AtelierB
Captronic
01 octobre 2013

ClearSy
contact@clearsy.com

Téléphone :
04.42.37.12.70
01.40.28.14.57

www.clearsy.com

Sommaire

- Résumé des dernières années
- Politique de distribution source et budget
- Projet : développement logiciel
- Projet : développement système
- Projet : de validation de donnée
- Développement en cours
- Développement envisagé

AtelierB

Atelier logiciel qui permet une utilisation opérationnelle de la méthode formelle B.

- Ces fonctionnalités se regroupent en quatre catégories :
  - Une aide à la preuve : pour démontrer les obligations de preuve, grâce à des outils de preuve
  - Une aide au développement : Gestion des dépendances entre composants B, Le raffinement automatique, Analyse statique
  - Des outils de confort pour l'utilisateur: Représentation graphique de projets, Affichage de l'état d'un projet, Gestion des règles

Résumé des dernières années

1994

METEOR

MATRA

2000

Clic'n'prove

ROISSY

SIEMENS

PEKIN

ALSTOM

ComenC

Bart

AREVA

1998

2010

EVENT B

COPPILOT

deploy

PARIS, SAO PAULO

Politique de distribution

Version :

- V X.Y : 1 Version tous les 2 ans environ. Téléchargeable directement sur le site http://www.atelierb.eu (V4.1).
- V X.Y.z : Version de maintenance (version de correction + version spécifique projet) .Téléchargeable après identification sur le site

Source

Source :

- Les IHM en Qt et le B compilateur (Bcomp) sont open source

sourceforge

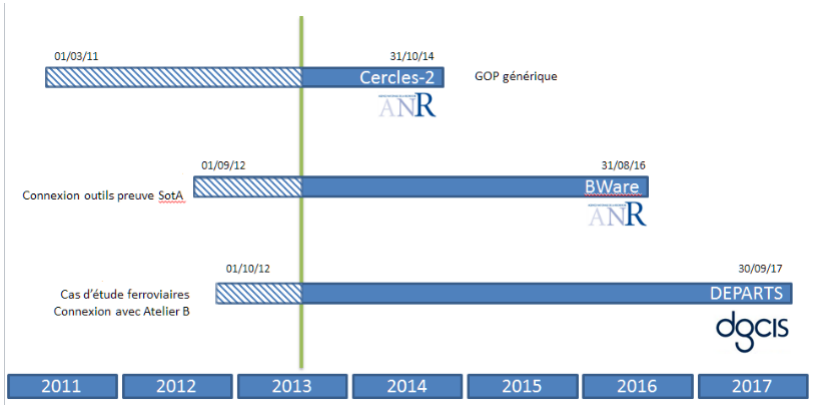
Regrouper sur http://www.tools.clearsy.com/tools/ :

CLEARSY
SYSTEM ENGINEERING
Tools

- Les autres outils (MS, krt, TC, PO, PR) restent fermés :
  Validation/Qualification des modifications

Financement de l'AtelierB

- Projet de R & D (80%) :
  - CERCLE 2
  - DEPARTS BWare
  - DEPARTS
  - Cx



- Maintenance de l'AtelierB (10%) :
  - Entreprise : Variable suivant utilisation.
  - Université : 950€ licences illimité (élève, enseignant/chercheur)
- Développement particulier (10%) :
  - Demande d'un projet Interne
  - Demande d'un client

Développement logiciel

Méthode formelle de développement logiciel

- Modéliser de façon abstraite du comportement d'un programme
- Raffinements successifs jusqu'à un modèle concret transcodable
- Preuve de consistance et de raffinement

Pas de magie

- Les tests sont remplacés par la preuve
- Les résultats obtenus dépendent de l'effort de spécification

  WYGIWYP : What you get is what you provided
- Not a « push-button » method

Projet Urbalis Evolution (**Alstom**)

- **Environ 50 métros en exploitation**
- **11 versions majeures à ce jour**

Plaquette URBALIS

12 raisons de plus
de choisir URBALIS

- **Sécurité élevée –** Utilisation de méthodes formelles pour l'élaboration de logiciels
- **Economies d'énergie –** Jusqu'à 30 % en utilisant au maximum la marche sur l'erre
- **Interfaces standards –** TCP/IP et Ethernet, IEEE 802.11
- **Technologie radio robuste et fiable –** en service sur des lignes majeures depuis plus de 10 ans (depuis 2003)

Some implementations (B)

L1
Paris

L1
Algiers

SHUTTLE
ROISSY AIRPORT
Paris

L2 L3
Sao Paulo

New York
Canarsie

L3
Paris

Lille

Mexico

L9
Seoul

L9
Barcelona

Instanbul

Airport Express
Hong Kong

L2
Budapest

Toronto

Madrid

Metro L10
Beijing

Circle Line
Singapore

KVB
6000 trains
France

San Juan

L1 L2
Malaga

METEOR L14
Paris

Delhi

Metro
Lausanne

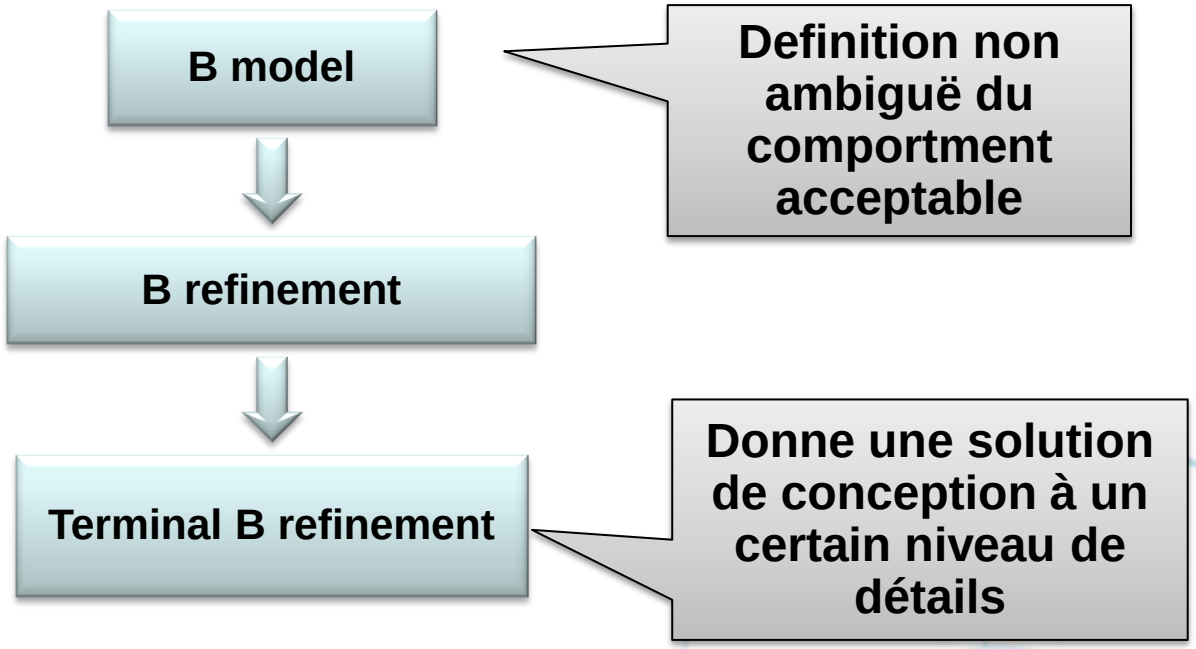
L5
Milano

1990 2000 2010 0

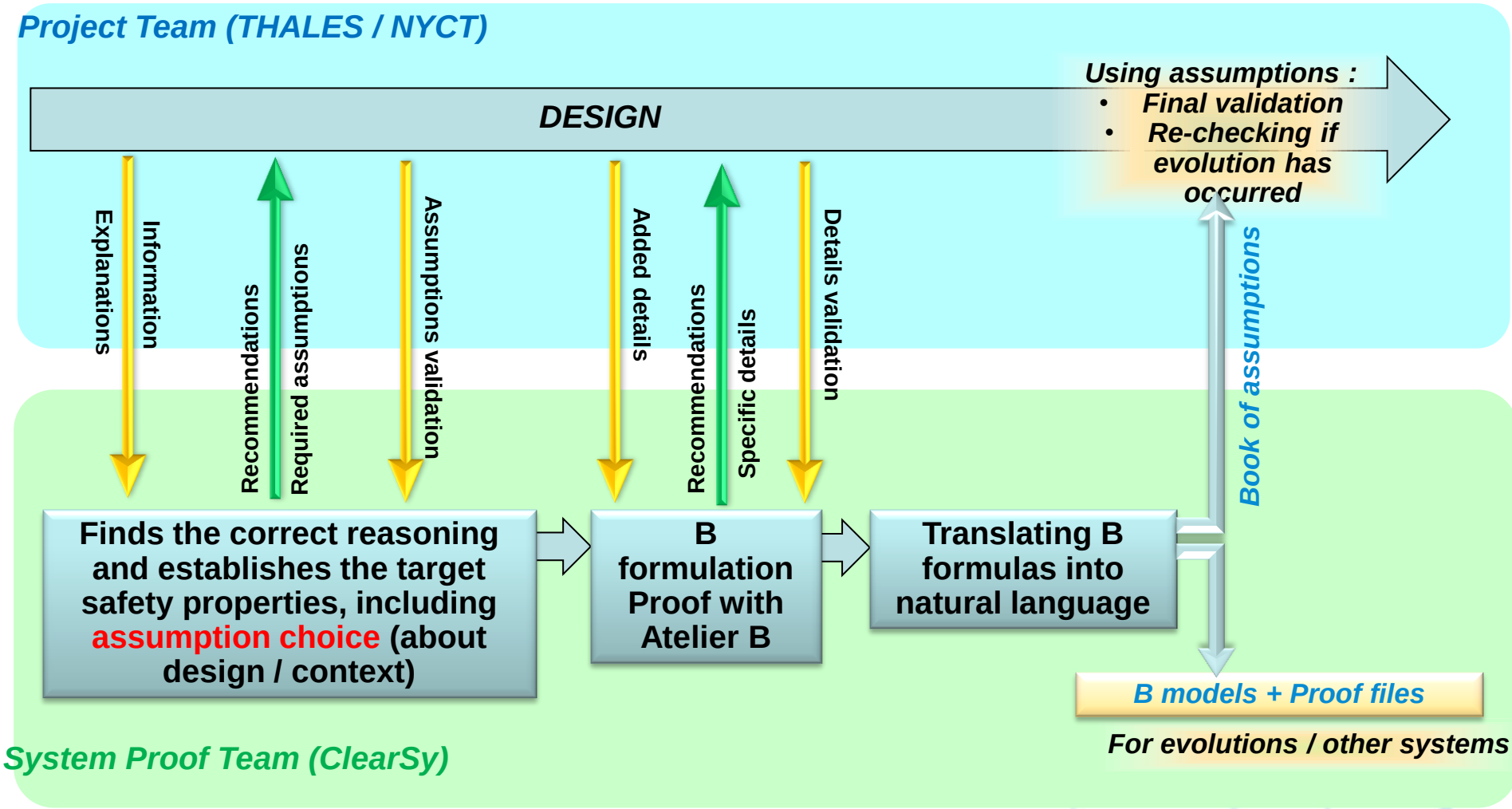
B évènementiel / B système

Utilisée pour décrire formellement les systèmes et raisonner mathématiquement sur leurs propriétés.

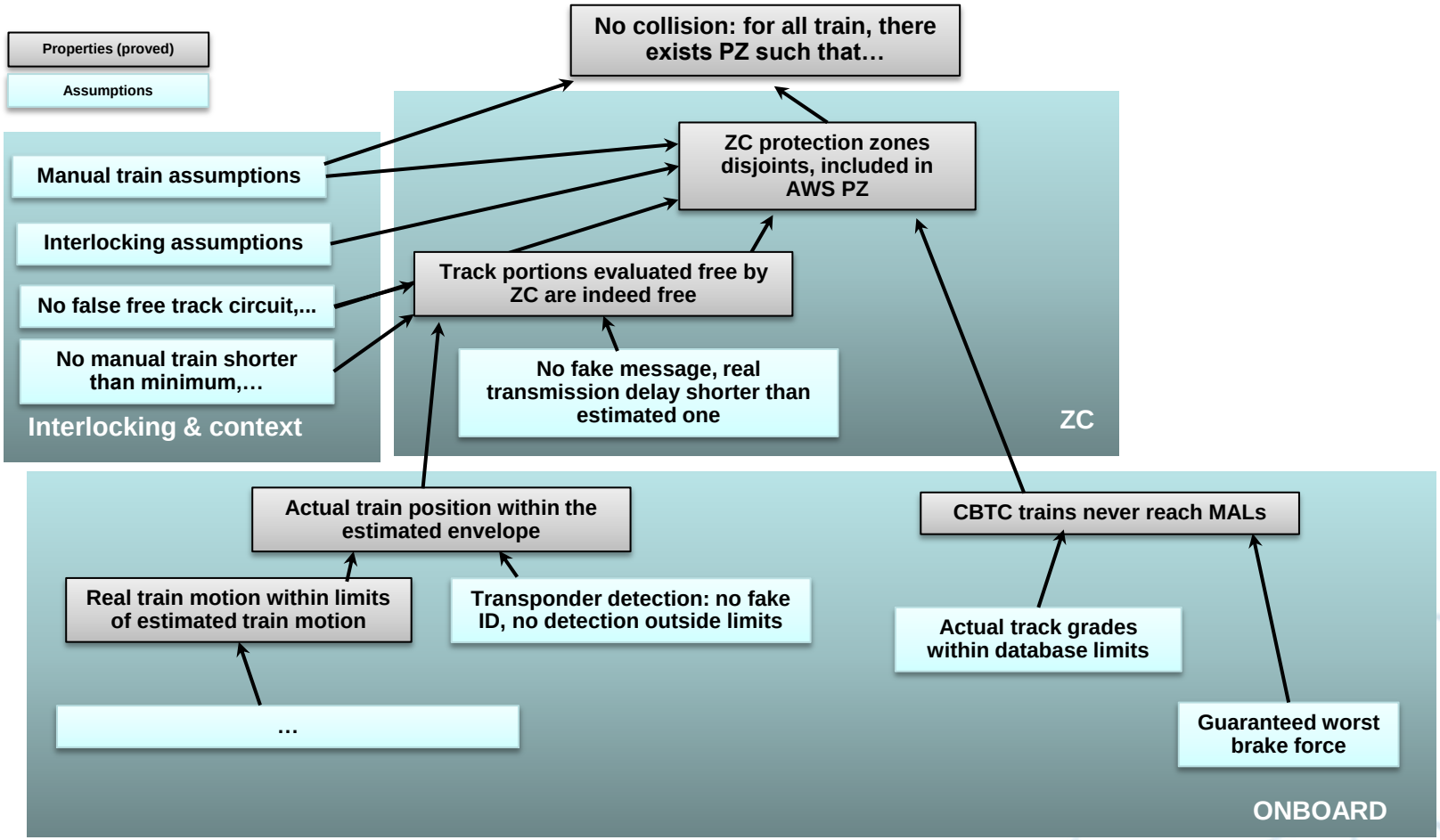
- Basé sur la Méthode B :
  - Formalisme
  - Preuve
  - Raffinement



Flushing line (NewYork)



Properties & sub-properties



Vérifications de Donnée

Utilisée pour en vérifier mathématiquement les propriétés sur les données

Etats de l'art

- Plusieurs projets de R&D ont permis de développer les outils
- Utilisation courante dans le monde industriel : Alstom, RATP, Siemens

Permet :

- Le traitement de grande quantité
- Formalisation des données (consistantes, corrections)
- Validation d'un ensemble de données partiellement construit
- Mise en évidence des contre-exemples de manière simple

Projet : TMS DATA (Alstom)

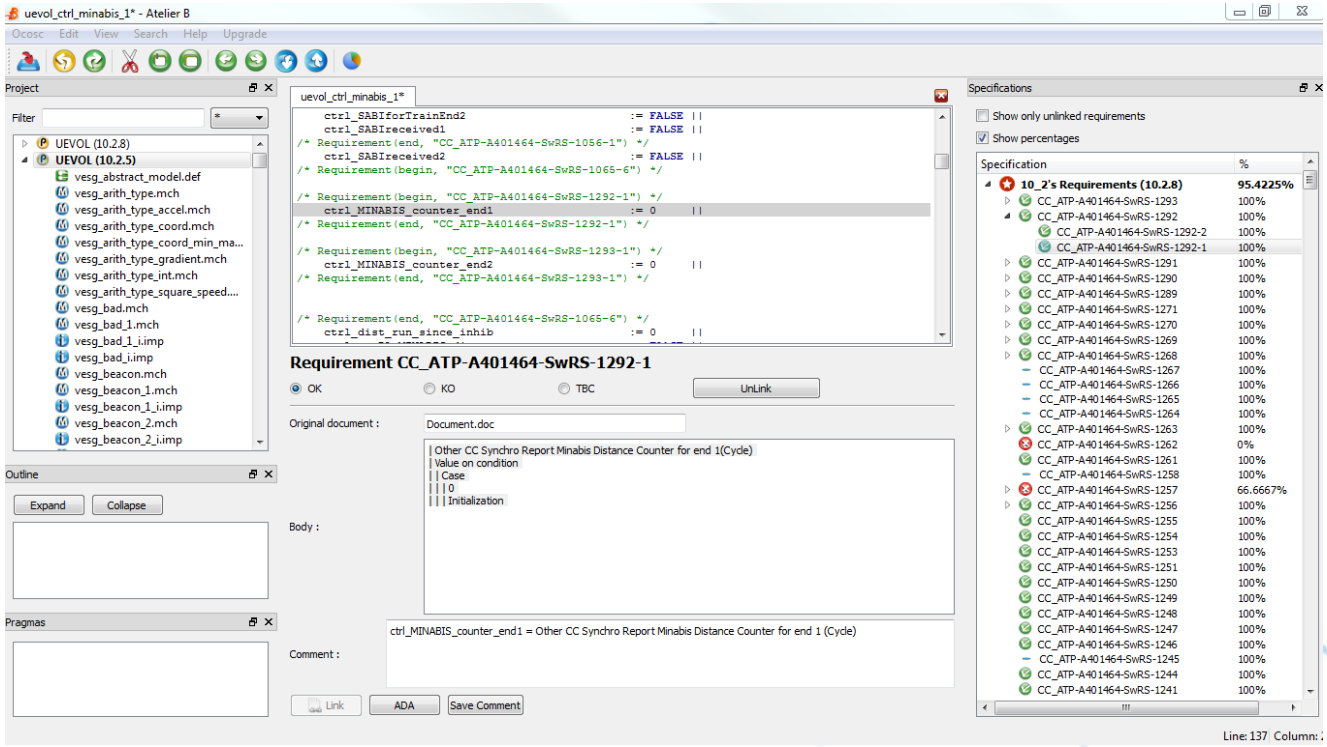
- Gain en temps :
  - Environ 30 jours pour vérifier 300 règles manuellement,
  - Quelques heures pour vérifier 300 règles modélisées
- Plus de 100000 informations
- Environ 1500 propriétés modélisé
- Poursuite de l'automatisation sur d'autres données

En développement

- AtelierB V 4.2.0
  - Début 2014
  - GOP Générique :
    - Possibilité de paramétrer les OP générer.
    - Traçabilité des OP
    - Parallélisassions du GOP.
  - Interfaçage avec plusieurs Prouveurs
  - Gestion du 64 bits
  - Améliorer le rajout de plug-in

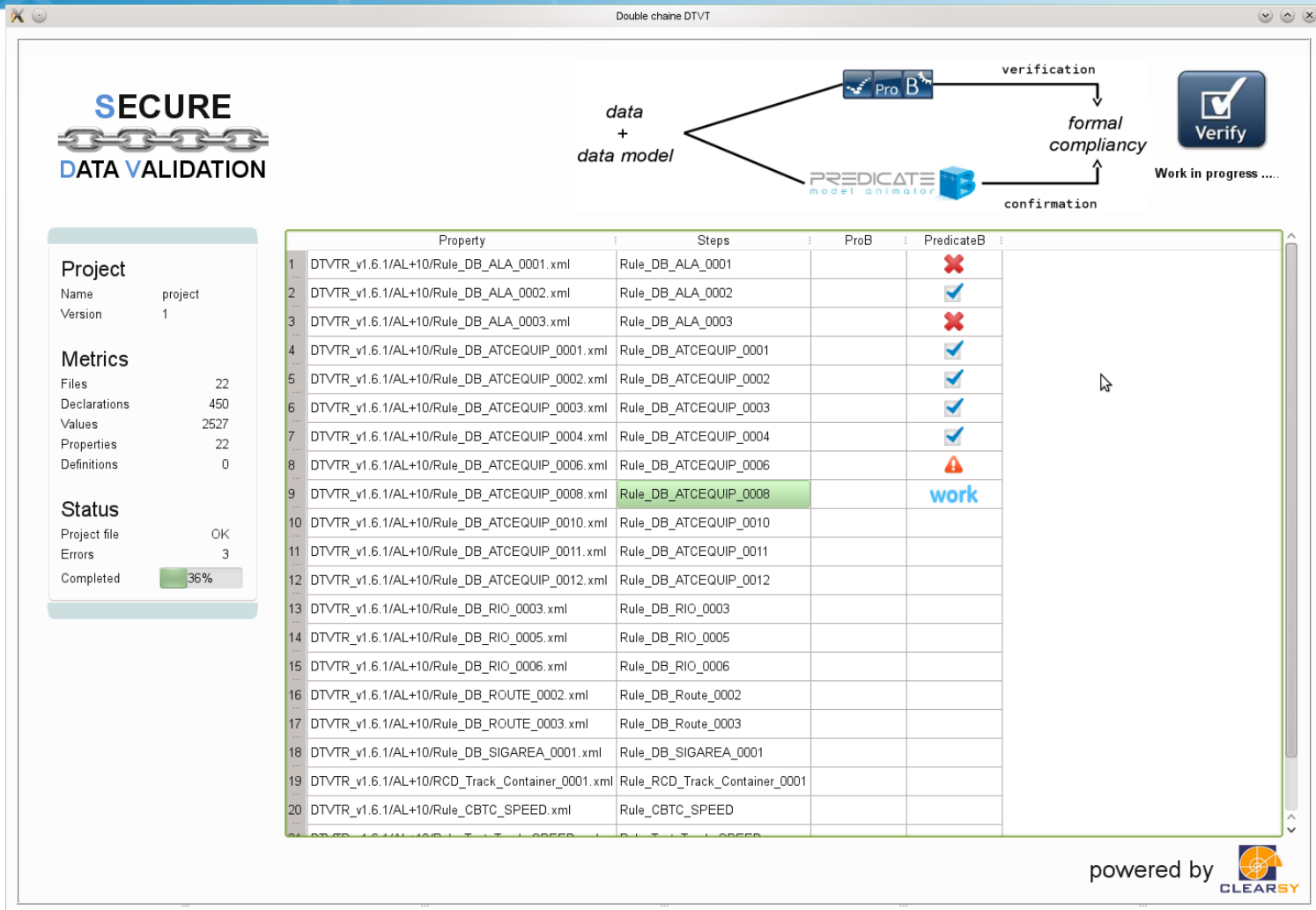
En développement : Gestion des exigences

- **Lier une partie d'exigence avec des substitutions**
- **Vérification du lien et du commentaire**
- **Gestion des versions**
- **Vérification de couverture**
- **Métrique**



En développement : SDV

- Vérification de propriété
- Double chaine de vérification
- Production du rapport (csv)
- Visualisation des témoins



En projet

- Théorique
  - Prise en compte des réels/flottants dans les prouveurs
  - Intégration plus fine avec pro-B
  - Règles (Prouver interactif, recherche …)
  - Représentation graphique d'un modèle (équivalence)
- Pratique
  - Service de gestion d'un pool de serveur.
  - Gestion des modèles en conf par l'atelierB
  - Amélioration de l'interfaçage GOP générique avec l'éditeur
  - Génération de code : VHDL, Ladder, c, assembleur