

Technologie XSL et cycle de développement logiciel

Pierre Gradit,
GDTech France
pierre.gradit@gdtech.fr



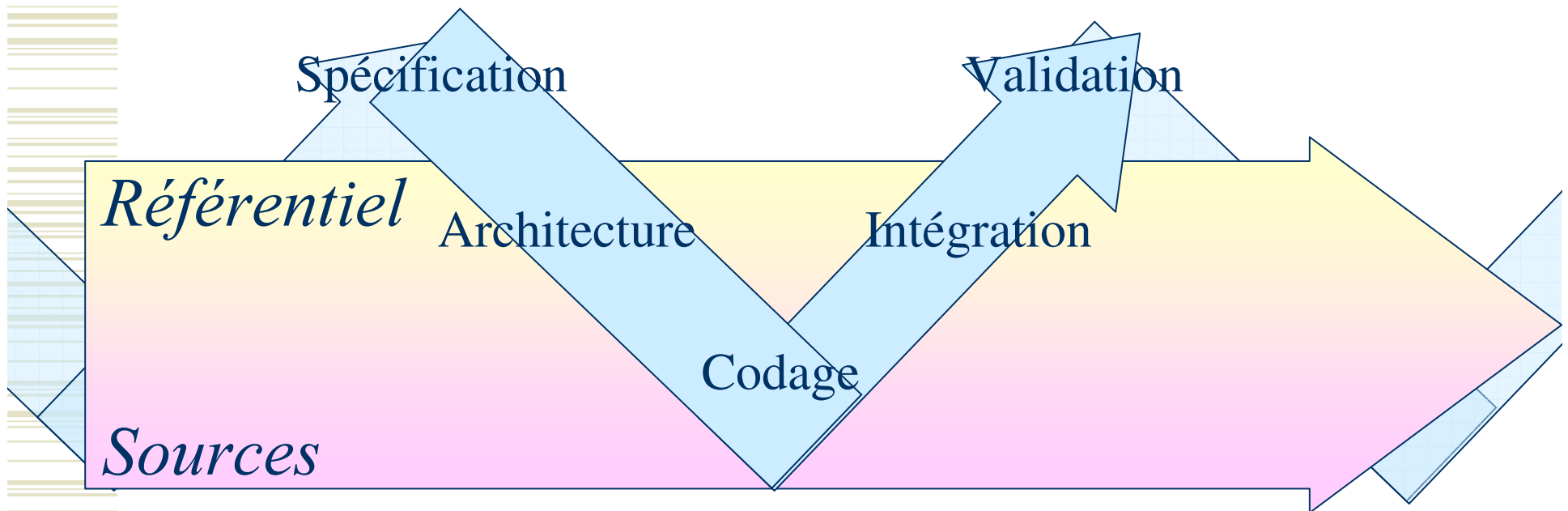
Plan

- ◆ Présentation de la problématique
- ◆ Les opérations du Pilote
- ◆ Expérience acquise lors du projet PTIV
- ◆ Développements prévus

Plan

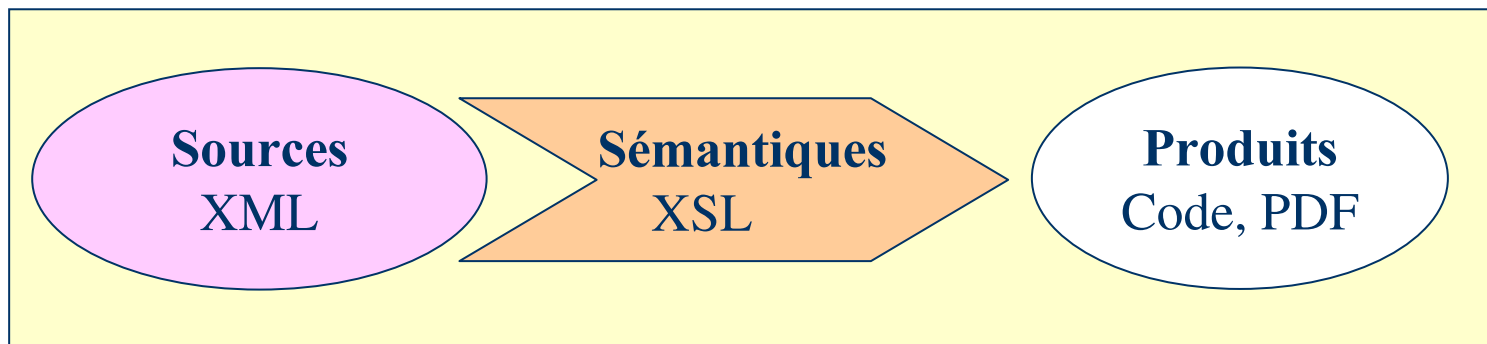
- ◆ **Présentation de la problématique**
 - Cycle de développement
 - Cadre de production
 - Objectif
 - Environnement
 - Implémentation
- ◆ Les quatre opérations
- ◆ Expérience acquise
- ◆ Développements prévus

Le cycle de développement



Comment capitaliser l'expérience acquise ?

Le cadre de production



Est appelé **source** «*l'ensemble des fichiers sources du projet*»

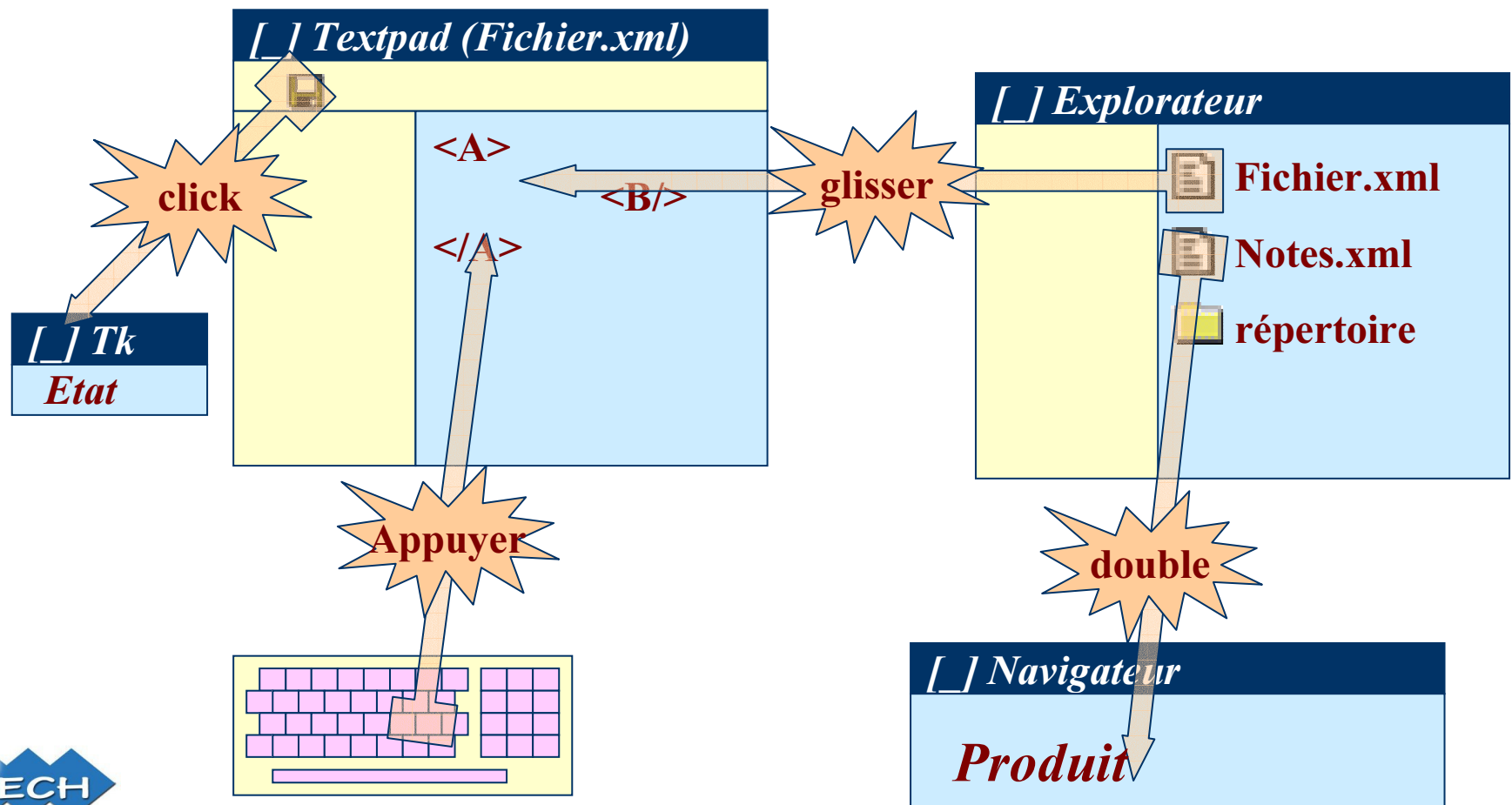
Est appelé **produit** «*tout objet observable sur la machine*»

- Délivrable : livré au client.
- Interne : pour le suivi du projet

L'objectif du Pilote

- ◆ Réalisation de produit :
 - Réutilisable : *les duplications de sources sont minimisées.*
 - Coopérative : *plusieurs personnes travaillent sur le même projet.*
 - Réactive : *les produits sont disponibles le plus souvent possible*
 - Réflexive : *les sémantiques sont des sources*

L'environnement du Pilote



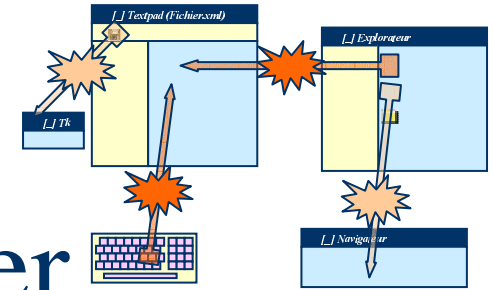
L'implémentation du Pilote

- ◆ Réalisation d'un prototype opérationnel
 - Windows 98/2000/NT
 - Technologie XML/XSL.
 - Glu : Tool Command Language (Tcl)
- ◆ Intégration de composants extérieurs
 - Utilisation de l'interface COM de Windows

Plan

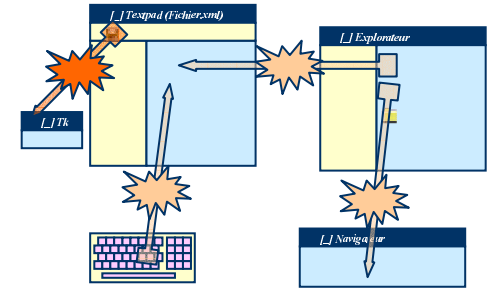
- ◆ Présentation de la problématique
- ◆ Les opérations du Pilote
 - Glisser/Appuyer : *Edition*
 - Clicker : *Sauvegarder*
 - Double Clicker : *Visualiser*
- ◆ Expérience acquise
- ◆ Développements prévus

Glisser/Appuyer



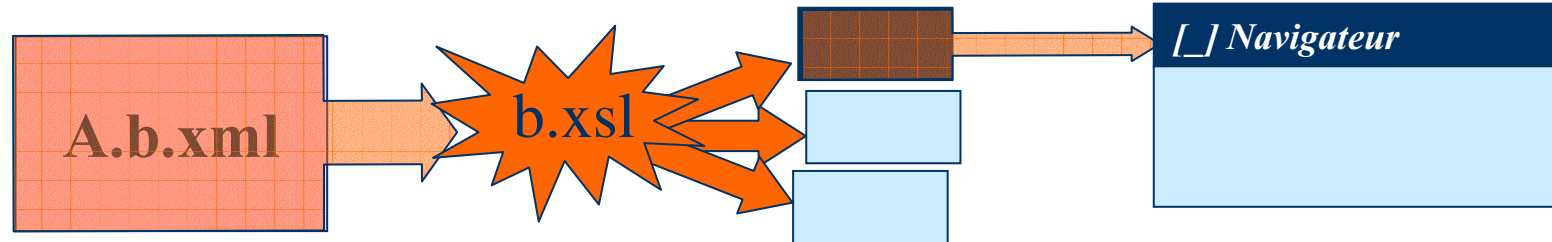
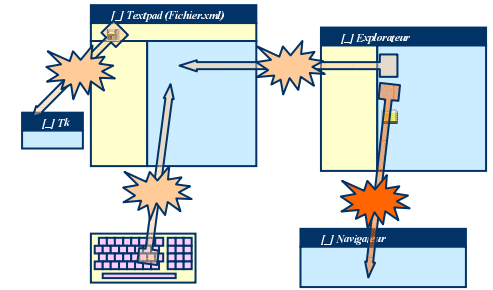
- ◆ Réutilisation de l'éditeur de texte *TextPad*
 - Fonctionnalités exhaustives d'un éditeur (undo, search, ...).
 - Détection de l'édition par un autre utilisateur
 - Capacité de développer des réflexes sur les boutons.
- ◆ Tous les utilisateurs peuvent lire les sources

Clicker



- ◆ Dispositif de gestion de la sauvegarde
 - Sauvegarde de fichiers correct syntaxiquement
 - Réflexe sur la sauvegarde (vérification du XML, autorisation d'écriture).
 - Une boite de dialogue informe de l'état de la sauvegarde (copie locale proposée)

Double-click



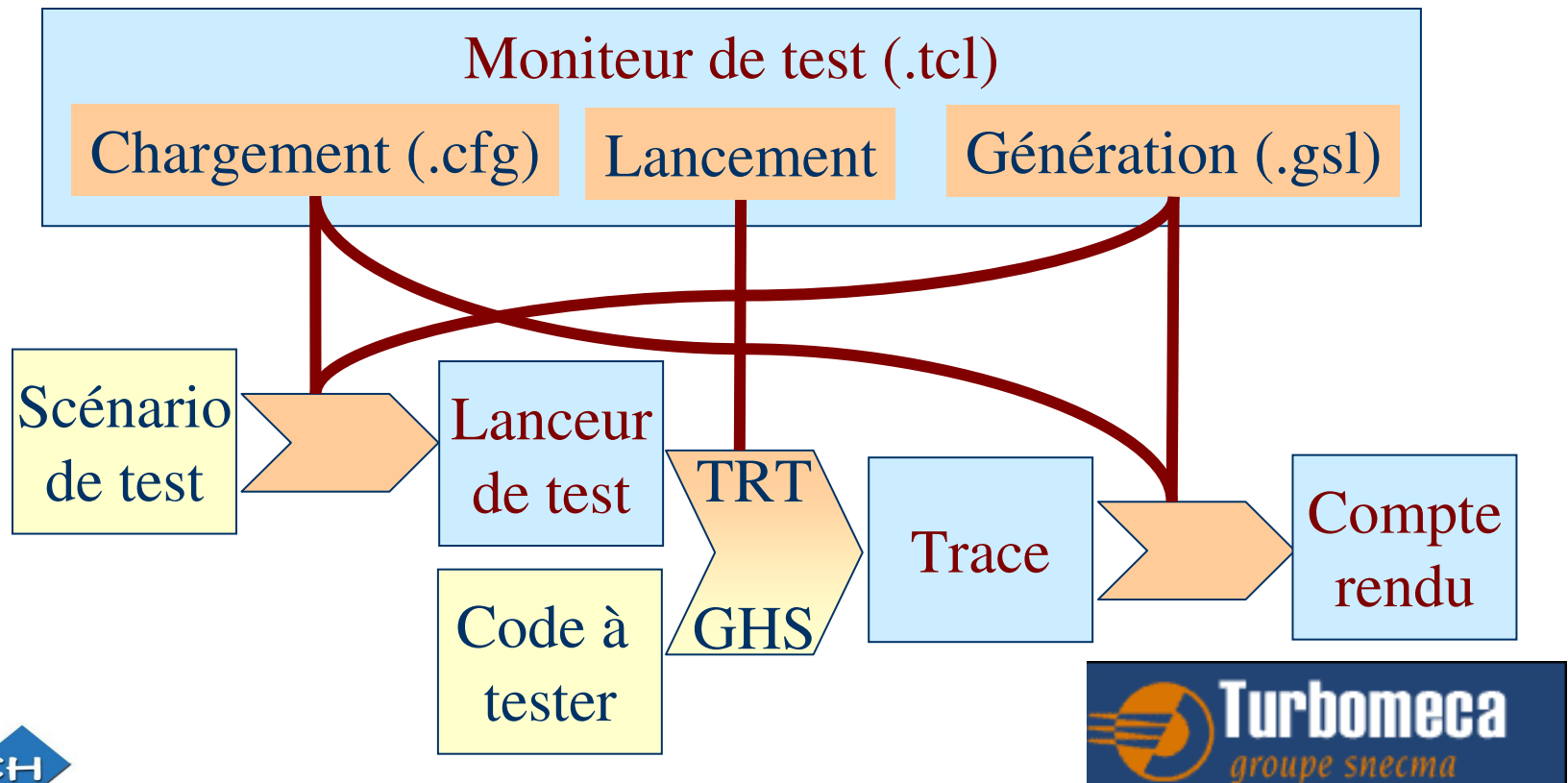
- ◆ Appel à un programme Tcl
- ◆ Détermination de l'XSL utilisé.
- ◆ Possibilité de produire plusieurs fichiers.
- ◆ Explorateur ouvert sur le premier fichier.

Plan

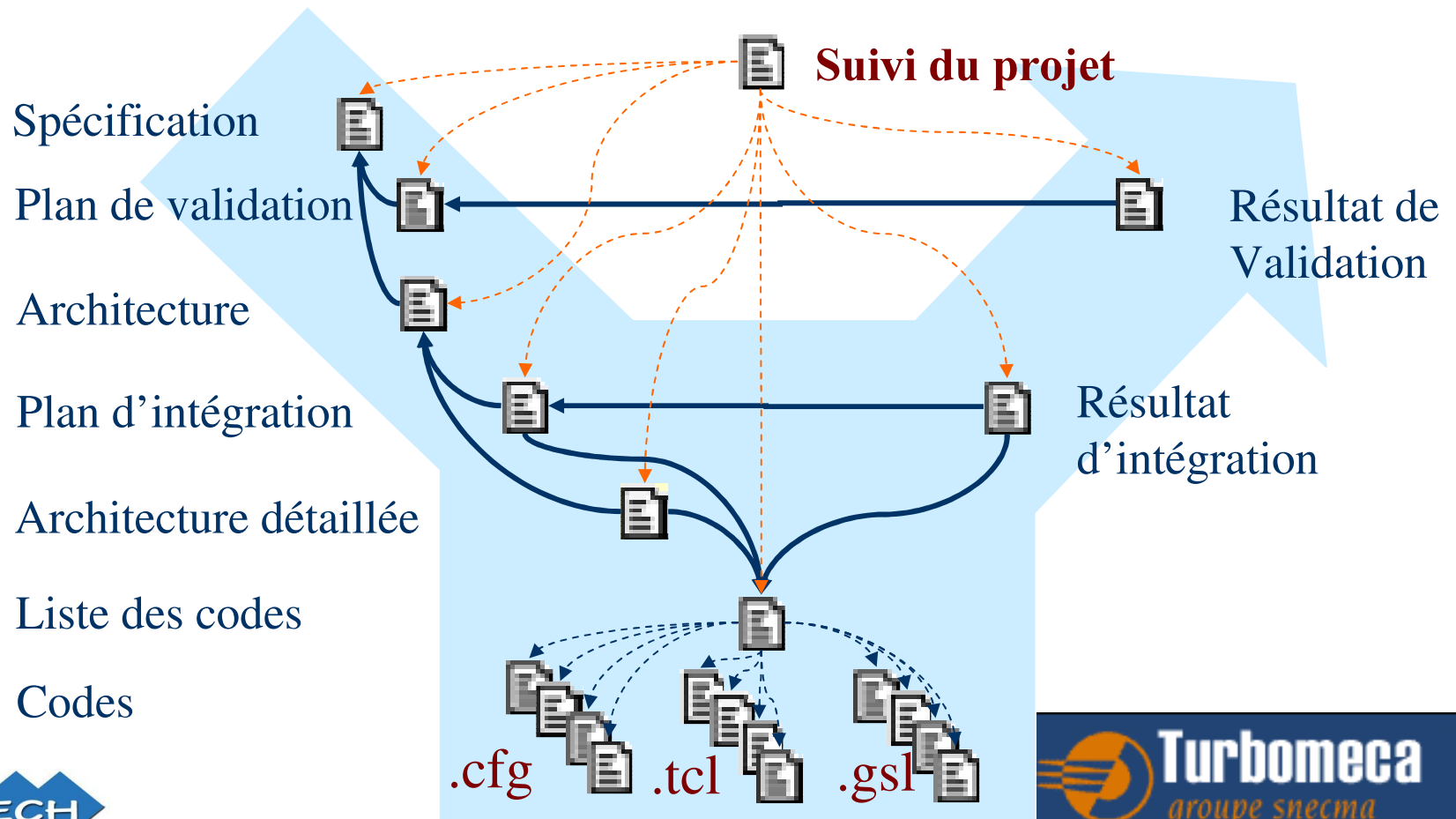
- ◆ Présentation de la problématique
- ◆ Les quatre opérations
- ◆ **Expérience acquise**
 - Projet PTIV
 - Cycle de vie simplifié
 - Architecture détaillée
 - Gestion des tests
 - Bilan du projet PTIV
- ◆ Développements prévus

Poste de Test et d'Intégration Virtuel

- ◆ Simulation de baie automatique



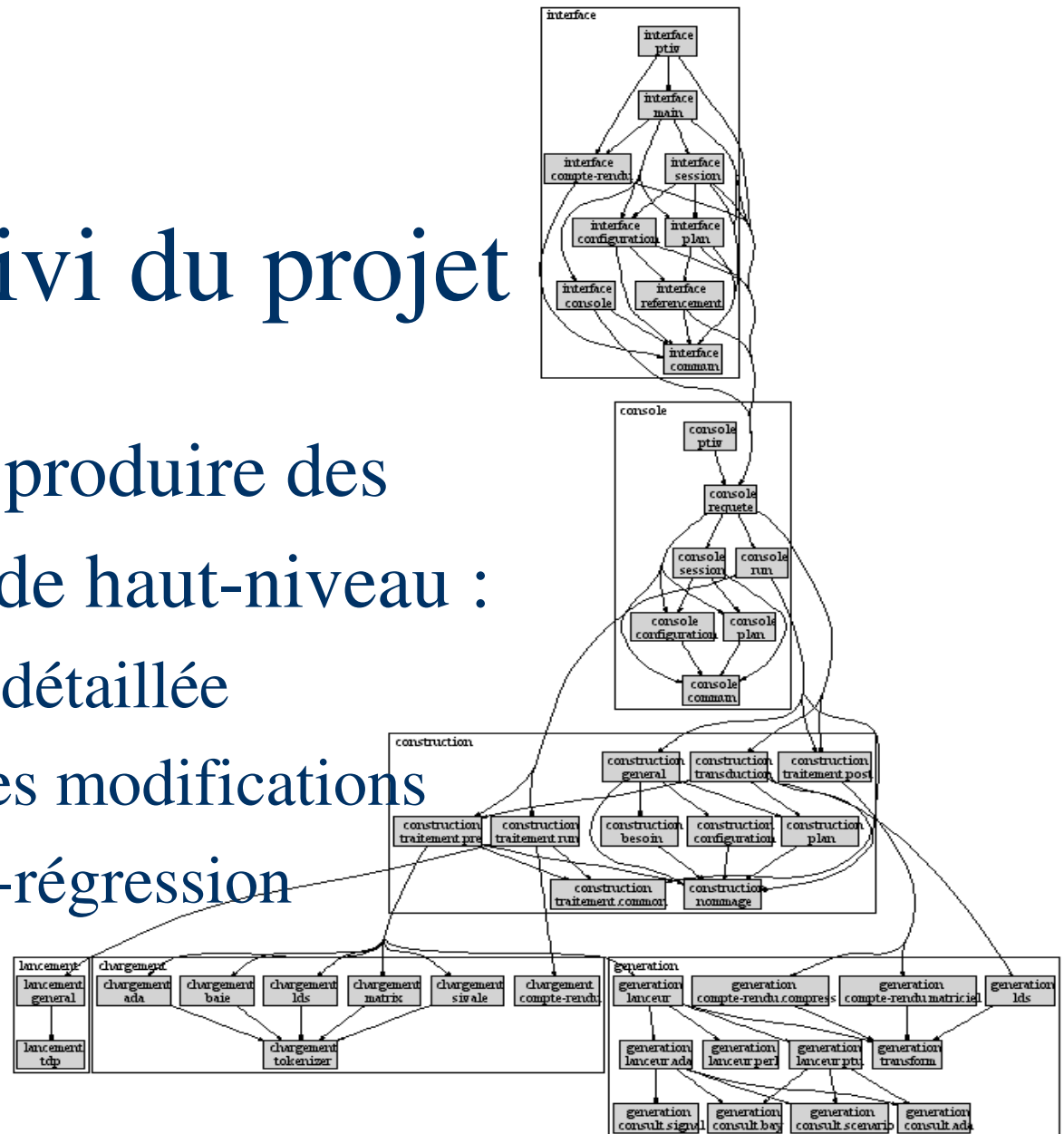
Cycle de vie simplifié



Suivi du projet

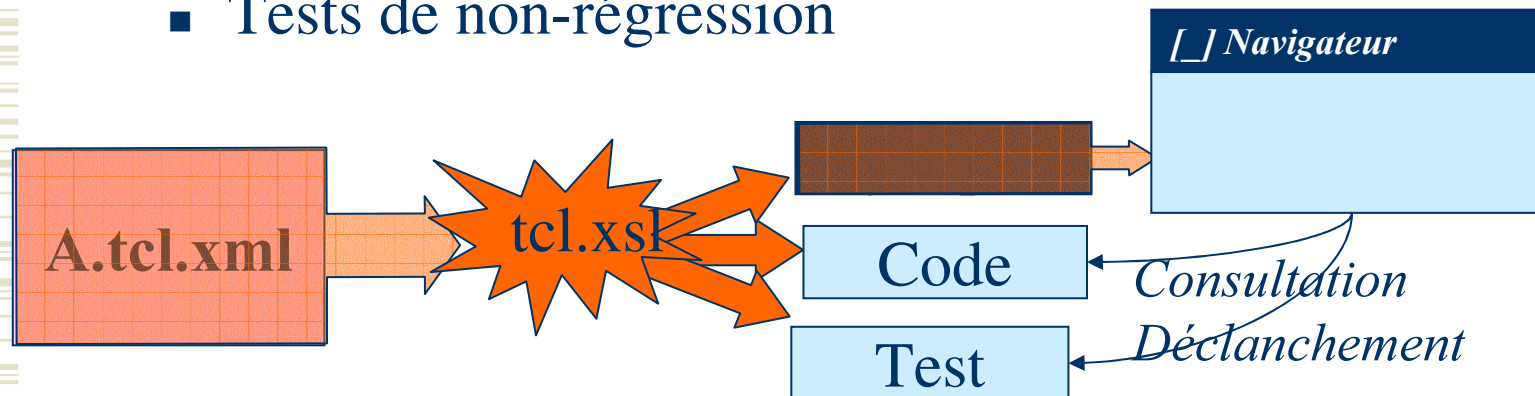
◆ Possibilité de produire des représentations de haut-niveau :

- Architecture détaillée
- Historique des modifications
- Bilan de non-régression



Gestion des tests

- ◆ Codage des tests avec le code
 - Génération conjointe
 - Tests de non-régression



- ◆ Pas de tests reproductibles d'interface

Bilan du projet PTIV

- ◆ Gains de productivité (facteur 4 en remontée)
 - *Redondances minimisées*
 - Coopération aidée par le suivi
 - Capacité de définir de nouveaux point de vues en ligne.
- ◆ Gains de qualité
 - *Cohérence garantie structurellement*
 - Uniformité des produits
 - Automatisation de la non-régression

Plan

- ◆ Présentation de la problématique
- ◆ Les quatre opérations
- ◆ Expérience acquise
- ◆ Développements prévus
 - Formalisation
 - Comparaisons
 - Tests d'interfaces
 - Passé nominal

Formalisation

- ◆ Contrat ANVAR de recherche et développement
- ◆ Rédaction d'une spécification technique du besoin
- ◆ Rédaction d'une architecture



Comparaison



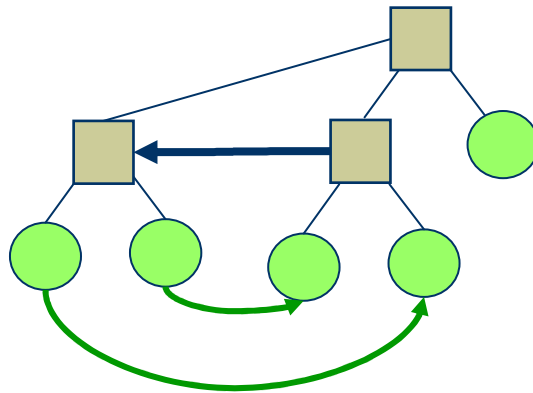
- ◆ Intégration d'un moteur de comparaison XML
 - Production de documents .pdf dénotant l'évolution par rapport à la version antérieure (barre de modifications)
 - Automatisation de l'historique des codes produits.

Stratégie de tests

- ◆ Intégration d'un graphe états/événements
 - Développement d'une méthode de tests d'interfaces reproductibles.
 - Inférence sur une description de haut-niveau de scénarios de tests couvrants.
 - Eliminer les redondances dans les tests de fonctions
 - Identification d'une notion de «passé nominal»

Passé nominal

- ◆ Les redondances dans les préluces de test (\approx états) sont couvert par une notion de « passé nominal ».



- ◆ Jeu de test d'intégration = Chemins vers un état
- ◆ Jeu de test de validation = Parcours couvrants