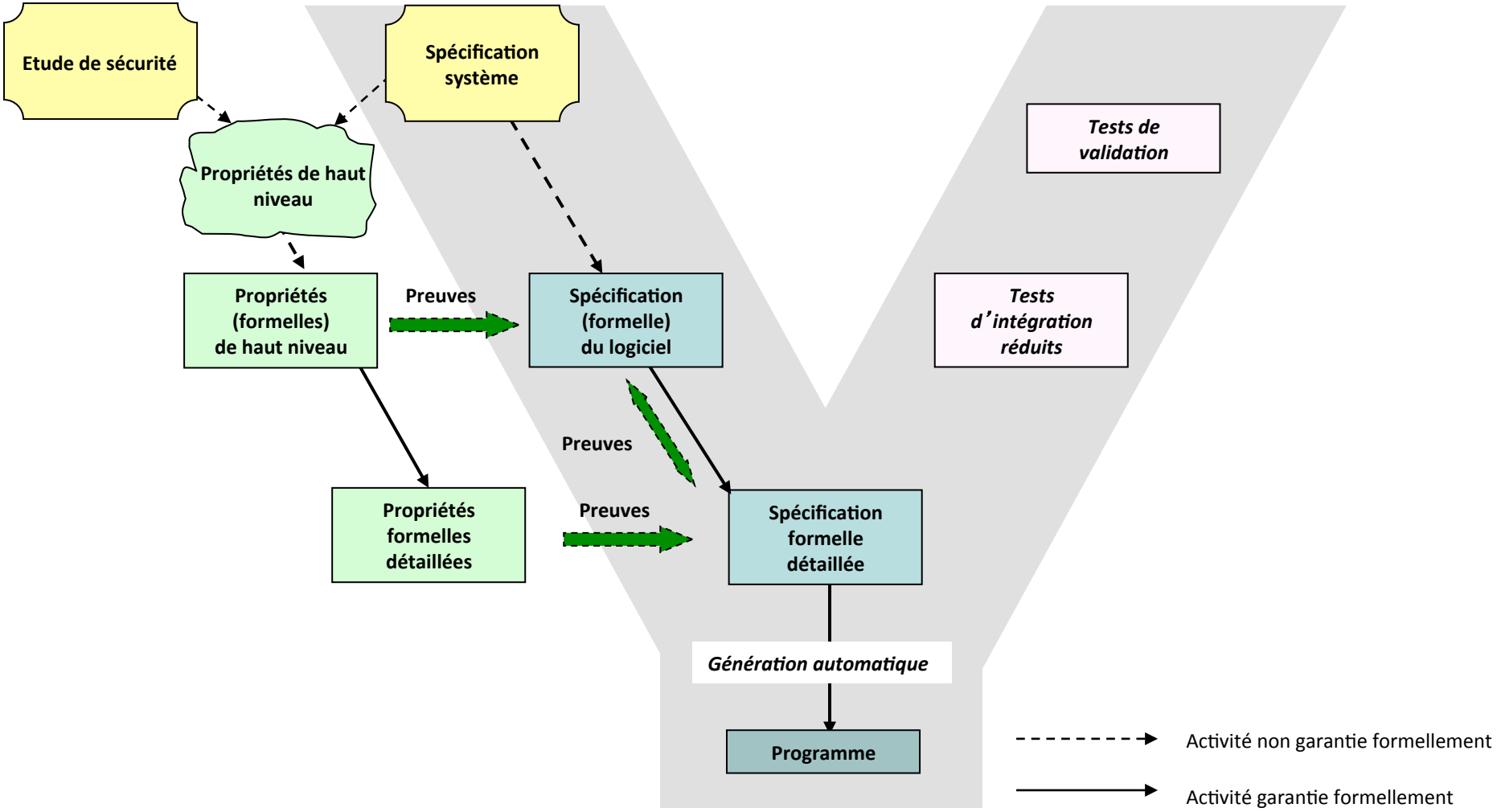
# Apport du « formel » au cycle en V



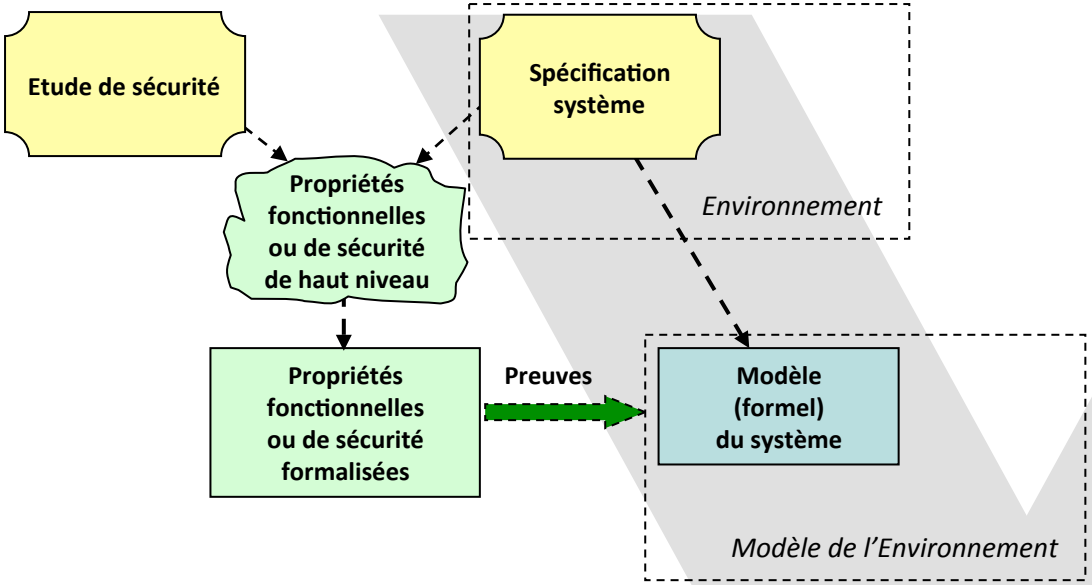
# Apport du « formel » au cycle en V



# Apport du « formel » au cycle en V



# Apport du « formel » au cycle en V




Valider des exigences fonctionnelles

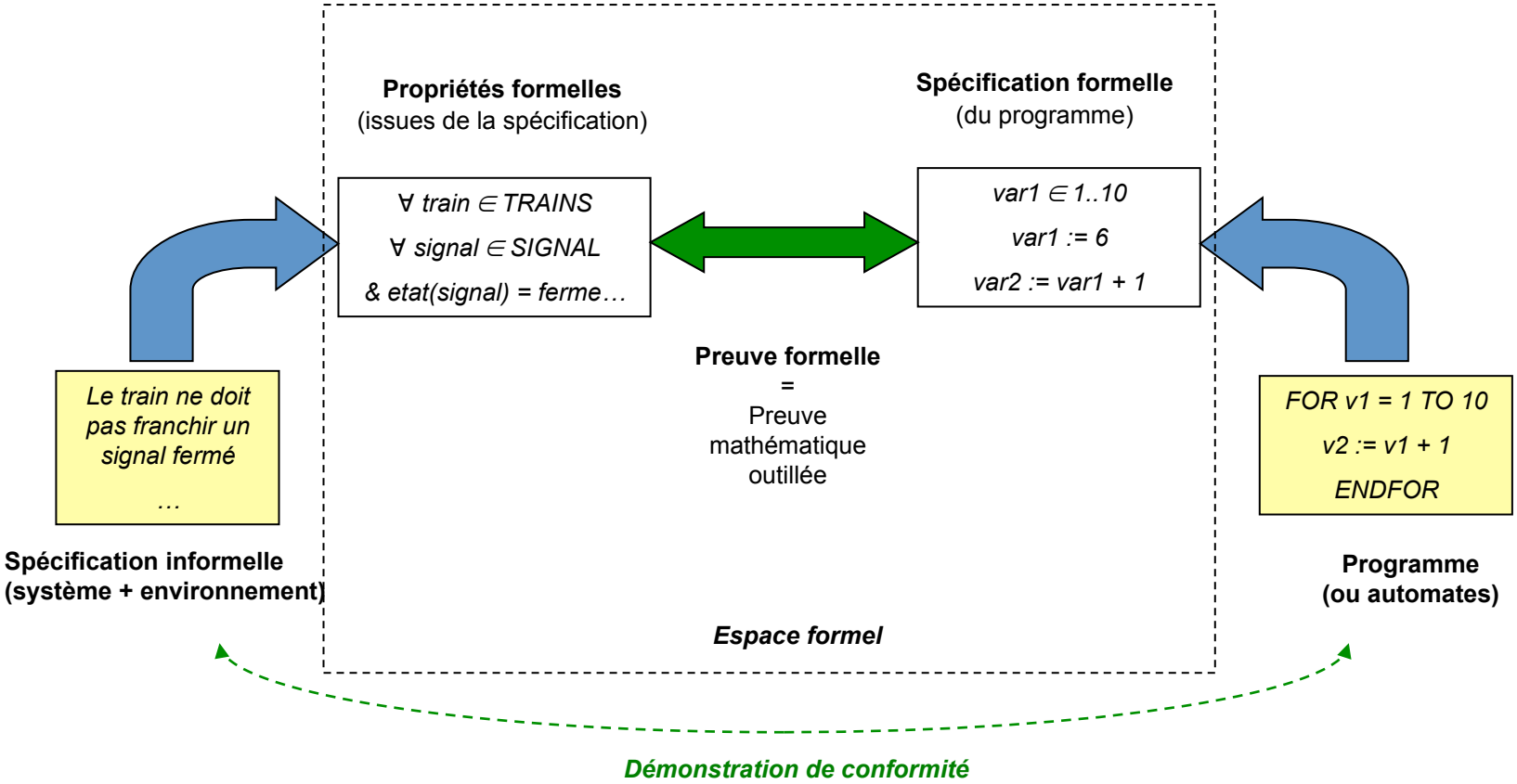
Prouver des exigences de sécurité

Exhiber par contre-exemple ou preuve indémontrable une situation contraire à un attendu

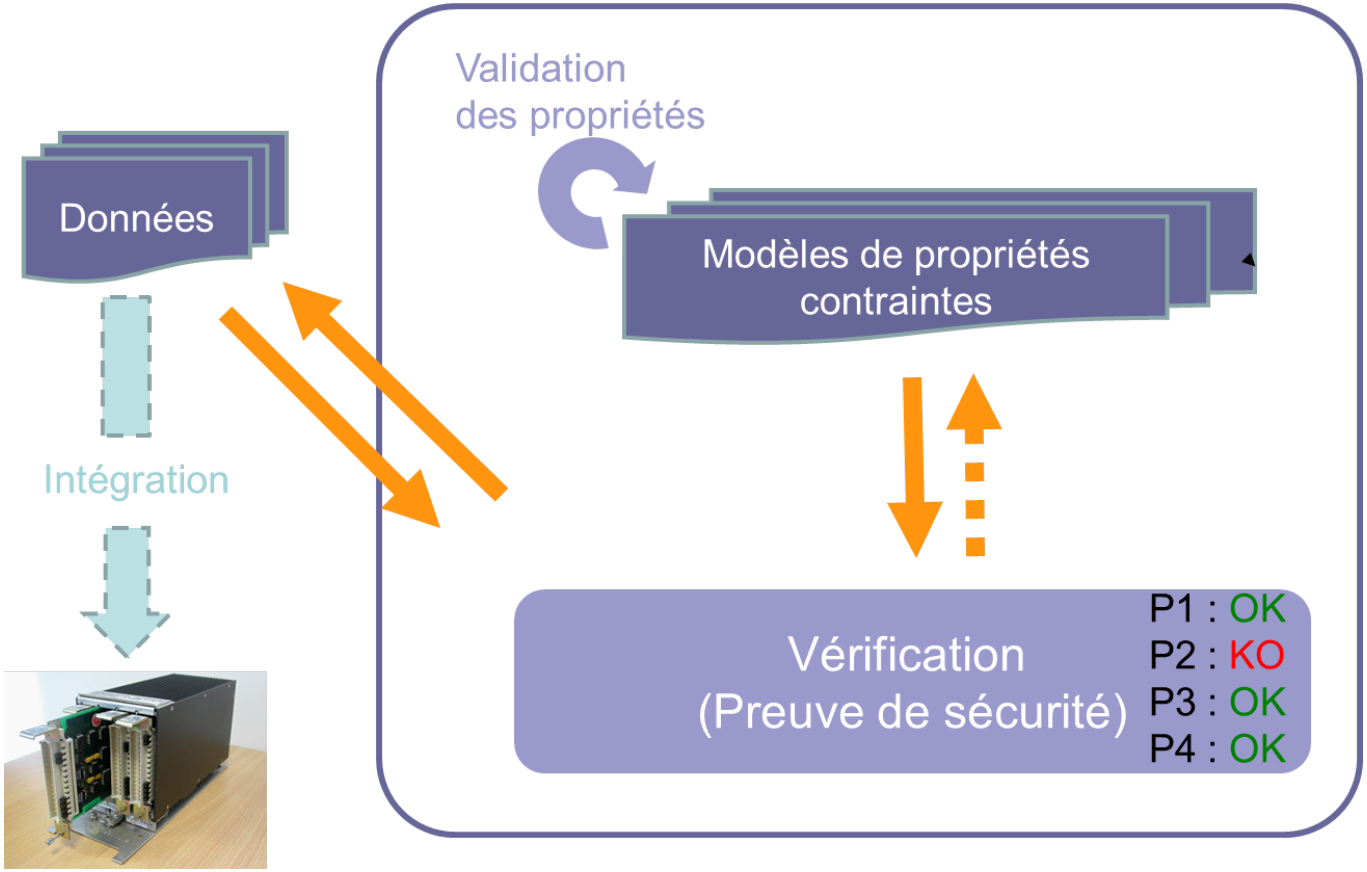
Activité non garantie formellement

Activité garantie formellement

# Apport du « formel » au cycle en V



# Apport du « formel » au cycle en V




Vérification formelle des données (OVADO)

# Quelles techniques pour quelle problématique ?

- A chaque problème, une solution :

Raisonner sur le système

Event B + Rodin

Simulink + Design Verifier

Etablir une spécification formelle

Méthode B + Atelier B

SCADE + Design Verifier

KCG ou traducteurs B/C ou Ada

Produire un logiciel « correct par construction »

Vérifier certaines propriétés du code

Polyspace, (Frama C)

Vérifier a posteriori certaines propriétés du logiciel sur le code

Prover tools

Vérifier a posteriori une spécification logicielle

Vérifier a posteriori une spécification système

Event B + Rodin

Prover tools

ProB+Rodin

Générer les données critiques de configuration

Vérifier la correction des données critiques de configuration

OVADO

ProB+Rodin

# Essor du formel dans le ferroviaire

| 1990 | | 2000 | 2010 | |
|---|---|---|---|---|
| | | | | GE Transportation |
| | | | THALES<br>AREVA-TA | |
| | ALSTOM<br>SIEMENS<br>ANSALDO<br>INRETS | BOMBARDIER<br>INVENSYS | | |
| RATP<br>SNCF | | | | Event B<br>Méthode B<br>SCADE + Prover Certifier<br>Simulink + Design Verifier<br>Prover<br>OVADO<br>ProB |
| Spark<br>Langage Z<br>Lustre | Méthode B<br>SCADE | Méthode B<br>SCADE<br>Prover | | |

ATTENTION : Dates indicatives / liste des acteurs non exhaustive

# Pourquoi un tel essor ? (1/2)

- Une réelle adéquation aux besoins

  Réduire les ambigüités des spécifications

  Pouvoir raisonner sur ces spécifications, idéalement en prouver certaines propriétés

  Disposer des moyens de produire un code conforme à ces spécifications

  Vérifier sur le code l'existence de certaines propriétés

  …

  malgré certaines limitations, insuffisantes pour ne pas retenir une approche formelle

- Une volonté continue des grands donneurs d'ordre

  Une grande constance de la RATP depuis le début des années 1990

  Une implication de la SNCF et de l'INRETS pour certaines technologies

  Des moyens financiers conséquents (à mettre en regard des gains)

# Pourquoi un tel essor ? (2/2)

- Une aptitude des industriels à « jouer » le jeu
  - Acceptation progressive des technologies mises en avant
  - Investissement industriel (financier, formation des équipes, adaptation des processus…)
  - Application progressive aux différents logiciels de C/C ou équipements de signalisation
- Emergence d'une offre adaptée de formation, services et outils
  - Offre d'outils en progrès constant (Atelier B, SCADE, Prover, Rodin, OVADO…)
  - Sociétés à même d'accompagner les industriels dans la mise en œuvre de ces technologies
  - Formations universitaires spécialisées
  - Projets de recherche (Rodin, Deploy, Advance…)
  - Participation active des inventeurs et promoteurs des technologies (Abrial, Caspi…)
- Gain en productivité et parts de marché
  - + grande maîtrise de systèmes à la complexité croissante
  - Gain sur le coût global de vie d'un produit
  - Impact marketing lié à l'utilisation de techniques formelles
  - Certains clients imposent le recours à ces technologies

# REX sur Event B (B système)

- > 5 ans de mise en œuvre « industrielle »

  Projets de C/C de métros, PMI, portes palières… (DIR41, L13, New-York, DOF1, COPPILOT…)

- Aide au raisonnement formel sur une spécification de haut niveau

  Repose sur Event-B, évolution apportée au langage B pour pouvoir raisonner sur des systèmes

  Modélisation du système et son environnement sous forme d'événements conditionnés par des « gardes »

  Un outil libre, RODIN, plateforme sous Eclipse entièrement dédiée à Event B (www.event-b.org)

  → *Flyer à disposition*

- Apports et premier bilan de la technologie

  Aptitude à maîtriser par l'abstraction la complexité d'une spécification

  Introduction progressive (raffinement) des concepts de sécurité guidée par la preuve

  Explicitation de la logique sécuritaire d'un système : principes et hypothèses ( /!\ )

  Mise en avant des éléments manquants à une preuve de sécurité

  → Technologie à fort potentiel mais nécessite apprentissage pour exploiter la preuve

# REX sur Simulink + Design Verifier (1/2)

- > 4 ans de mise en œuvre « industrielle »

  Vérification de propriétés comportementales de C/C (métro, PMI) modélisés en Simulink

- Design Verifier

  Moteur de preuve basé sur le « plug-in » de Prover Technology

  Plusieurs stratégies de preuves: preuve et/ou recherche de contre-exemples

- Quelques exemples de mise en oeuvre

  Définition de propriétés pour valider les exigences fonctionnelles d'un Interlocking (contrôle des aiguillages et des feux de signalisation)

  Définition de propriétés pour valider la génération d'un modèle Interlocking conformément aux règles d'instanciation et au plan de voie

  Définition de propriétés pour prouver la sécurité d'un Interlocking (non occurrence d'événements redoutés du type déraillement ou collision, ...)

# REX sur Simulink + Design Verifier (2/2)

- ## Points forts de Matlab/Simulink

  Nombreuses fonctionnalités et possibilités de modélisation (Schéma Bloc / Machine à états / Scripts en langage Matlab)

  Langage très performant pour automatiser les simulations, les calculs et les traitements sur les modèles (gain en productivité)

- ## Points forts de Design Verifier

  Utilisation relativement simple et documentation claire

  Génération automatique des rapports de preuves et des contre-exemples

- ## Limitations et inconvénients

  Incapacité à traiter la vérification de certains systèmes dès que la combinatoire augmente

  Temps de traduction et d'analyse des modèles pouvant devenir très long (interminable…)

  Besoin d'optimiser ou simplifier les modèles (découpage en sous-modèles de taille raisonnable)

  Besoin de définir des lemmes pour accélérer la convergence des preuves

  Consommation en mémoire très élevée sur des modèles complexes

  Coût total des licences

# REX sur la méthode B

- 20 ans de mise en œuvre « industrielle »

  Très nombreux projets de C/C de métros (Meteor, Mexico, Singapour, Pékin, RATP L1, L3, L5, Lyon…)

  Des développements pour les grandes lignes (KVB, STM-KVB, codeur Eurobalise, …)

  Des expérimentations sur des postes à aiguillages informatisés

- Bilan synthétique

  - Points positifs :

    Technologie mature (méthode, outils, intégration à la norme 50128…)

    Efficacité démontrée pour le développement de C/C du niveau de complexité d'un CBTC

    Seule méthode à couvrir le cycle de vie de manière consistante de la spécification du logiciel à la production du code (techniques de raffinement et preuve intrinsèquement liées)

    Seule méthode à pouvoir initialement dissocier, puis par la suite verrouiller, le Quoi et le Comment

    Coût des outils attractif comparé à celui des autres technologies

# REX sur la méthode B (suite…)

- Bilan synthétique

➢ Points négatifs :

Savoir faire nécessaire pour aborder une modélisation et pousser la preuve à son terme

Théorie liant étroitement la modélisation et la preuve (le défaut de ses qualités)

→ Les modélisateurs doivent maîtriser les 2 aspects

→ Peu d'ouverture aux compromis

*« A quelques exceptions près, jusqu'au début des années 2000, il était parfois nécessaire d'avoir recours à des compromis sur la preuve pour boucler une modélisation B. »*

Peu adaptée à la reprise d'un existant non formel

Peu adaptée à la preuve a posteriori d'une spécification

→ En B, on prouve difficilement a posteriori des logiciels qui n'ont pas été conçus pour l'être

Moyennement adaptée à une spécification logicielle mouvante

Evaluation de l'effort de développement de nouveaux types de C/C

Gain économique lié à la mise en œuvre de B (comme bcp d'autres techniques formelles) difficile à quantifier

# REX sur la méthode B (fin)

- Autres raisons de retenir la méthode B

  Potentiel d'abstraction de B pour maîtriser la complexité d'un C/C

  Ensemble de mécanismes outillés pour passer de la spécification au code avec la preuve que le contenu de cette spécification n'est jamais remis en cause

  Capacité à prouver l'implantation algorithmique d'une spécification

  Architecture du logiciel « saine » car contrainte par la preuve (atout pour le client)

  Maîtrise du code produit (performance, lisibilité, comportement, debug, optimisation garantie par la preuve…)

  Analyse d'impact (rejeu de la preuve) d'une modification de spécification ou de code

  Amélioration continue du processus de modélisation

  Amélioration continue des performances de la preuve automatique

  Amélioration continue des design pattern utilisés dans les modèles abstraits (meilleur taux de preuve automatique, raffinements mécanisables…)

  Modélisation, raffinement et preuve font un tout cohérent et outillé

# REX sur SCADE (1/2)

- 5 ans de mise en œuvre « industrielle »

  Projets de C/C de métros (Octys pour RATP L5, Bord de Lyon)

- Conditions de mises en oeuvre

  Vérification d'une spécification de poste à enclenchement

  Vérification partielle de spécification système ou sous-systèmes CBTC

- Bilan synthétique

  - Points positifs :
    - Apprentissage rapide du langage et des outils
    - Lisibilité des modèles (pour un automaticien)
    - Gain de productivité (génération automatique de code, suppression de tests unitaires mais renfort des tests d'intégration), à mettre en regard du coût de possession
    - Possibilité de coupler SCADE à un outil de preuve
    - Beaucoup de REX sur mise en œuvre (Ansaldo STS, AREVA-TA, THALES, Aéronautique…)

# REX sur SCADE (2/2)

## Points négatifs :

Lisibilité d'un modèle SCADE par rapport à 100 lignes de code (vision de l'informaticien contre celle de l'automaticien)

- Séquencement des planches difficiles à lire (rapidement 10 à 20 planches pour une fonction spécifiée, interdépendance des planches)
- Rationalisation des découpes en planche / Réutilisation

Code généré lourd et peu performant dans certaines situations (manipulation de structures de données conduisant à de vrais problèmes sur certains systèmes TR !)

Maintenabilité (+ lourd qu'un programme mais pas les TU à refaire) car nécessité de bien appréhender le fonctionnel pour pouvoir intervenir sur les schémas SCADE

Coût de possession en regard de l'utilisation des outils (favoriser la mutualisation des coûts sur « n » projets).

# REX sur Prover Certifier couplé à automates (1/2)

- > 5 ans de mise en œuvre « industrielle »
  - Nombreux projets de postes à aiguillages informatisés (RATP L1, L4, L8, L12, Santiago, … )
  - Expérimentation sur des postes à aiguillages informatisés pour les grandes lignes
- Exemples de mises en oeuvre
  - Vérification d'une spécification de poste à enclenchement
  - Vérification de l'aspect sécuritaire d'un plan technique.
  - Vérification partielle de spécification système ou sous-systèmes CBTC
  - Vérification de l'exécutable d'un poste à enclenchement
- Outil très puissant de raisonnement au niveau système
  - Exécutable considéré comme une boite noire
  - Mise en place facile d'un environnement métier (ferroviaire, …)
  - Expression de propriétés au niveau système
- Outil qualifié par la RATP et s'intégrant dans un processus hautement sécuritaire

# REX sur Prover Certifier couplé à automates (2/2)

- Bilan synthétique

➢ Points positifs :

- Accessible à des personnes ayant fait peu de formels
- Approche basé sur les compétences métier
- Moteur de recherche de contre-exemples
- Evolutions facile et régulière des outils
- Preuve de l'exécutable
- Validation de données inutile

➢ Points négatifs :

- Recherche de lemmes pour aider la preuve
- Raisonnement sur l'instancier et non sur du générique
- Explosion combinatoire (liée à la taille du binaire)
- Lenteur de la solution certifiée

# REX sur SCADE + outils Prover Technology

- > 5 ans de mise en œuvre « industrielle »

  Mise en œuvre sur des parties de C/C de métro développé en SCADE (L5, L13…)

- Utilisation sur projets

  Transposition d'exigences ayant conduit aux planches SCADE en langage de preuve puis vérification de la tenue de ces dernières à l'aide des outils Prover

  Complément de la preuve non réalisable par des tests

  Approche similaire à partir d'exigences de plus haut niveau comme alternative au rejeu de tests de validation (approche RATP)

- Intérêt et limitations

  + Identifie clairement les propriétés de fonctions codées en SCADE / remplace les tests unitaires

  + Augmente le niveau de confiance par l'exploration exhaustive des états atteignables

  + Quasi-indépendance de la vérification vis à vis des détails de conception (« boîte noire »)

  - Limitation classiquement rencontrée avec ce type d'outils (explosion combinatoire…)

  - Aptitude d'un évaluateur à extraire/identifier les propriétés à vérifier à partir des spécifications

# REX sur OVADO

Voir la présentation « Validation formelle des données »

par Christophe Metayer

# Bilan du formel dans le ferroviaire (1/2)

- ## Une palette de solutions

  Produire correct par construction avec assistance de la preuve

  Produire et vérifier progressivement

  Produire puis vérifier formellement, a posteriori, la correction du produit

  Traiter formellement la génération ou la vérification de données critiques de configuration

- ## Passage à l'échelle réussi

  Développement complet prouvé de logiciels de métros automatiques, de sous-systèmes de signalisation (codeur, EVC…)

  Vérification formelle de données de sécurité sur des jeux (> 100 000 param. et 1 000 propriétés)

  Analyse des systèmes par modélisation et preuve (Event B, Simulink…) de + en + fréquente

- ## Gain économique

  Prise en compte dans les normes -> ajustement des activités (documentation, vérification)

  Exigences raisonnables sur les outils de preuve ou de vérification

  Suppression ou mécanisation d'un ensemble d'activités sans forte valeur ajoutée

  Meilleure maîtrise de la spécification et la conception des systèmes

# Bilan du formel dans le ferroviaire (2/2)

- ## Avantages

  Amélioration significative du niveau de confiance dans la sécurité d'un système

  Optimisation des développements par une efficacité croissante des méthodes et outils : schémas de raffinement prédéfinis, performance croissante de la preuve en automatique, exhibition de contre-exemples…

  Prise en compte partielle ou totale par la norme CENELEC

- ## Inconvénients et manques

  Ne couvre pas toutes les problématiques des C/C (flottants…).

  Encore des limitations de performances en fonction de la taille ou de la complexité des systèmes (PAI grandes gares, spécification complète d'un CBTC…)

  Courbe d'apprentissage de certaines technologies (Event B et méthode B par exemple)

  Equilibre ROI / coût de possession

→ Imagine-t-on aujourd'hui faire marche arrière ?!

# Conclusion

Le panel actuel de techniques formelles applicable au monde ferroviaire

est une occasion unique de réattribuer à l'ingénieur son rôle originel,

à savoir spécifier et concevoir des systèmes ou des logiciels,

raisonner sur ces derniers avec des outils puissants,

tout en se déchargeant des activités à faible valeur ajoutée

de pure programmation et pour partie de vérification.

Contacts :

francois.bustany@systerel.fr

christophe.metayer@systerel.fr