

# Application industrielle de la Méthode formelle B

**Guilhem Pouzancre**

**Thierry Servat**

**novembre 2005**

**Contact@Clearsy.com**

EUROPARC de Pichaury  
Bâtiment C1  
1330, av. Guilibert de la Lauzière  
13 856 Aix en Provence Cedex 3

Téléphone : 04.42.37.12.70  
Télécopie : 04.42.37.12.71  
**www.clearsy.com**



# Plan

- ❑ **Clearsy en deux mots**
- ❑ **Quelques définitions**
- ❑ **Principes de la méthode**
- ❑ **Les outils**
- ❑ **Application industrielle : Système sécuritaire  
SIL 3 (RATP)**
- ❑ **Application industrielle : SPRAT (CNIM-DGA)**

# Clearsy en deux mots

Clearsy exploite ses compétences en modélisation formelle pour réaliser ses prestations de développement de logiciels ou de systèmes

## Ingénierie

- Système (dev et appli)
- Logiciels Garantis
- Sûreté de Fonctionnement

## Recherche & Développement

- Innovation méthodologique
- Expérimentation de modélisation
- Réalisation de Logiciels d'Ingénierie

**Leader dans le développement industriel de la Méthode B**

Méthode de modélisation formelle avec preuve

**Éditeur de l'Atelier B**



# Systemes et methodes formelles

(extrait de la norme RTCA DO-178B/EUROCAE ED-12B)

*Une analyse avec methode formelle peut fournir la preuve que le systeme est complet et correct vis à vis de ses exigences.*

*L'utilisation de specifications formelles seules rend les exigences non ambiguës.*

# Logiciels et méthodes formelles

(extrait de la norme aéronautique DO-178B)

*L'utilisation de méthode formelle a pour but d'éviter et d'éliminer les erreurs de spécification, de conception et de codage lors du développement du logiciel.*

## Norme du ferroviaire

*Pour les spécifications, des méthodes formelles mathématiques sont recommandées car le modèle formel fournit précision, non ambiguïté et cohérence.*

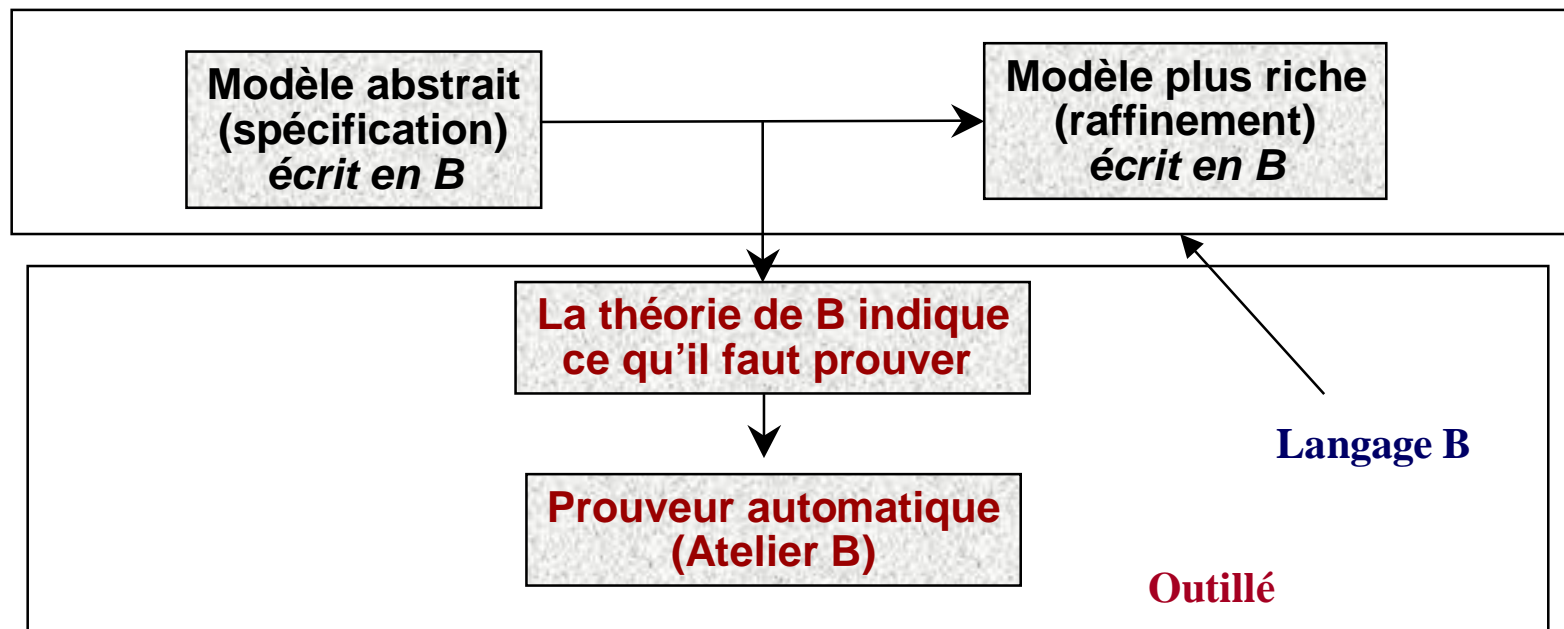
# Norme Critères Communs pour l'évaluation de la sécurité des systèmes d'information

Préconise l'utilisation de modèles formels à partir du niveau 5 et indique que le niveau 7 est atteint *par la réalisation d'une spécification et d'une conception formelle de haut niveau, avec démonstration formelle de la correspondance entre elles.*

# Principes de B

## 3 ingrédients : modélisation, raffinement et preuve

Le langage B est basé sur la théorie des ensembles et la logique des prédicats



# Apports du langage formel et de la preuve

- ❑ **Le langage mathématique amène la précision**
- ❑ **Le raffinement amène la structuration des modèles :  
décomposition, précision, traçabilité, preuve**
- ❑ **Les preuves de propriétés amènent cohérence des  
fonctions entre elles et la vérification de celles-ci au  
besoin**
- ❑ **Les preuves de code logiciel : divisions par zéro, boucles,  
dépassement de tableau, mémoire**

# Deux applications : système et logicielle

## Méthode B

**B'Logiciel**

**B'Système**

**Développement de logiciels sûrs prouvés**

**Modélisation de systèmes**

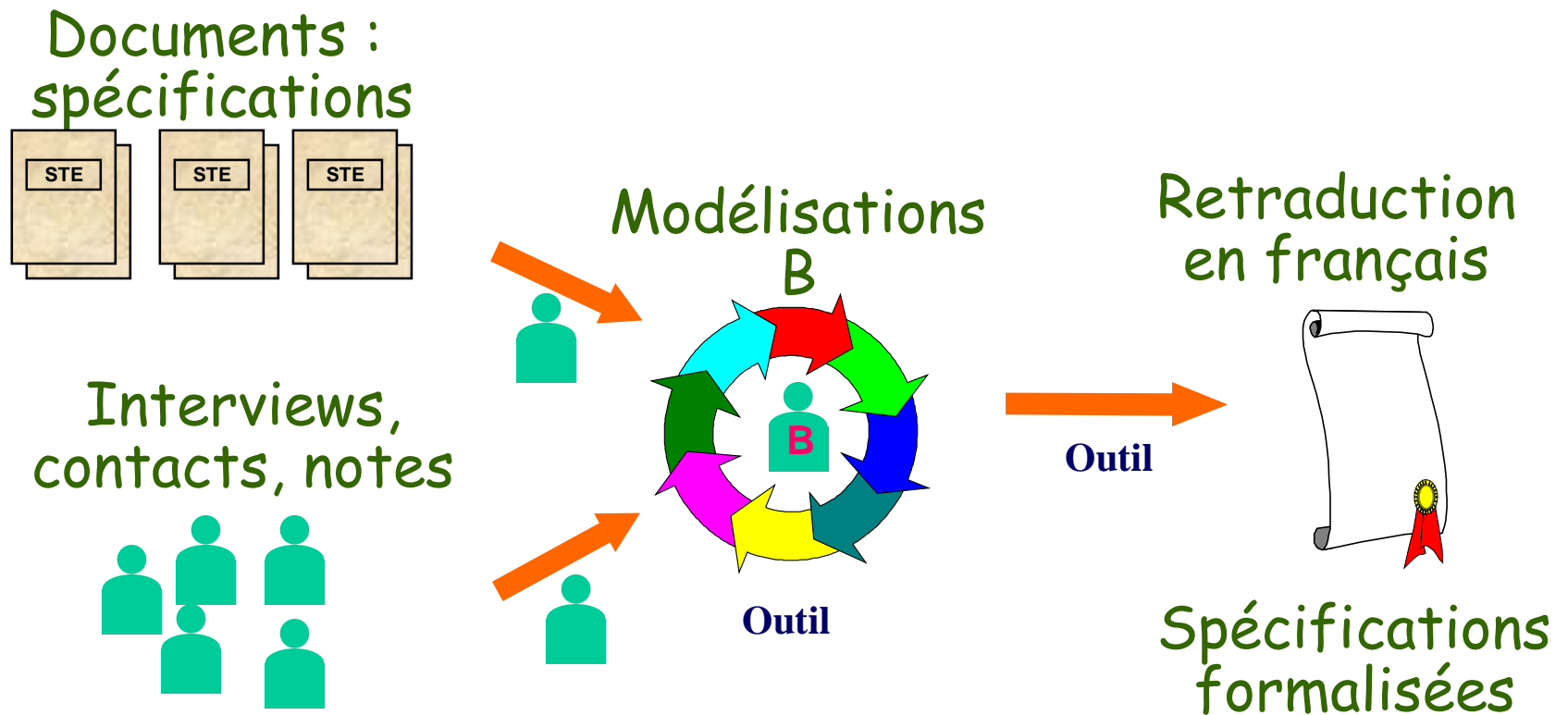
Méthode outillée  
Par CompoSys  
Et des outils de  
l'Atelier B

**Preuve formelle de propriétés**

**Exploitation du modèle**

Méthode outillée  
Par l'Atelier B et  
B4free

# Processus simplifié B'Système



# Processus simplifié B'Logiciel

- ❑ **Modélisation des spécifications informelles du logiciel**
- ❑ **Raffinement des modèles en modèles plus précis**
- ❑ **Jusqu'à obtenir un code en langage B0**
- ❑ **La preuve est réalisée au fur et à mesure**
- ❑ **Traduction automatique des modèles en un langage informatique**

# Les outils

- ❑ **Pour le logiciel : Atelier B (industriel) et B4free (académique et expérimental)**
- ❑ **Pour la réalisation de modèles systèmes (sans raffinement) : CompoSys (Beta test)**
- ❑ **Pour la réalisation de modèles systèmes prouvés : les trois outils utilisés par des experts**

# Les outils futurs en cours de développement

## ACADEMIQUES

- Un projet Européen en cours
- Réalisation d'une nouvelle plate forme ouverte et open source d'outils d'aide à la modélisation système
- Une plate forme open source sous Eclipse

## INDUSTRIELS

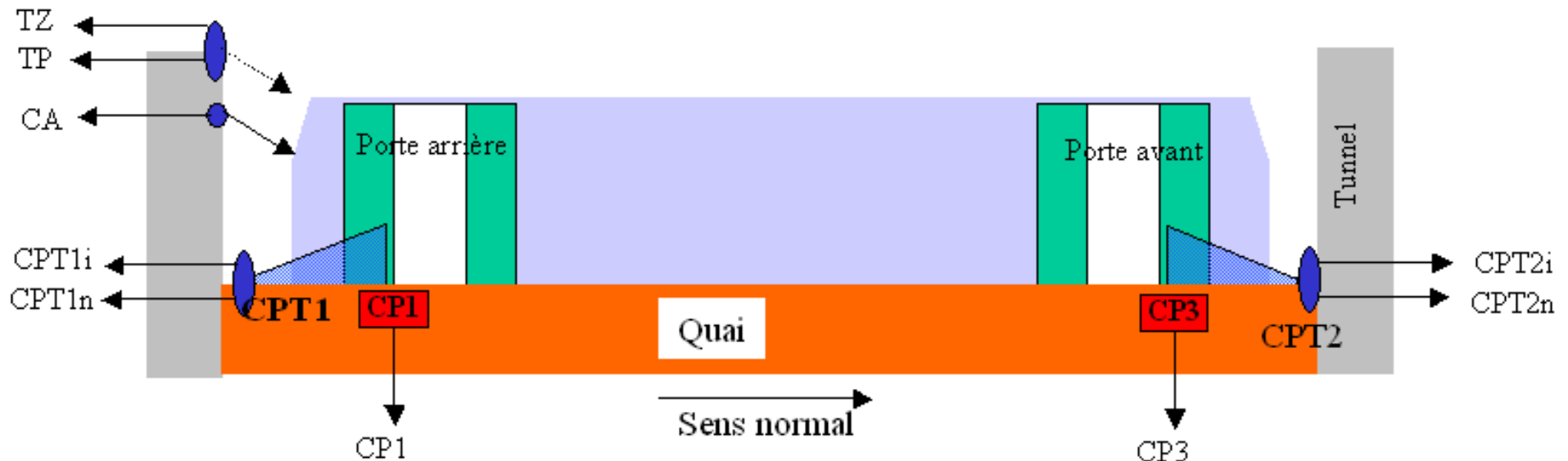
- Réalisation d'outils industriels à partir de la plate forme Rodin
- Outil d'animation de modèles et de documentation

# Des Applications industrielles

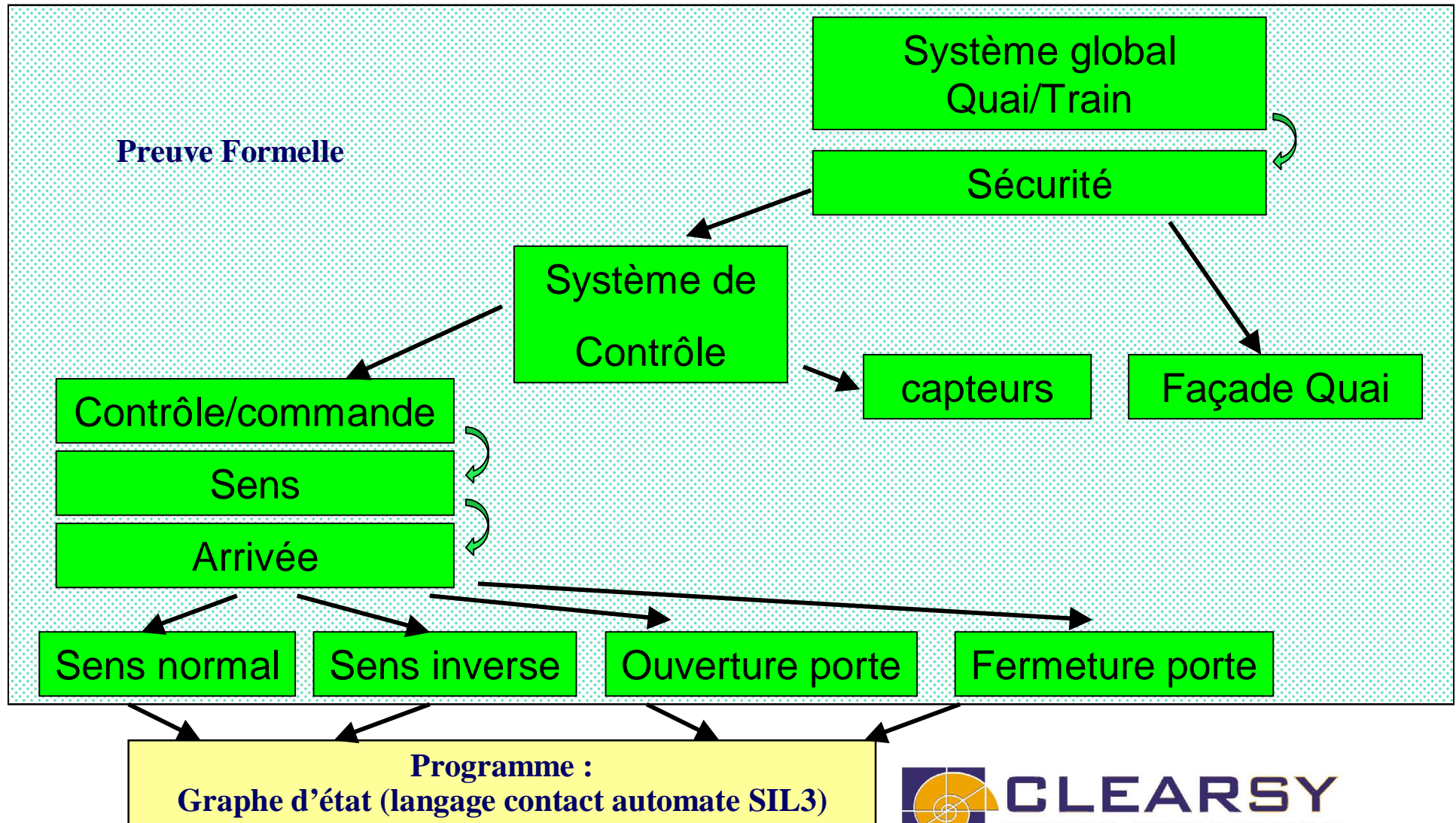
- ❑ **ST : modélisation de politique de sécurité de composant**
- ❑ **PEUGEOT : Modélisation des principes de fonctionnement de trois modèles de voitures et production des données de diagnostic associées**
- ❑ **CLEARSY : Développement du système de contrôle de portes palières pour RATP**
- ❑ **SIEMENS : Développement des automatismes embarqués et au sol du métro sans conducteur Meteor, de Barcelone et de la Carnasie Line de New York**
- ❑ **CLEARSY : Développement des logiciels sécuritaires des automatismes fixes du futur Val de Roissy pour Siemens**
- ❑ **Expérimentation de modélisation du système de commande des tuyères d'Ariane par Clearsy pour le CNES**
- ❑ **Modèles de politiques de sécurité de composants hardware**

# Exemple d'un système de Façade (métro)

- ❑ 2 Capteurs infrarouges : **détection de la présence d'un train**
- ❑ 1 Télémètre hyperfréquence : **détection de la position et de la présence d'un train**
- ❑ 1 Radar DOPPLER : **détection des mouvements du train**
- ❑ 2 Radars DOPPLER : **détection des mouvements des portes du train**



# Systeme de Façade de Quai



# Exemple industriel de modélisation système



« Le "SPRAT" est le système de franchissement le plus novateur actuellement conçu. Il n'a aucun équivalent au monde, et intéresse plusieurs armées étrangères. »

Extrait : [www.cnim.fr](http://www.cnim.fr)

# Réalisation d'un document de conception système (avec CompoSys)

Pour chaque sous-système on décrit :

- Ses fonctions
- Ses interfaces (fil, capteur, actionneur, paramètre réseaux, circuit hydraulique, circuit pneumatique, etc.)
- Ses règles de fonctionnement

Résultat : 50 sous-systèmes électroniques, 400 interfaces, 200 fonctions, ont été décrits.

## Utilisation d'une méthode et d'un outil pour les descriptions : CompoSys

- Modélisation couplée : méthode formelle B / langage naturel
- Vérifications automatiques et semi-automatiques de la cohérence
- Génération automatique du document final et de différentes vues du système

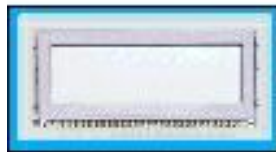
# Exemples de sous systèmes

## Fonction climatisation assurée par 5 éléments

Dans la maquette les schémas sont beaucoup plus détaillés

CAN Moteur

1. Ecran déporté

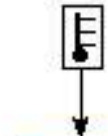


CAN  
Servitude

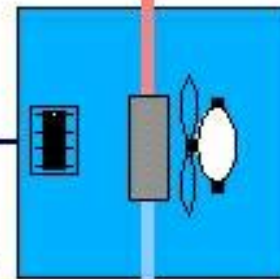


2. Console  
multifonctions

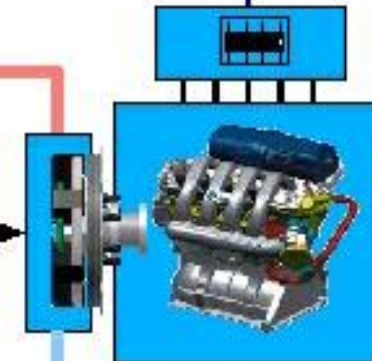
3. Système de  
servitude



4. Système de  
climatisation



5. Moteur +  
ECU + compresseur



# Méthode : cinq tâches itératives

1. Décomposition du système en sous-systèmes
2. Recensement des fonctions des sous-systèmes
3. Descriptions formelles des fonctions
  - ❑ Quels paramètres et comment elles les utilisent
4. Description informelle de l'implantation des interfaces dans le système
5. Enrichissement du modèle formel
  - ❑ Langage naturel, schémas, autres formalismes.

# Modèle à double face

## □ Face formelle

- ✓ Explications non ambiguës
- ✓ Automatisation des vérifications et calculs des vues

## □ Face informelle

- ✓ On réalise un lien entre les entités formelles et les entités du système
- ✓ Toutes les interfaces et les fonctions sont décrites

# Face formelle

## ❑ Vérifications Syntaxiques Automatiques

- ✓ Vérifications syntaxiques : modèle, liens modèle – langage naturel ...
- ✓ Vérifications de type, portée des paramètres, ...
- ✓ Règles de cohérences : environ 50 dans **COMPOSys**.

Ex : Vérifier que chaque paramètre est utilisé et produit par un sous-système, avec des types de données compatibles

## ❑ Preuves formelles : automatiques / semi-automatiques

- ✓ Invariants/Raffinements/Abstractions : techniques pour reformuler les descriptions (utile pour les descriptions compliquées)
- ✓ Preuves mathématiques que les différentes formulations ne se contredisent pas.

Exemple d'Invariant : Le véhicule a besoin de courant électrique pour démarrer

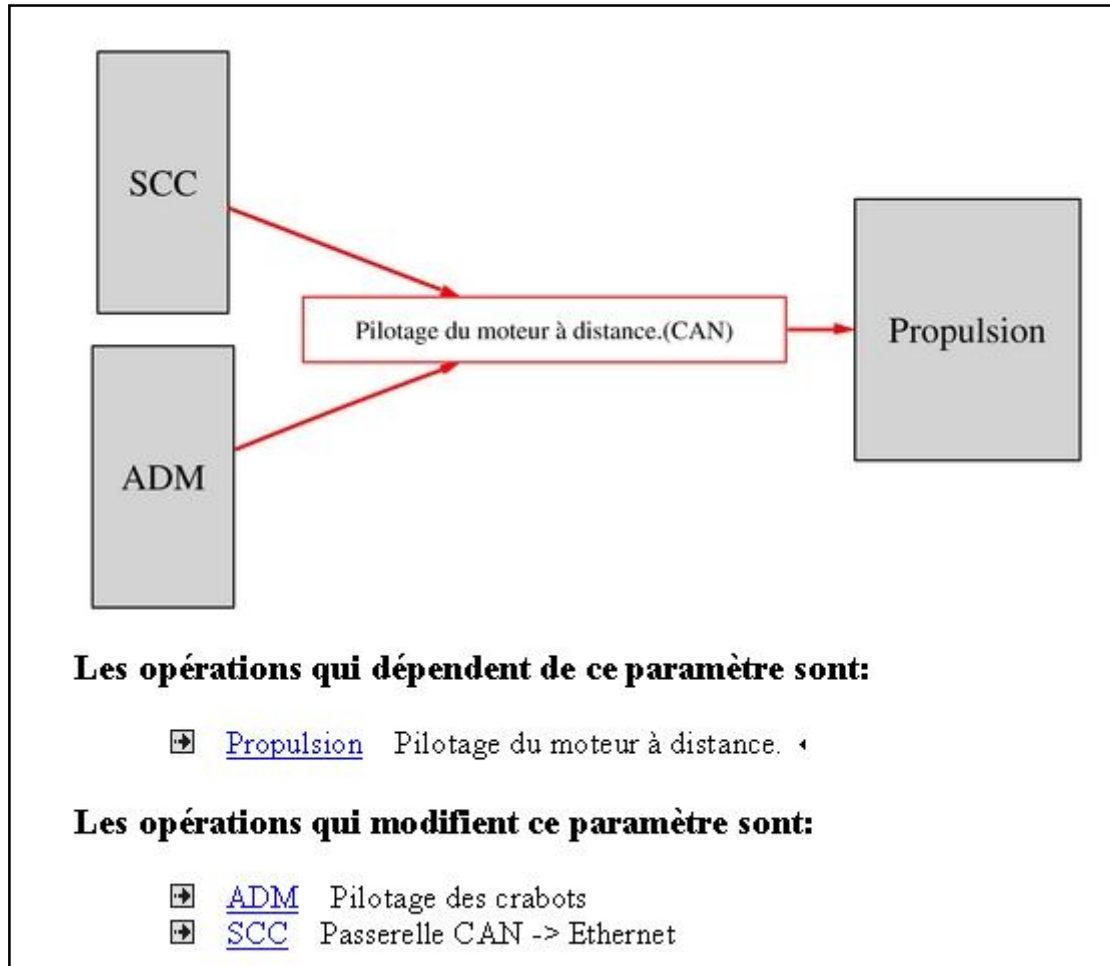
**Moteur = Dem => AlimPrincipale = TRUE**

## ❑ Génération automatique des vues.

- ✓ L'outil génère plusieurs représentations de la même information.

# Face informelle

Les vues générées sont en langage naturel



Autres exemples de vues :

- Matrices réseaux
- Réseaux électriques
- Vue par composant
- Vue par chaîne fonctionnelle

# Apports lors de la création du modèle : la précision d'un langage mathématique

## □ Les spécifications fonctionnelles :

- ✓ Elles sont vérifiées
- ✓ Elles sont complétées
- ✓ Des difficultés sont anticipées

**Une « pré-intégration » fonctionnelle des  
composants est réalisée**

# Apport lors de l'exploitation du modèle

- ❑ **Le modèle est une base d'information commune aux activités transverses de :**
  - ✓ Vérifications et intégration
  - ✓ Sûreté de fonctionnement
  - ✓ Dimensionnement des réseaux
    - CAN, électriques, ...
  - ✓ Études d'impact
  - ✓ Formation

# Point de vue

## ❑ Principales difficultés techniques :

- ✓ Choix du niveau de formalisation (à confier à une personne expérimentée)
- ✓ Des descriptions formelles multi-métiers (mécanique, électronique, hydraulique, ...) qui requièrent des notions de mathématiques

## ❑ Principaux avantages

- ✓ Une approche guidée, outillée et éprouvée sur des cas industriels.
- ✓ Très bonne et rapide appréhension du système.
- ✓ Connaissance transmissible et réutilisable.
- ✓ Économique, chiffrable, travail itératif.

# Bilan des modèles Sprat

- ❑ **Le modèle est une « *base d'informations* »**
  - ✓ Sous-systèmes, Fonctions, Paramètres
  - ✓ Que l'on alimente en « expliquant » les fonctions
  - ✓ « Explications » mixtes : B et autre formalisme
  
- ❑ **Utilisation de cette base pour les travaux transverses**
  - ✓ Vérifications de compatibilité entre les composants
  - ✓ SDF, bilans électriques, diagnostic ...
  - ✓ Calculs
  
- ❑ **Une méthode outillée et économique de modélisation en B'Système : COMPOSYS**
  - ✓ Bientôt une version en bêta test.
  - ✓ Actuellement ce type de modélisation est réalisée par ClearSy