

## Les outils qui nous veulent du bien

*Plus de temps pour développer en LabVIEW*



Rhône-Alpes LUG :  
Rencontre développeurs 04/07/2013

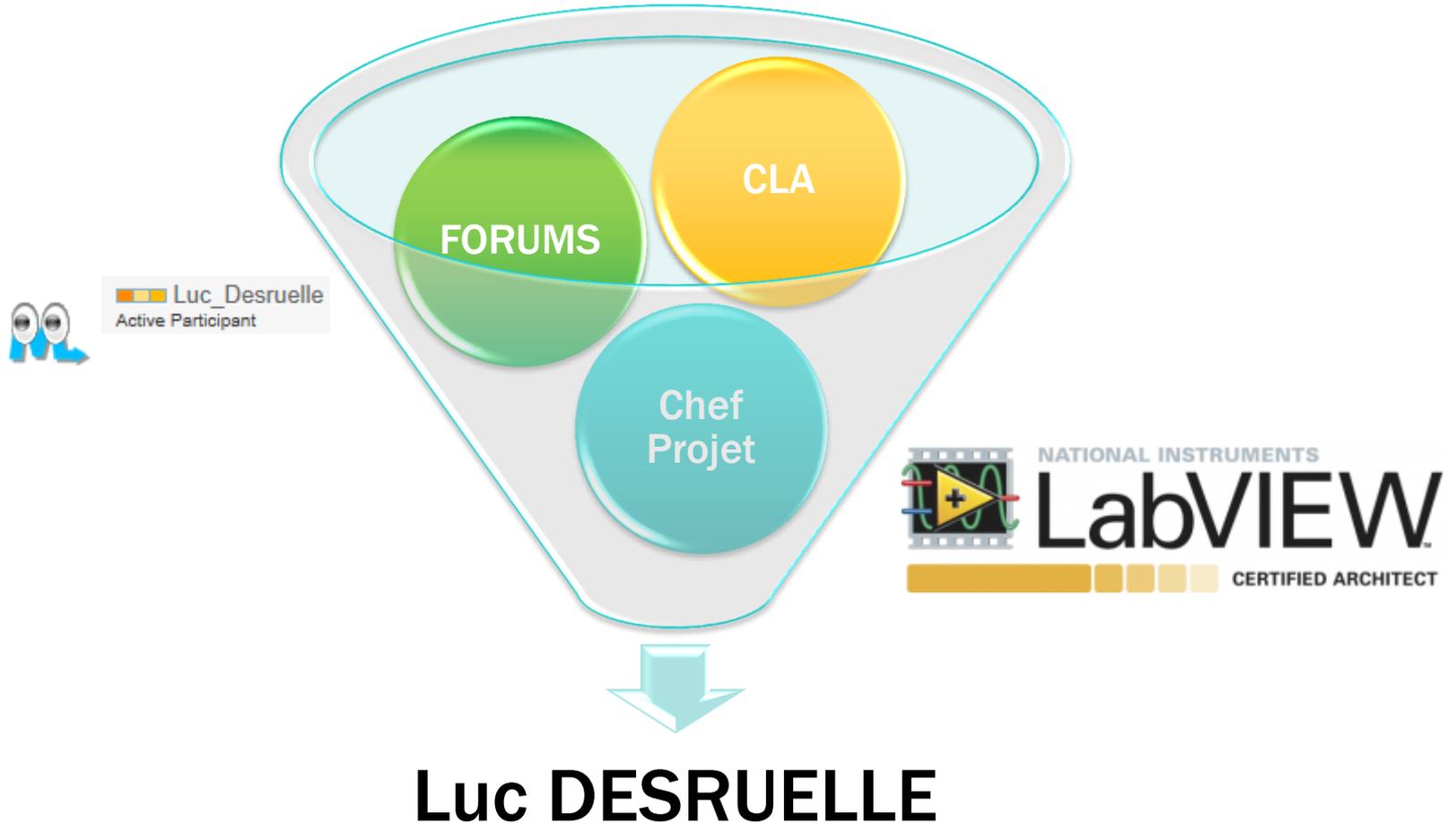
- ❑ Activité : Développement logiciel test et mesure
- ❑ Compétences : LabVIEW (Windows, RT, DSC, PDA, FPGA), TestStand
- ❑ Localisation : Grenoble (Moirans, 38)
- ❑ Partenaire National Instruments (2001)
- ❑ Développeurs certifiés LabVIEW et TestStand
- ❑ [www.mesulog.fr](http://www.mesulog.fr) (exemples de réalisation)



### □ Ils nous ont fait confiance :

- AREVA T&D
- ALSTOM Grid
- ATMEL
- BEAMIND
- CEA (Commissariat à l'Énergie Atomique)
- CETIAT (Centre Technique des Industries Aéronautiques et Thermiques)
- CNES (Centre National d'Études Spatiales)
- CNRS (Centre National de la Recherche Scientifique)
- EDF
- HONEYWELL Security
- LNE (Laboratoire National d'Essais)
- NIST (National Institute of Standards and Technology)
- NMIJ (National Metrology Institute of Japan)
- PECHINEY
- RADIALL
- STMicroelectronics
- THALES
- VALEO
- ...





## Le constat



PLUS DE TEMPS POUR DEVELOPPER EN LOBVIEN

## Gestion projet : Juste avec

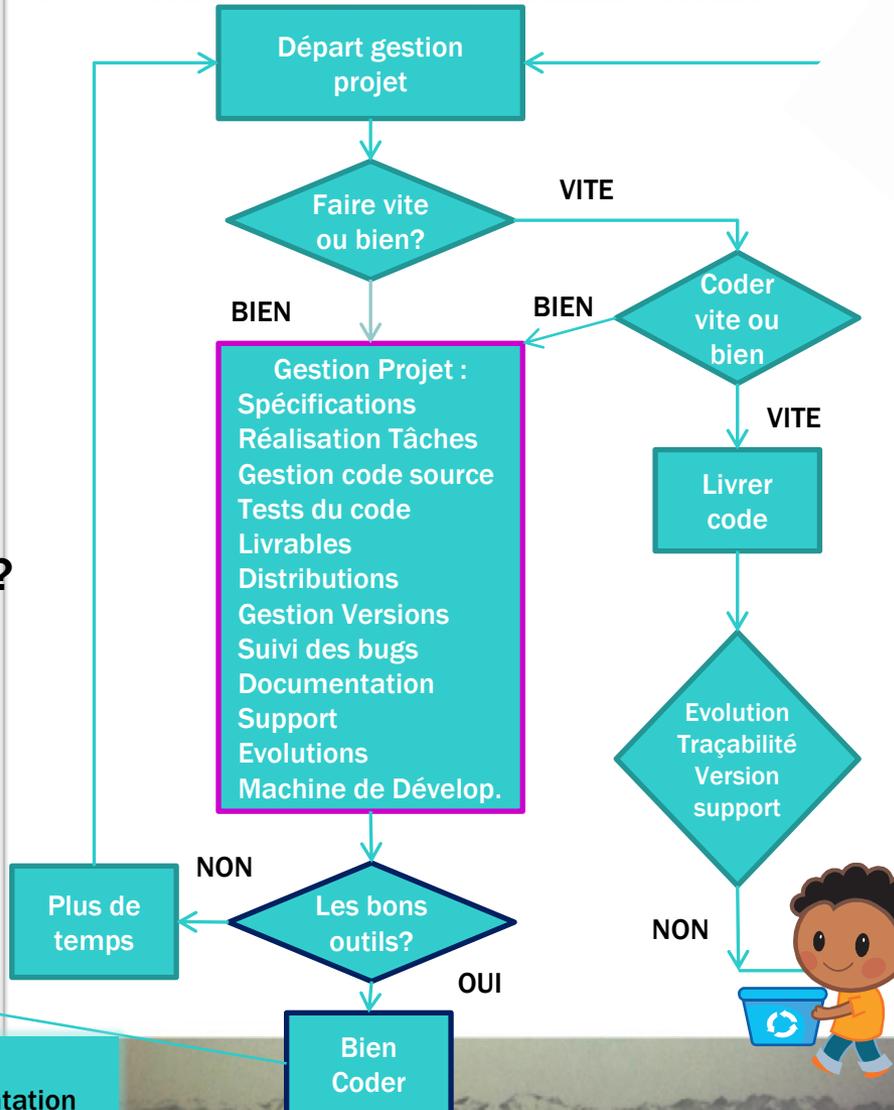
➤ un bloc notes

et

➤ un stylo MESULOG?



Autre  
présentation



Notre métier : chef de projet – développeur... LabVIEW

Mais...

Combien de temps passez-vous à coder par semaine?

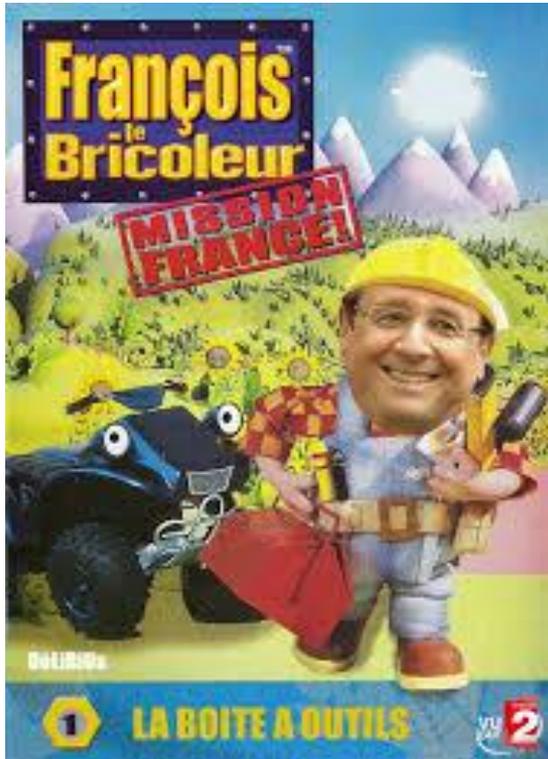
70% : Technicien Développeur  
30% : Chef de projet  
10% - 20% : Architecte LabVIEW  
Pas beaucoup de temps pour coder



- ❑ Discuter – Partager - Réfléchir



- ❑ Une *vrai* boîte à outils pour trouver plus de temps pour coder



- ✓ Ne pas savoir où trouver de l'aide
- ✓ Pas de logiciel de gestion version
- ✓ Pas de centralisation de l'information
- ✓ Pas d'Analyse - ni Test du code source
- ✓ Pas de Planning - Gestion des bugs – Documentation
- ✓ Attendre la fin du projet pour générer l'exécutable
- ✓ Pas de gestion de la machine de développement



- ❑ L'histoire dont vous êtes le héros
- ❑ Où trouver de l'aide?
- ❑ Zoom sur : logiciel de gestion de code source
- ❑ FORGE (?)
- ❑ Logiciels d'analyse et test de National Instruments
- ❑ Autres outils *dans la boîte à outils*
- ❑ Zoom sur : Machine Virtuelle



### □ La communauté travaille pour vous



- 2 Forums National Instruments Francophone

- <http://forums.ni.com/t5/Discussions-au-sujet-de-NI/bd-p/4171>
- <http://forums.ni.com/t5/Discussions-au-sujet-des-autres/bd-p/4170>

- 1 communauté Francophone

- <https://decibel.ni.com/content/community/regional/france/labview>
- Rencontre développeurs

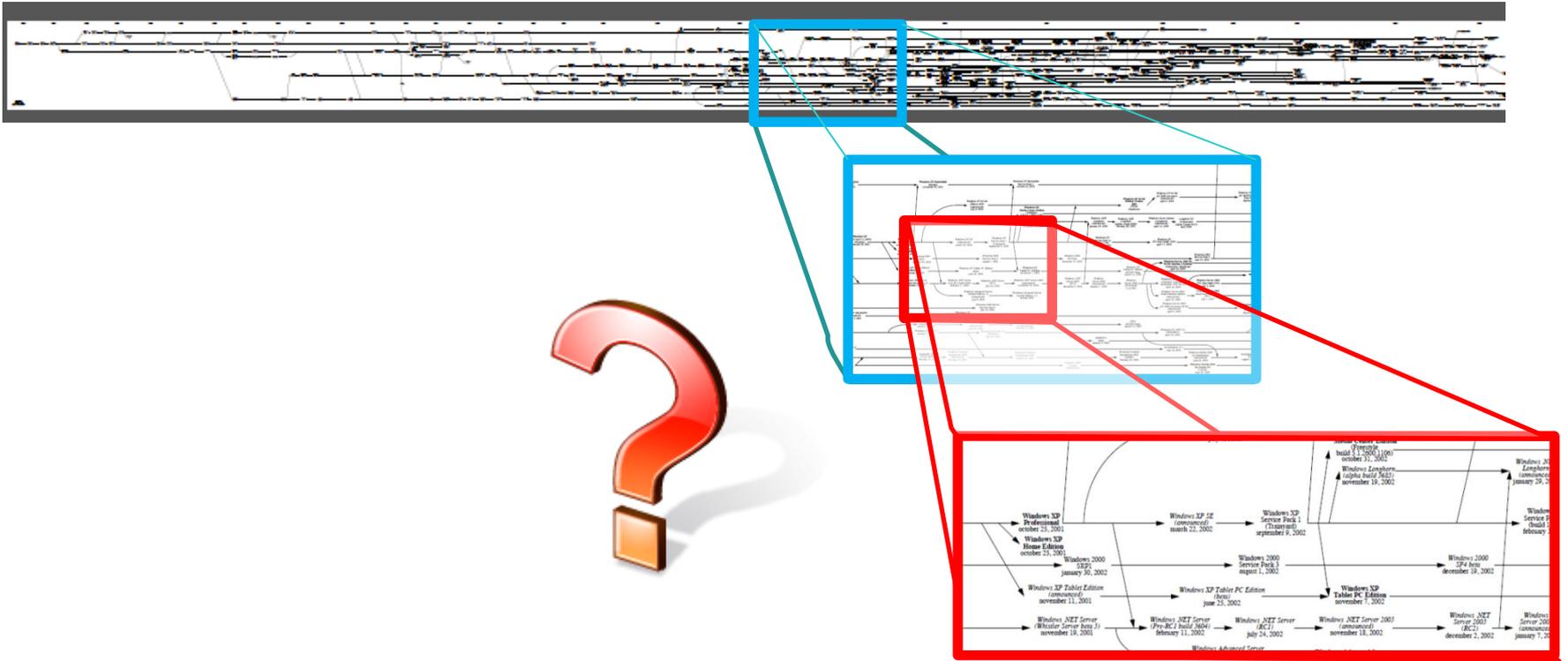
- Mais aussi... l'ensemble des forums et communautés de NI

- [ni.com/largeapps](http://ni.com/largeapps)
- [LabVIEW Development Best Practices](http://ni.com/largeapps)



- ❑ Des articles sur le site de NI
  - « Le logiciel ne se limite pas au code » *de l'équipe JKI* :
    - <http://zone.ni.com/devzone/cda/pub/p/id/1743>
  - Communautés de NI
    - [LabVIEW Development Best Practices](#)
    - [ni.com/largeapps](#)
    - .....
  
- ❑ Gestion projet chez MESULOG

## Logiciel de gestion de code source



- ❑ Le contrôle du code source est l'un des aspects les plus élémentaires du développement logiciel professionnel
- ❑ Mais aussi trop négligé

Qui utilise un logiciel de gestion code source?

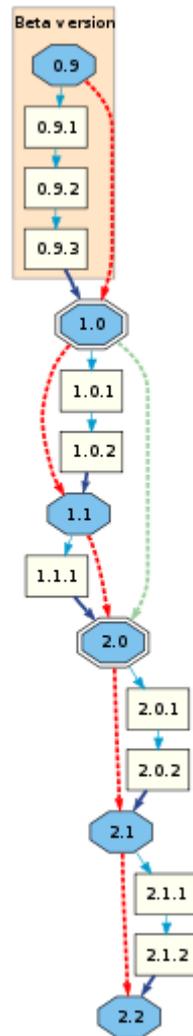
Lequel?

Mais quelle utilité??

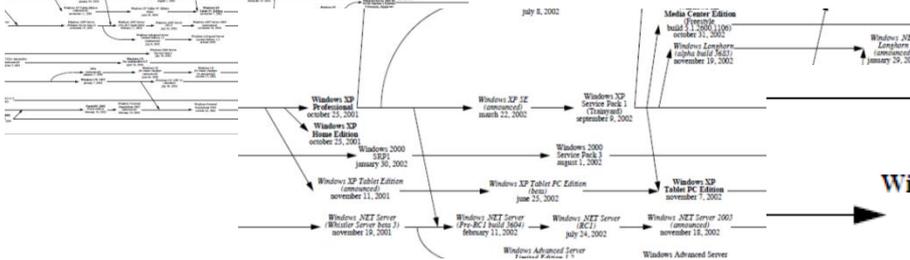
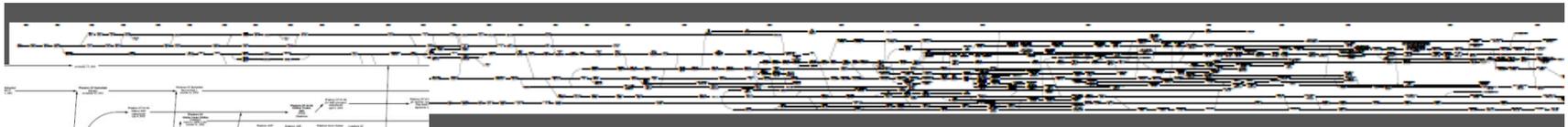
[http://fr.wikipedia.org/wiki/Version\\_d%27un\\_logiciel](http://fr.wikipedia.org/wiki/Version_d%27un_logiciel)

- ❑ Les logiciels évoluent = versions du logiciel
- ❑ Chaque étape d'avancement = version
- ❑ Les différentes versions sont liées à un ensemble de modifications de fichiers
- ❑ « *révision* » d'un fichier VS la « *version* » d'un logiciel.

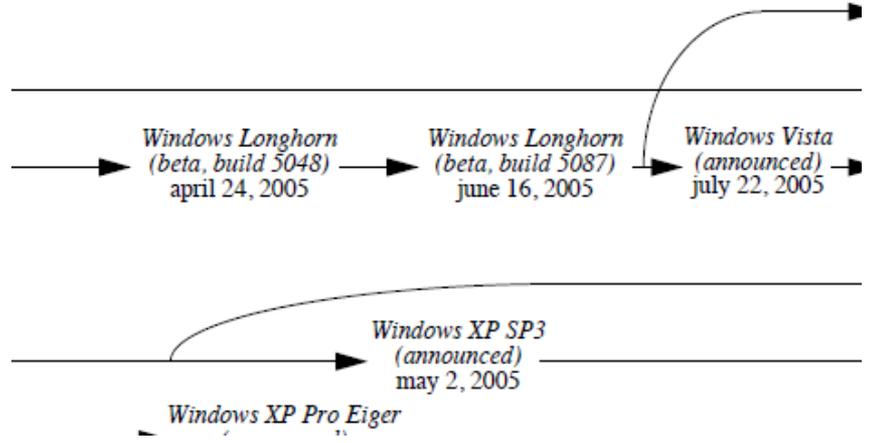
Cette activité étant fastidieuse et complexe, un appui logiciel est presque indispensable.



## Microsoft Windows



**Windows XP Professional  
x64 Edition  
april 25, 2005**



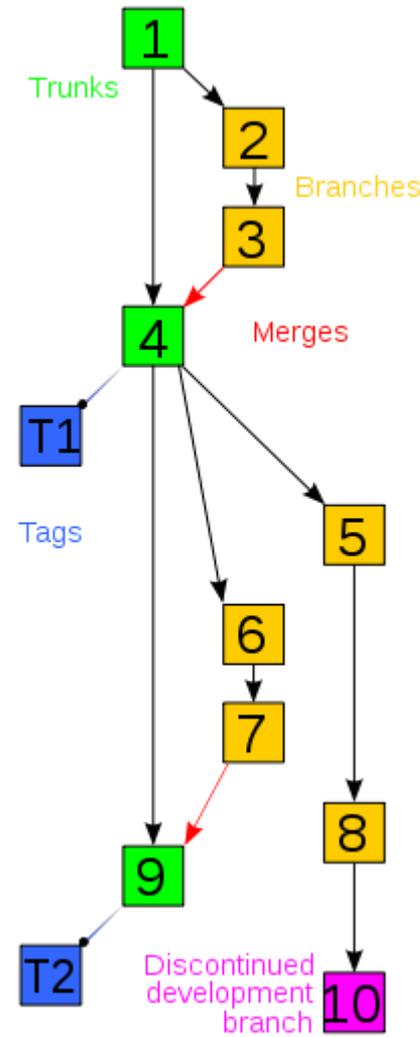
## Un logiciel de gestion de code source pourquoi?

### Pour répondre à :

- ❑ Comment faire pour gérer les révisions du code source ?
- ❑ Comment gérer les versions des exécutables ?
- ❑ Comment travailler à plusieurs développeurs ?
- ❑ Comment faire des tests et les conserver ?
- ❑ Comment revenir sur une ancienne version?
- ❑ ....

- ❑ **Source Code Control : SCC**
- ❑ Agit sur une arborescence de fichiers afin de :
  - Conserver toutes les révisions des fichiers (Historique)
  - Ajouter un commentaire pour chaque livraison.
  - Identifier & retrouver le code source des différentes versions du logiciel.
  - Aider aux développements multi-développeur (fusion de copies locale et distante, gestion conflits)

*... beaucoup d'autres fonctionnalités*

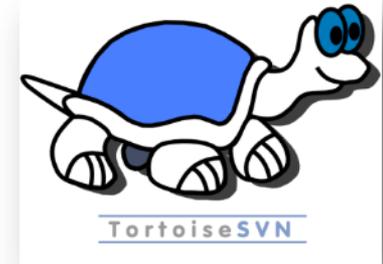


- ❑ 1972 Source Code Control System (SCCS) est un système de gestion de versions pour les Systèmes d'IBM .
- ❑ Puis RCS, CVS, puis Subversion, Mercurial, Git,...

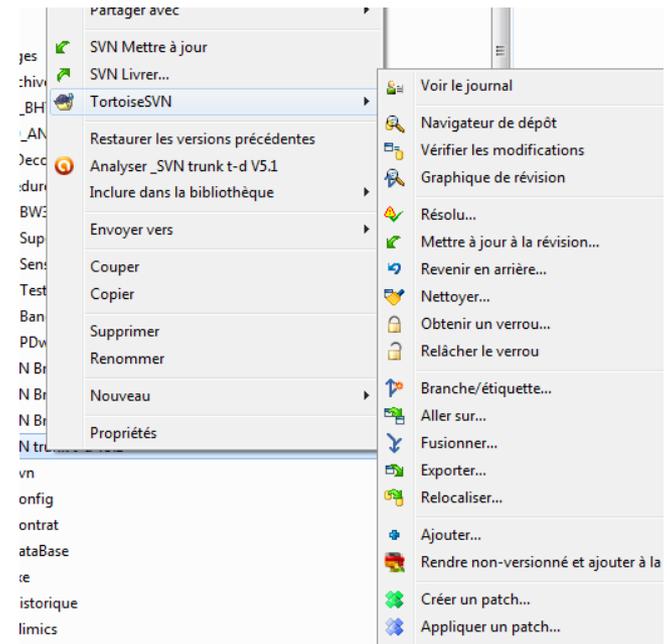
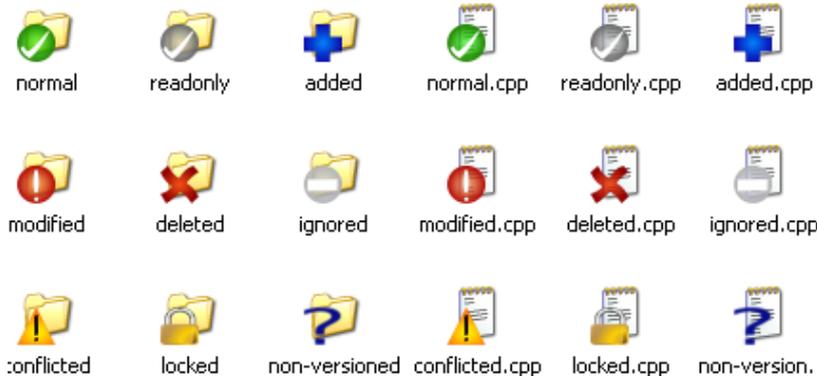
Logiciel de gestion de versions		
Gestion locale	Logiciels libres	GNU RCS (1982) GNU CSSC
	Logiciels propriétaires	PVCS (1985)
	D'importance historique	SCCS (1972) CVS (1990) CVSNT (1992) <b>SVN (2000)</b>
Client-serveur	<b>Logiciels libres</b>	
	Logiciels propriétaires	Rational ClearCase (1992) CCC/Harvest (Années 70) CMVC (1994) Visual SourceSafe (1994) <b>Perforce (1995)</b> AccuRev SCM (2002) Sourceanywhere (2003) Rational Synergy (2006)
Décentralisé	Logiciels libres	GNU arch (2001) Darcs (2002) DCVS (2002) SVK (2003) Monotone (2003) Codeville (2005) Git (2005) <b>Mercurial (2005)</b> Bazaar (2005) Fossil (2007) Veracity (2011)
		Logiciels propriétaires



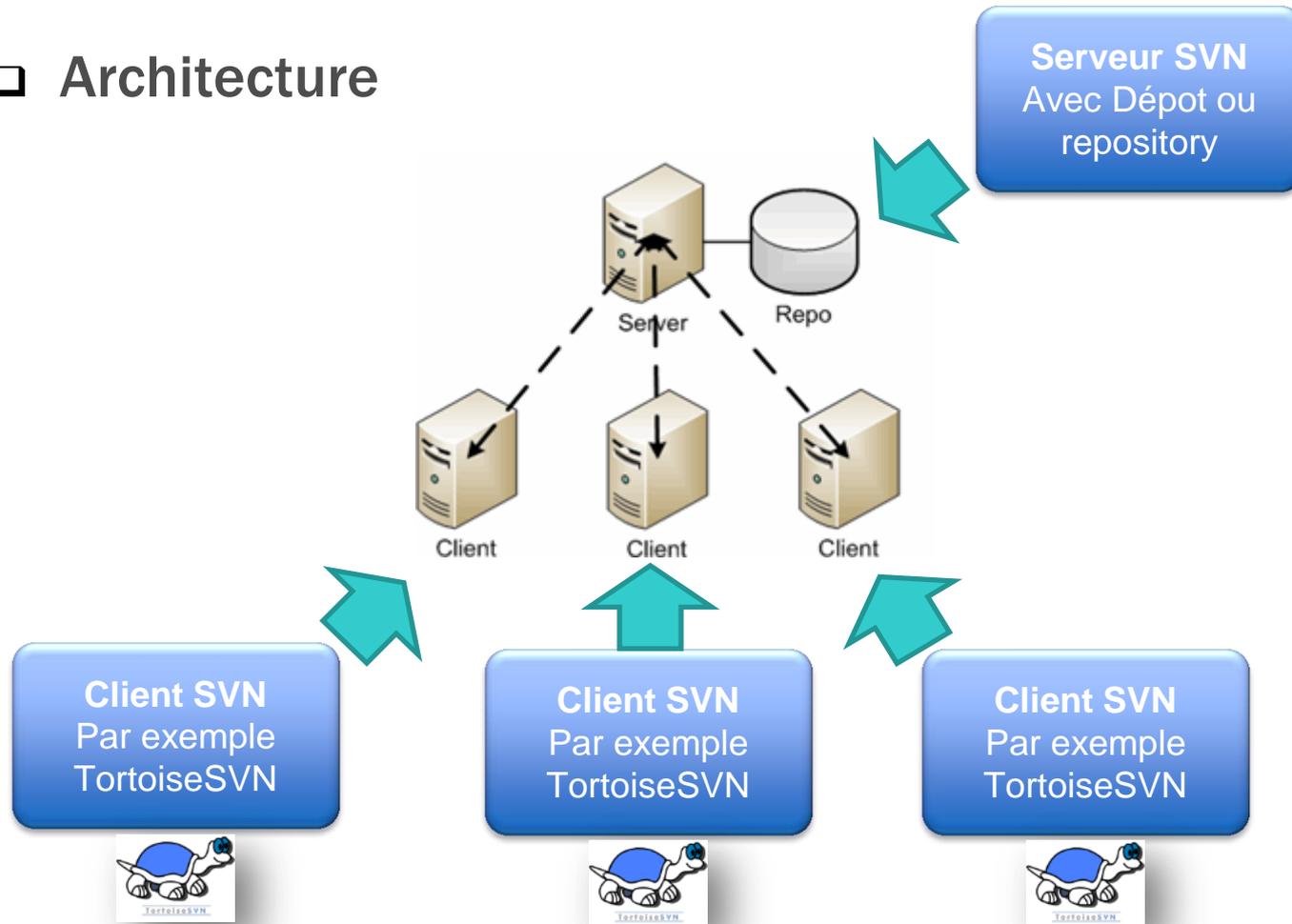
- ❑ **Subversion (en abrégé svn)**
- ❑ **Mode client-serveur, avec :**
  - **Un Serveur unique où se situent :**
    - les fichiers constituant la référence (le 'dépôt' ou 'référentiel')
    - un logiciel serveur Subversion tournant en 'tâche de fond'
  - **Des postes clients sur lesquels se trouvent :**
    - les fichiers copiés depuis le serveur, éventuellement modifiés localement depuis,
    - un logiciel client permettant la synchronisation entre chaque client et le serveur de référence

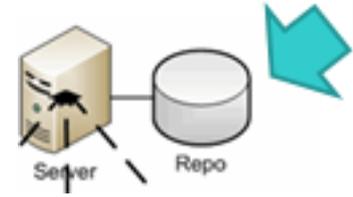


- ❑ Son nom vient de l'anglais *Tortoise*, « Tortue »
- ❑ Logiciel Libre
- ❑ S'intègre à l'explorateur de Windows :
  - Superposition d'icône aux répertoires et fichiers permettant de visualiser instantanément l'état
  - « Menu contextuel » des actions

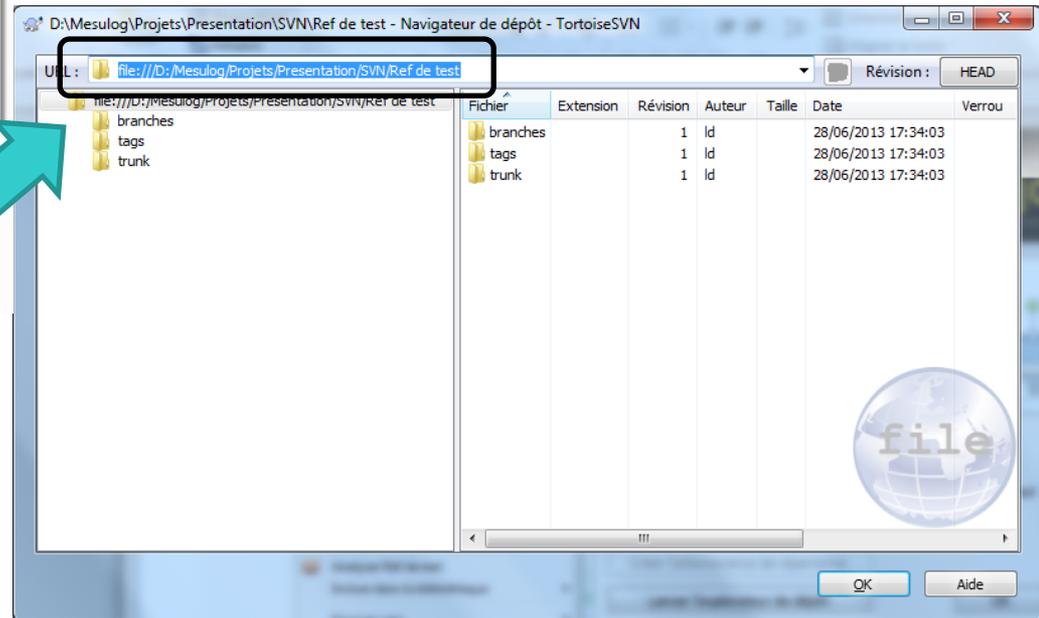
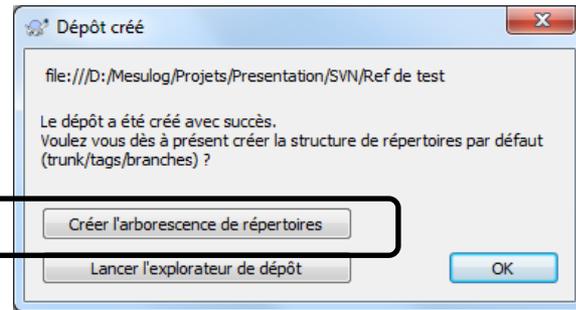
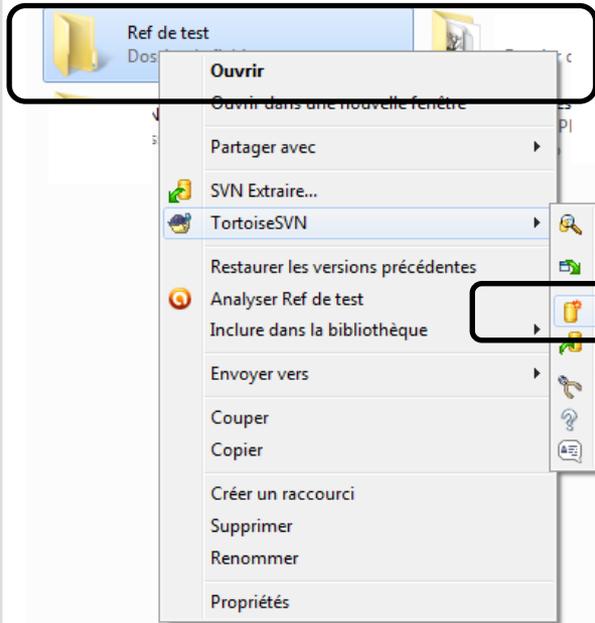


## Architecture



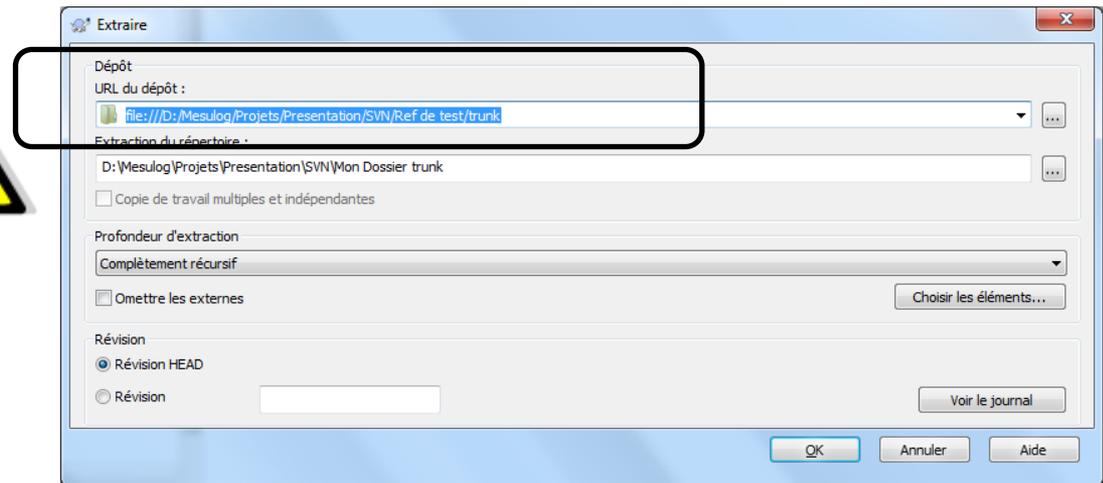
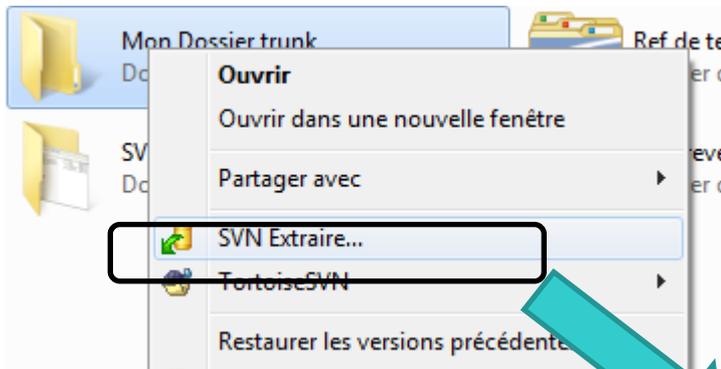


## ❑ Création du dépôt sur le serveur



Chemin du dépôt :  
Trunk  
Tags  
branches

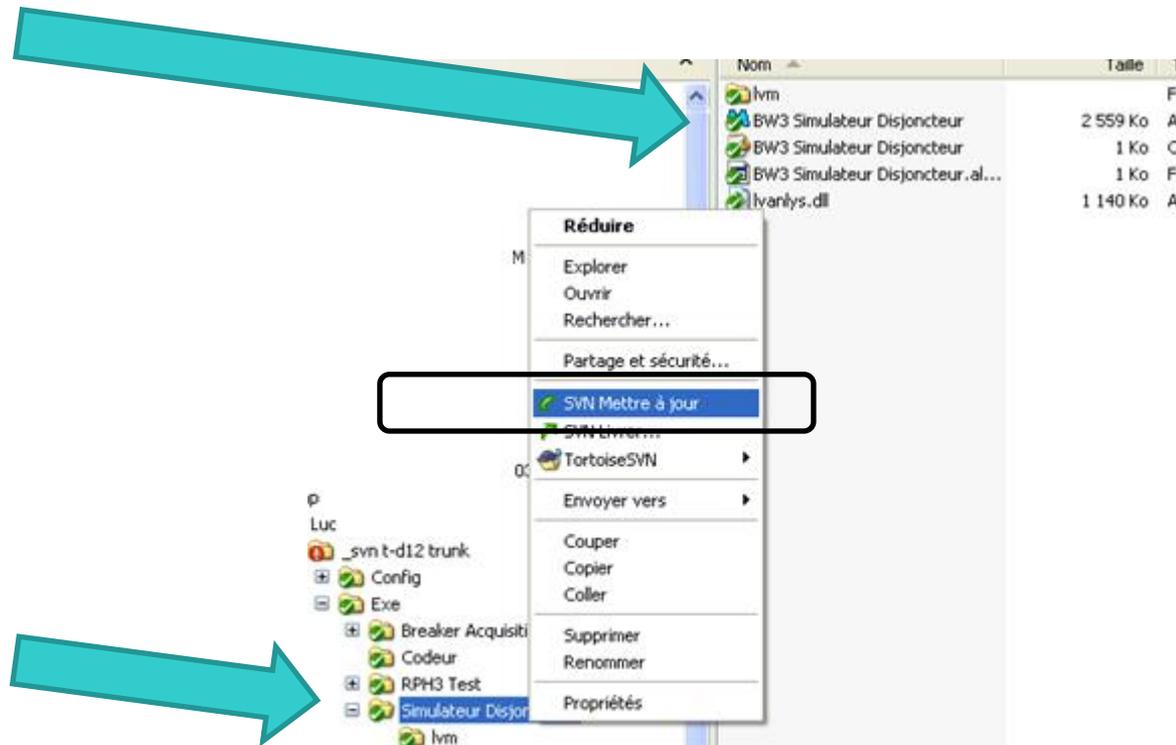
## ❑ Exemple : Extraire sur un client



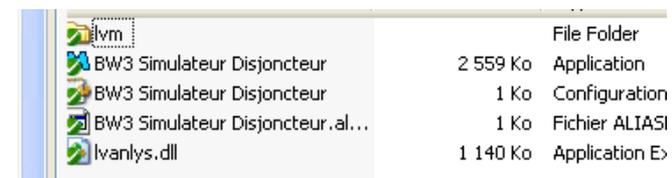
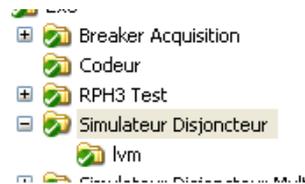
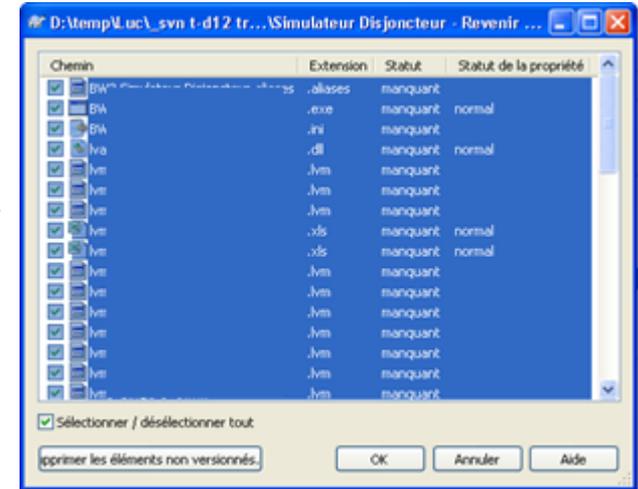
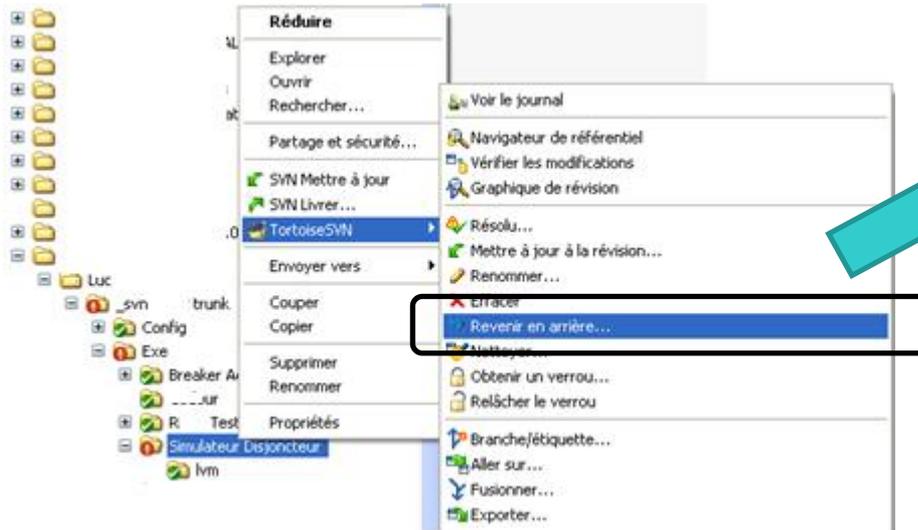
## ❑ Exemple : Commit ou livraison par un des clients

The image shows the TortoiseSVN 'Livrer' (Commit) dialog box. The 'Commentaire' field contains the text: "Evolution simulateur avec DAOx Clean". Below the comment field, the 'Chemin' (Path) list shows two files selected with checkmarks: "BW3 Simulateur Disjoncteur.alias" and "BW3 Simulateur Disjoncteur.exe". The dialog also includes options for "Voir les fichiers non versionnés", "Afficher les références externes d'autres référentiels", "Garder les verrous", and "Conserver les listes de changements". Buttons for "OK", "Annuler", and "Aide" are at the bottom.

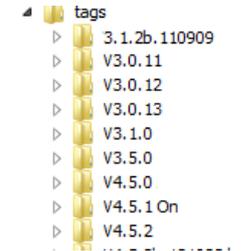
- ❑ Exemple : Mettre à jour ou Update par un autre client



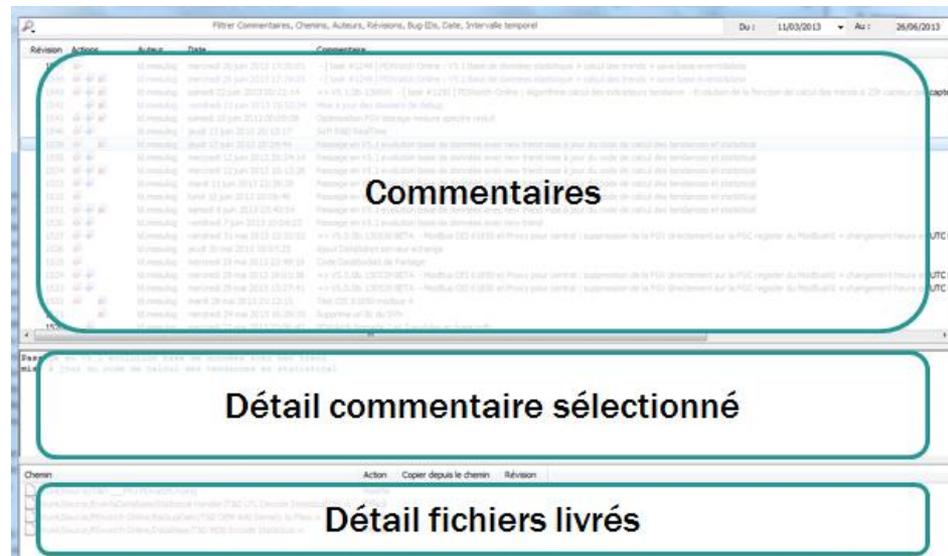
## Exemple : revenir en arrière ou Revert

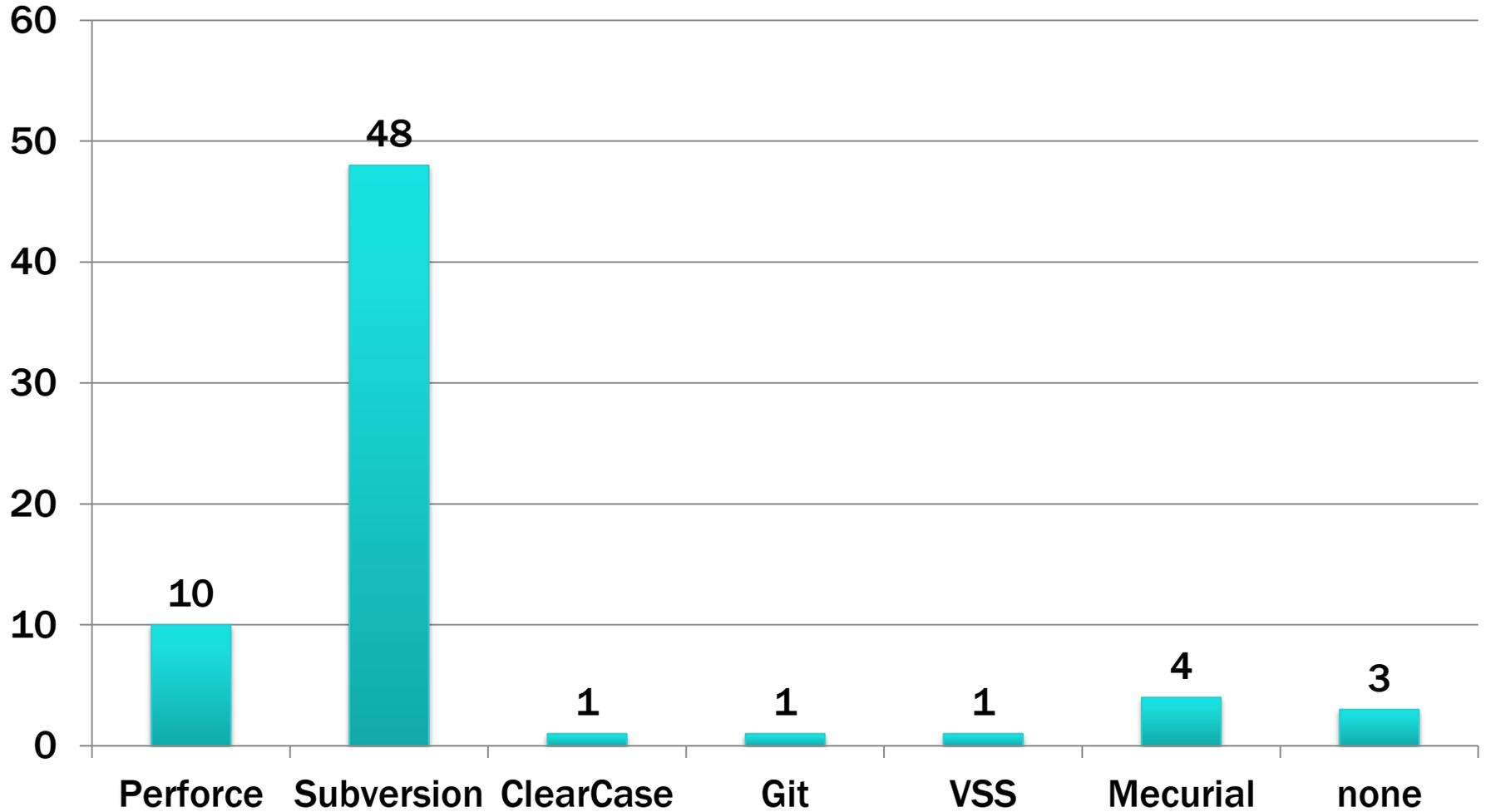


❑ Exemple : Tag ou étiquette : retrouver une version



❑ Exemple : journal





Source: 2010 [ni.com/largeapps](http://ni.com/largeapps) survey

## Native LabVIEW Integration

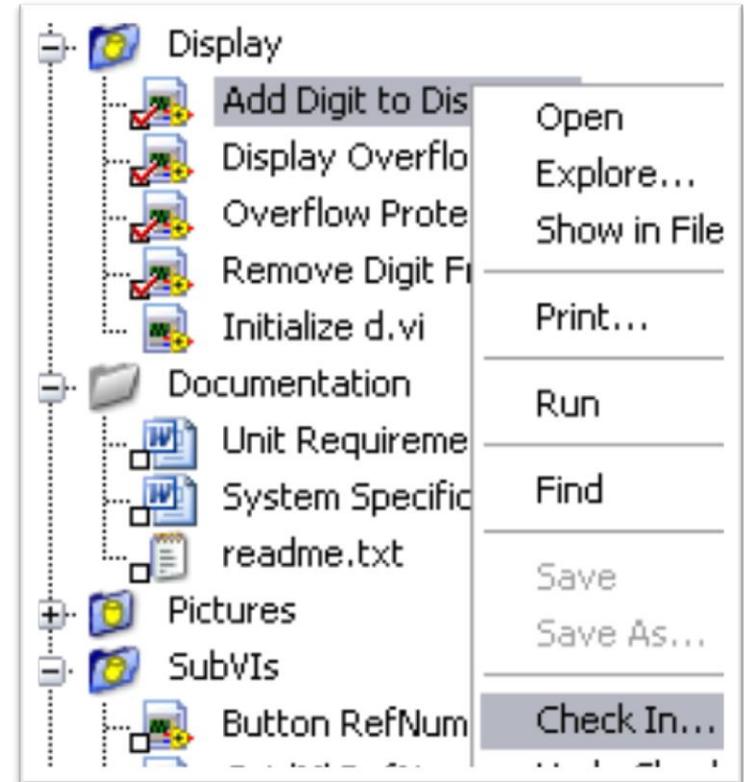
- ❑ Perforce

## Integration Through Standard API

- ❑ Microsoft Visual SourceSafe
- ❑ Microsoft Team System
- ❑ Rational ClearCase
- ❑ PCVS (Serena) Version Manager
- ❑ MKS Source Integrity
- ❑ Seapine Surround SCM
- ❑ Borland StarTeam
- ❑ Telelogic Synergy
- ❑ ionForge Evolution

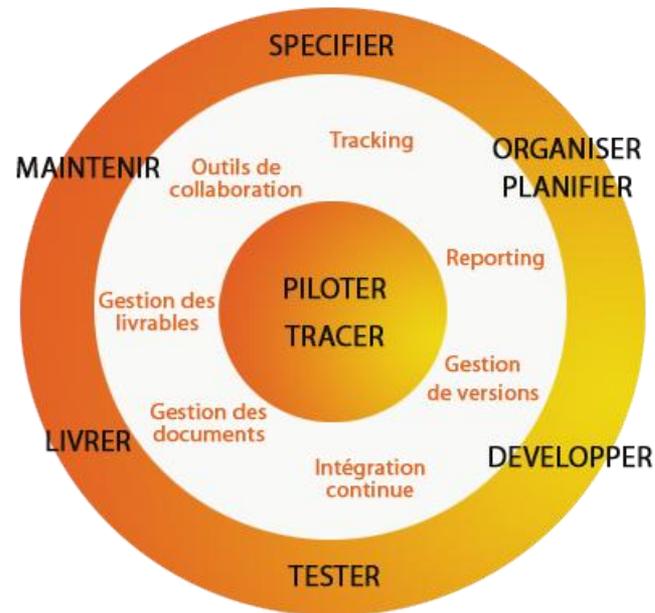
## Support through additional add-ons

- ❑ Subversion
- ❑ Mercurial



- ❑ Gestion utilisateurs – mot de passe
- ❑ Accès https
- ❑ Branches
- ❑ Merge
- ❑ Lock
- ❑ .....

# FORGE ?



## ❑ Bug tracker : outil de gestion de bug

- Identifier, Diffuser, sauvegarder
- Associer à une version de code
- Correctif code et distribution



## ❑ Quelques exemples

- FogBugz
- Mantis
- Bugzilla
- Tuleap



## ❑ Diffusion documentation

- Box (box.com)
- Tuleap



## ❑ Diffusion distribution

- Serveur FTP
- Tuleap



## ❑ Gestion des tâches

- Trello
- Tuleap

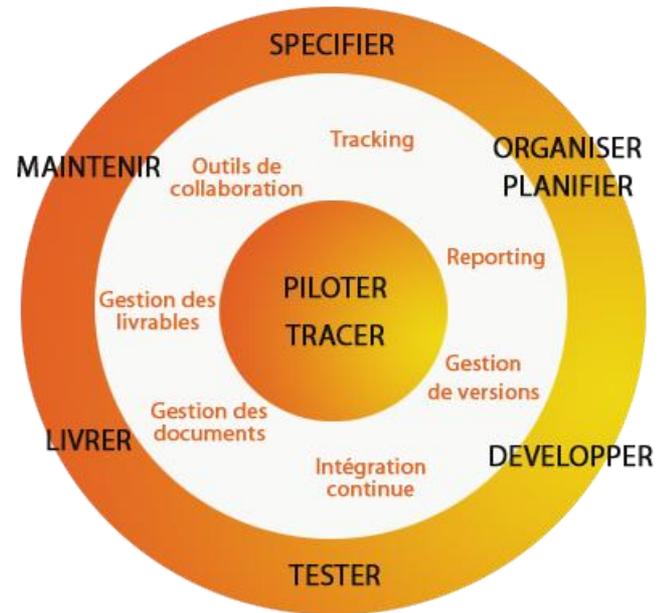
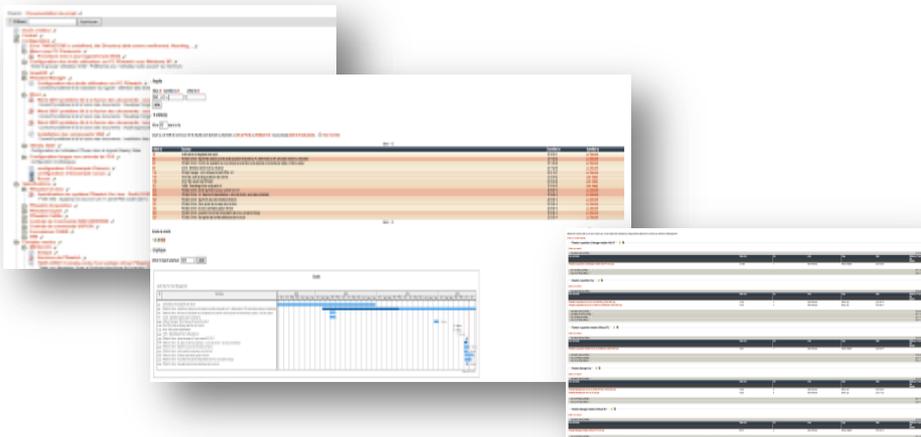
## ❑ Intégration Continue

- Jenkins



- ❑ Beaucoup de logiciels MAIS Logiciels séparés....
- ❑ FORGE?
  - Système qui gère « les développements collaboratif de logiciel »
  - Système simple et efficace
  - Regroupe les outils

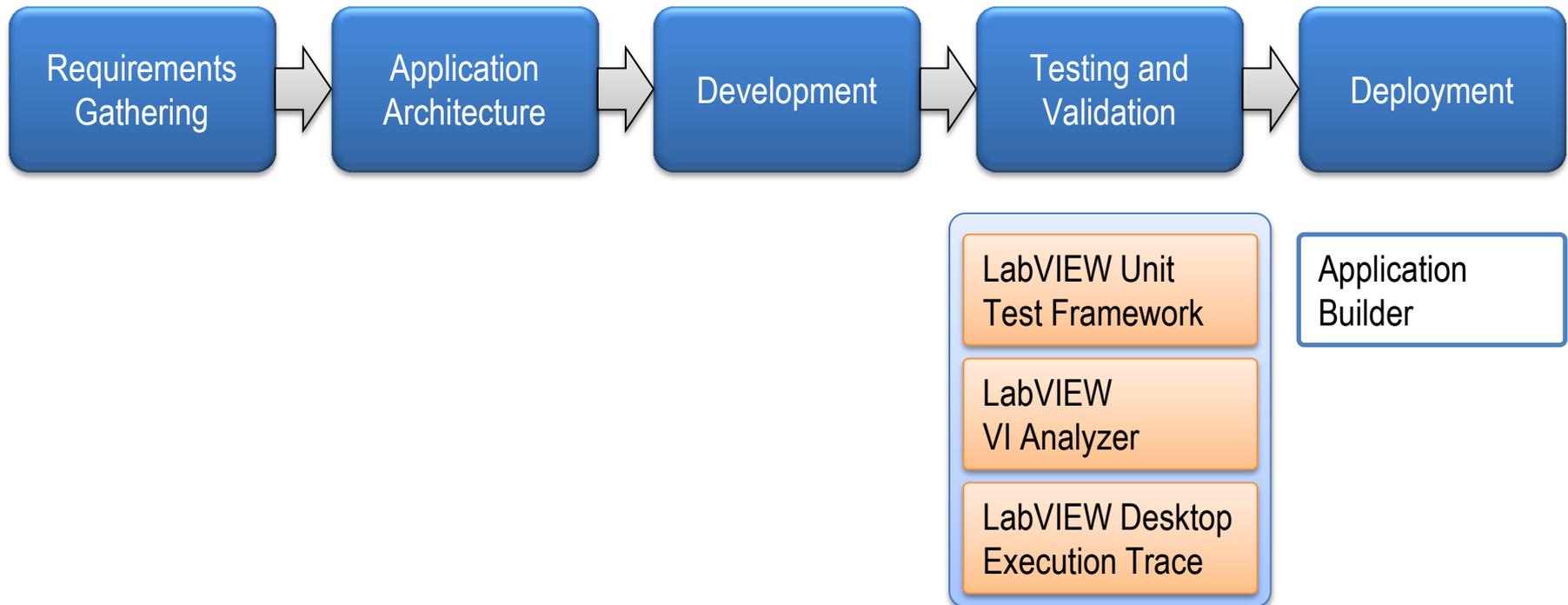
❑ TULEAP

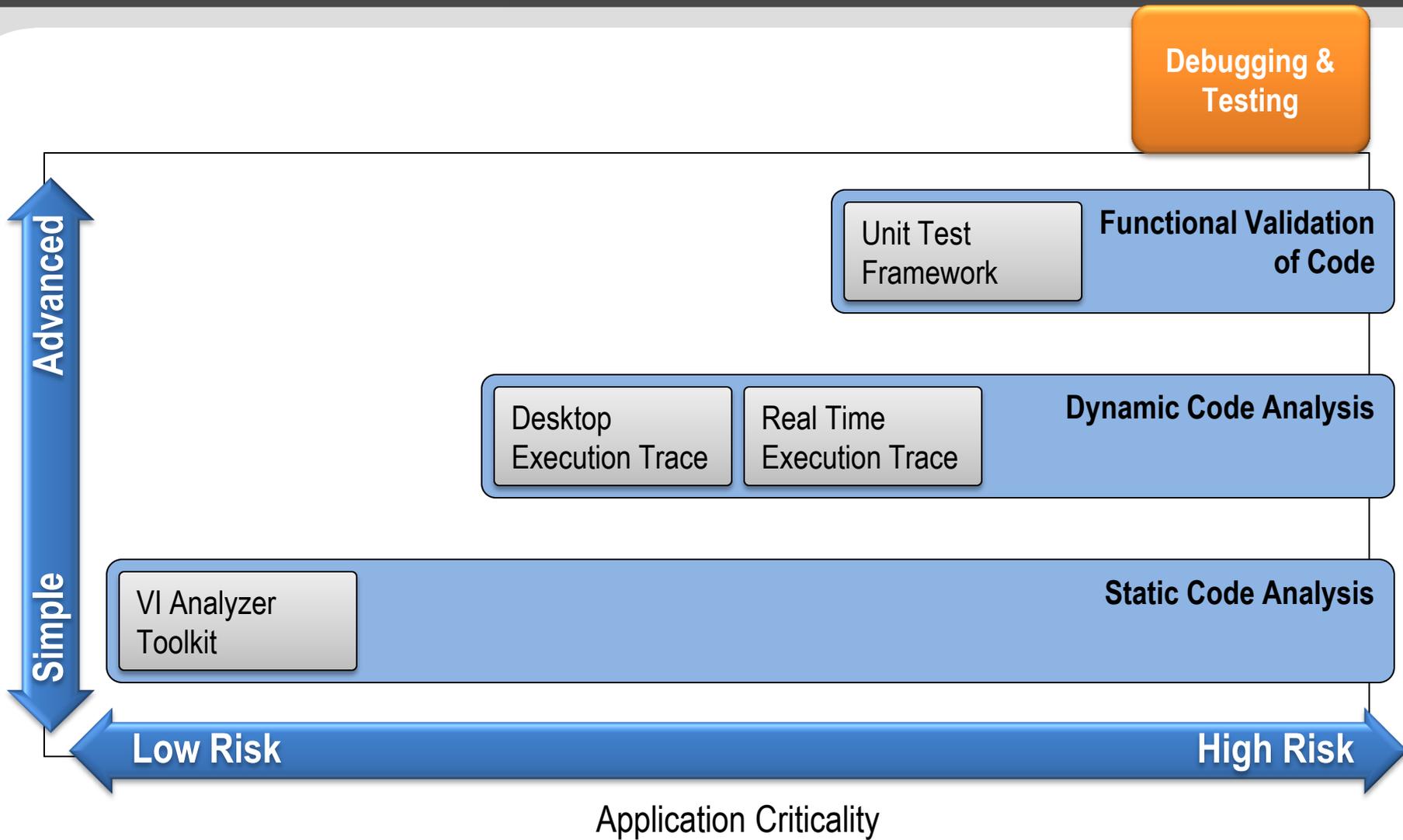


- ❑ **Mettre à disposition de tous les acteurs autorisés, l'ensemble des données du projet :**
  - Documentation (doc #ID)
  - Code source (gestion sous Subversion, https, snv #ID)
  - Exécutables, installeurs (release #ID unique)
  - Suivi de tâches (task #ID unique)
  - Suivi de bugs (bug #ID unique)
  - Espace de téléchargement FTP
  - Forums de discussion et wiki
  - Les commentaires SVN sont croisés avec les tags #ID unique des bugs - livrable - task. L'historique du projet est ainsi redoutable.

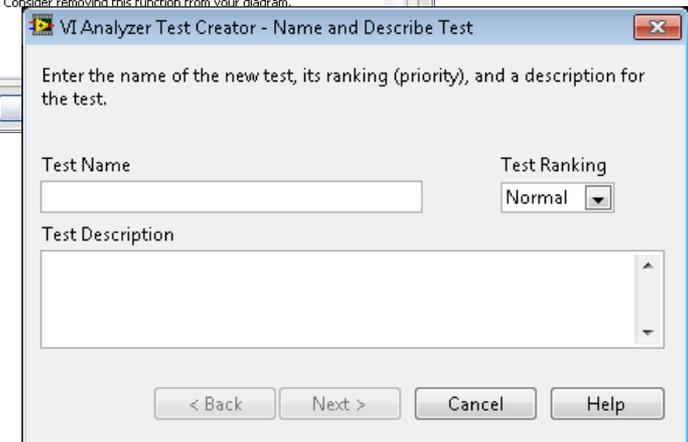
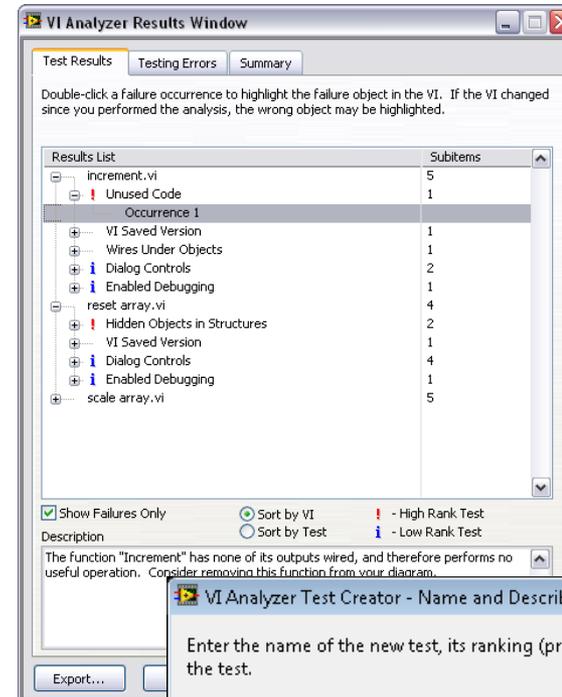
« La Suite ALM 100% Open-Source »

### Test et Analyse du code





- ❑ Automatise l'analyse du code via 80 fonctions
  - Performance
  - Style
  - Complexité
  
- ❑ Possibilité d'ajouter des fonctions de test (LV 2010)



## ❑ Fonctions de test

### List of Community VI Analyzer Tests

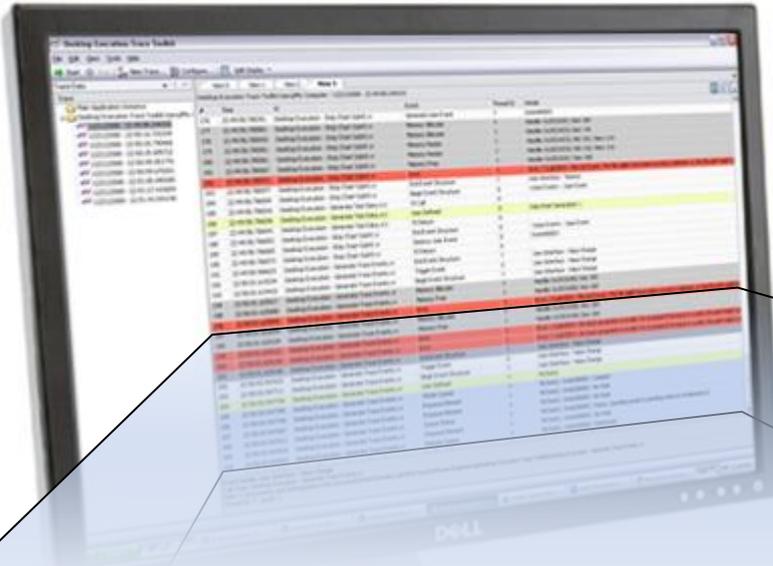
Créé le: 14 mars 2013 22:25 par [Jordan\\_M](#) - Dernier modifié: 13 juin 2013 22:36 par [warren\\_scott](#)

The following is a list of community-contributed VI Analyzer Tests:

-  Anti-Aliased Plots
-  Auto Error Handling Enabled
-  Case Sensitive String case structures
-  Check Captions and Labels Match
-  Check for Whitespace in FP names
-  Check VISA read/write Synchronous mode
-  Check whether Tunnels use default values if unwired
-  Checks whether references are closed
-  Copyright
-  Deprecated Functionality
-  Diagram Size and Position
-  Find Control
-  Find Function
-  Find Property or Method
-  Find Text
-  Find Typedef Instances
-  Merge Errors Style
-  Minimum Window Width
-  Panel Color
-  Property Node Name Format
-  Reentrancy
-  Separate Compiled
-  Subroutine and Not Reentrant
-  SubVI Owning Location
-  SubVIs Need Relinking
-  Unwired Event Nodes
-  View Cluster Constant as Icon
-  Custom Control / TypeDef Style

Feel free to add your own tests to this list as you post them to the LabVIEW community.

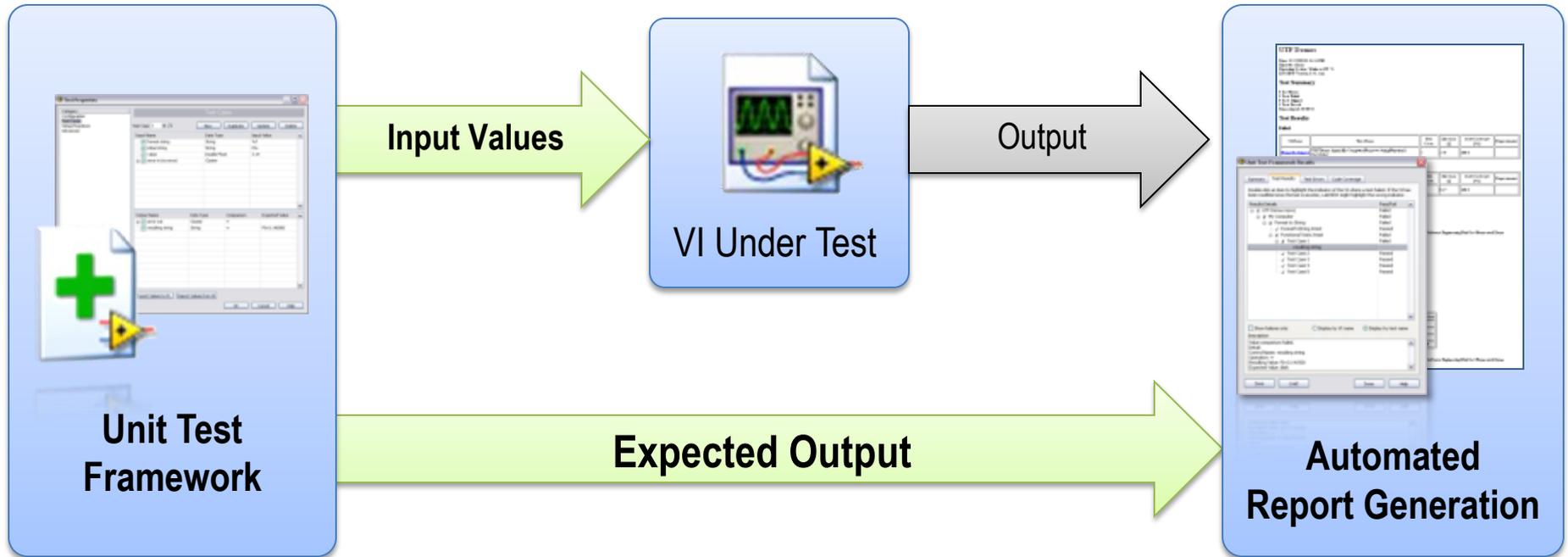
## ❑ <https://decibel.ni.com/content/docs/DOC-28198>



## Trace During Run-Time:

- Event Structures
- Memory Allocation
- Queues / Notifiers
- Reference Leaks
- Thread ID
- Unhandled Errors

Strip Chart SubVI.vi	Memory Resize	7	Handle: 0x25CA3C8; Old: 142; New: 118
Strip Chart SubVI.vi	Memory Allocate	7	Handle: 0x25CA3C8; Size: 142
Strip Chart SubVI.vi	Memory Resize	7	Handle: 0x25CA3C8; Old: 142; New: 118
Strip Chart SubVI.vi	Error	7	Error: 7 (LabVIEW: File not found. The file might have
Generate Trace Events.vi	User Defined	7	MyTestQ
Generate Trace Events.vi	Obtain Queue	7	MyTestQ - 0x66200002 : Created
Generate Trace Events.vi	Enqueue Element	7	MyTestQ - 0x66200002 : No Wait



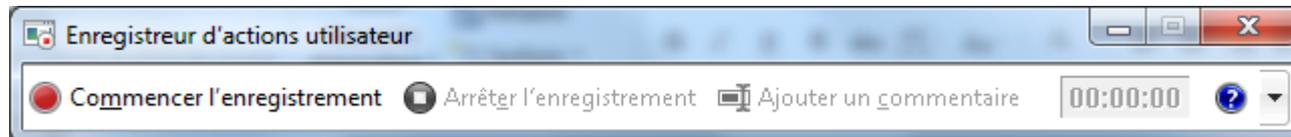
**Test vector = Input value(s) + Expected output(s)**





- ❑ L'enregistreur d'action utilisateur ... de Windows
- ❑ Fonction cachée de Windows 7.
- ❑ Enregistrer actions utilisateur à effectuer.
- ❑ Sauvegarde directement du descriptif des actions, images et commentaires dans un fichier .mht

Dans le menu Démarrer de Windows, tapez les lettres "PSR":



- ❑ Autoit
- ❑ production de logiciel utilitaire pour Microsoft Windows.
- ❑ Automatiser des tâches communes, comme le contrôle de site Web, le contrôle de réseau, la défragmentation de disques et la sauvegarde.
- ❑ Simuler des applications utilisateurs, par lequel un script Autoit *pilote* une application en lieu et place d'un contrôle manuel pendant le test d'un logiciel

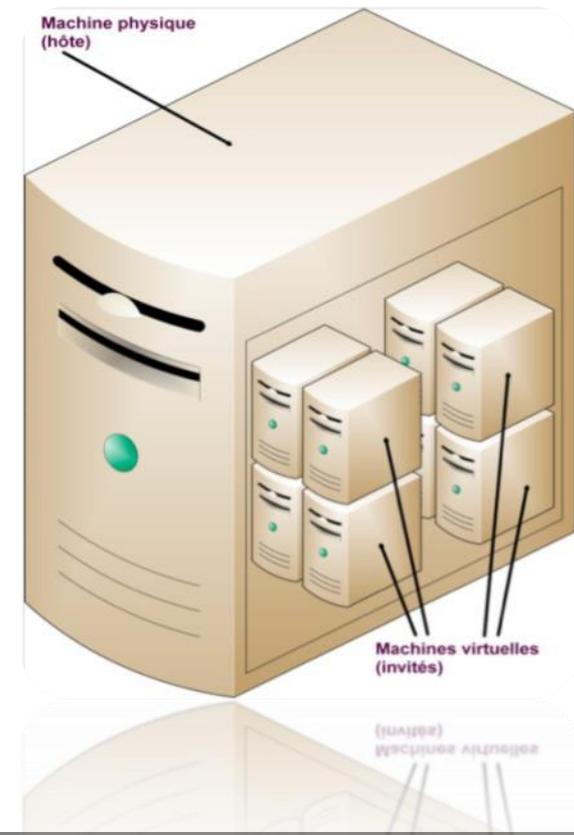


- ❑ NI Requirements Gateway : solution de traçabilité des exigences qui relie vos documents de développement et de vérification aux exigences formelles stockées dans les documents et les bases de données.

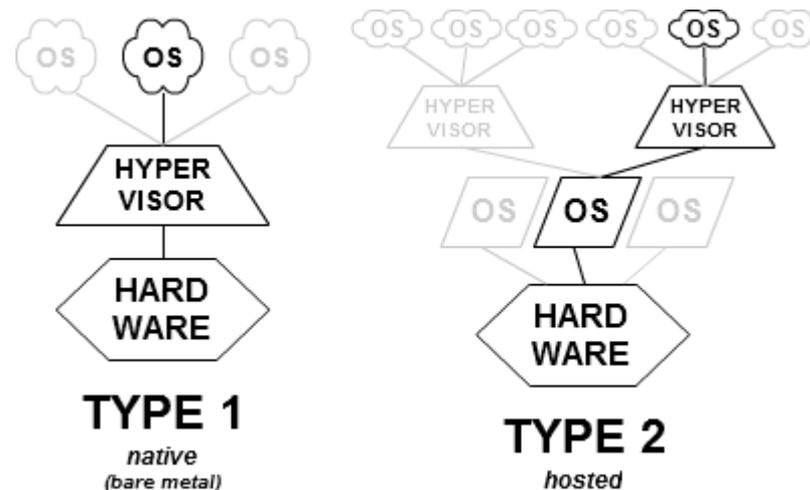
- ❑ Notepad ++
- ❑ FileZilla
- ❑ Axialis
- ❑ Copie Ecrans
- ❑ Jenkins : intégration continue



- ❑ *But : Virtualiser la plate-forme de développement*
- ❑ **Virtualisation** : simuler l'existence de plusieurs machines informatiques en utilisant une seule.
- ❑ **Machine virtuelle** illusion d'un appareil informatique créée par un logiciel d'émulation.
- ❑ Simule la présence de ressources matérielles et logicielles (mémoire, processeur, disque dur, système d'exploitation, pilotes)



- hyperviseur une plate-forme de virtualisation qui permet à plusieurs systèmes d'exploitation de travailler sur une même machine physique en même temps



- Type 2 : VMware, VirtualBox, VirtualPC, sont quelques logiciels de virtualisation

- ❑ Virtualisation des serveurs est le modèle le plus courant. Il s'agit de faire fonctionner plusieurs serveurs virtuels sur un ou plusieurs serveurs physiques (hôtes).
- ❑ OS moderne simule une machine qui exécute MS-DOS;
- ❑ Machine de développement : une machine virtuelle par plate-forme de développement (PDA, Windows, linux,..)



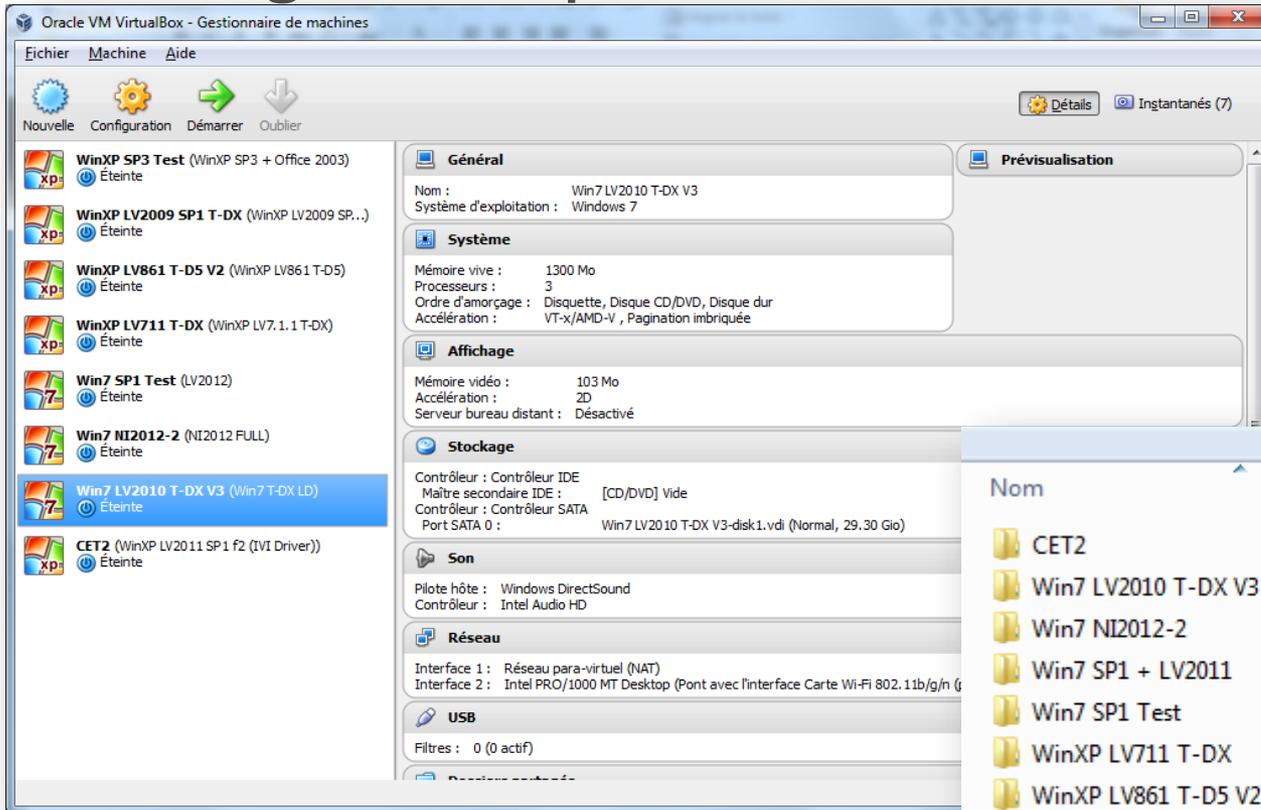
### ❑ Les avantages

- Sécurisation des plates-formes de développement (quelques fichiers à copier) : sauvegarde – retour en arrière possible.
- Installation, déploiement et migration facile des machines virtuelles d'une machine physique à une autre (copier – coller)
- Une machine virtuelle par version de LabVIEW : plus de conflit de version
- Facilité pour test de driver et logiciel

### ❑ Les Inconvénients

- Le matériel est virtualisé donc lenteur réseau, port série, USB,...
- Le matériel est virtualisé donc difficulté à installer un véritable matériel par exemple matériel National Instruments
- Conseillé d'utiliser une « Grosse » machine physique (8 Go RAM, processeurs multi-core,
- Gestion des licences : une Machine Virtuelle = une licence
- Taille sur le disque de la machine virtuelle et de ses snapshots

## ☐ Logiciel libre par Oracle



Nom	Modifié le
CET2	17/05/2013 21:0
Win7 LV2010 T-DX V3	26/06/2013 19:0
Win7 NI2012-2	03/04/2013 22:2
Win7 SP1 + LV2011	11/03/2013 22:2
Win7 SP1 Test	10/05/2013 22:2
WinXP LV711 T-DX	09/03/2013 11:2
WinXP LV861 T-D5 V2	22/05/2012 16:3
WinXP LV2009 SP1 T-DX	20/03/2013 21:3
WinXP SP3 Test	09/05/2012 10:0

- ❑ <http://fr.wikipedia.org/wiki/TortoiseSVN>
- ❑ [http://fr.wikipedia.org/wiki/Version\\_d%27un\\_logiciel](http://fr.wikipedia.org/wiki/Version_d%27un_logiciel)
- ❑ [https://tuleap.net/?utm\\_source=forge&utm\\_medium=forge&utm\\_campaign=forge](https://tuleap.net/?utm_source=forge&utm_medium=forge&utm_campaign=forge)
- ❑ <http://www.enalean.com/produits/tuleap/>
- ❑ [http://fr.wikipedia.org/wiki/Machine\\_virtuelle](http://fr.wikipedia.org/wiki/Machine_virtuelle)
- ❑ [https://fr.wikipedia.org/wiki/Jenkins\\_\(informatique\)](https://fr.wikipedia.org/wiki/Jenkins_(informatique))
- ❑ [ni.com/largeapps](http://ni.com/largeapps)

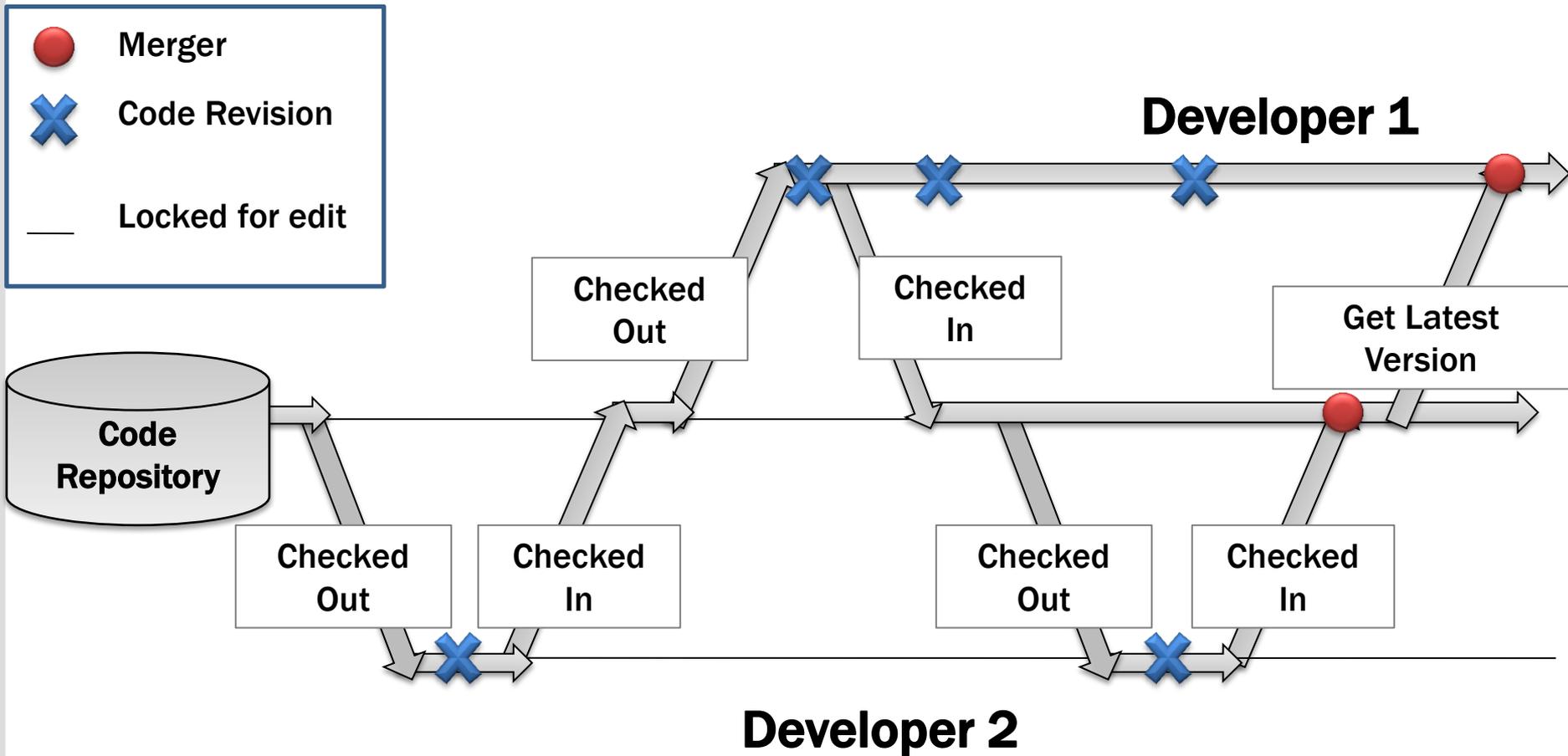


TortoiseSVN





- ❑ Sujets de réflexion :
- ❑ SCC : Base Centralisée VS décentralisée
- ❑ SCC : Principe de l'arborescence du dépôt : trunk, branches, tags
- ❑ SCC : LVDiff
- ❑ IC : Intégration Continue : Jenkins

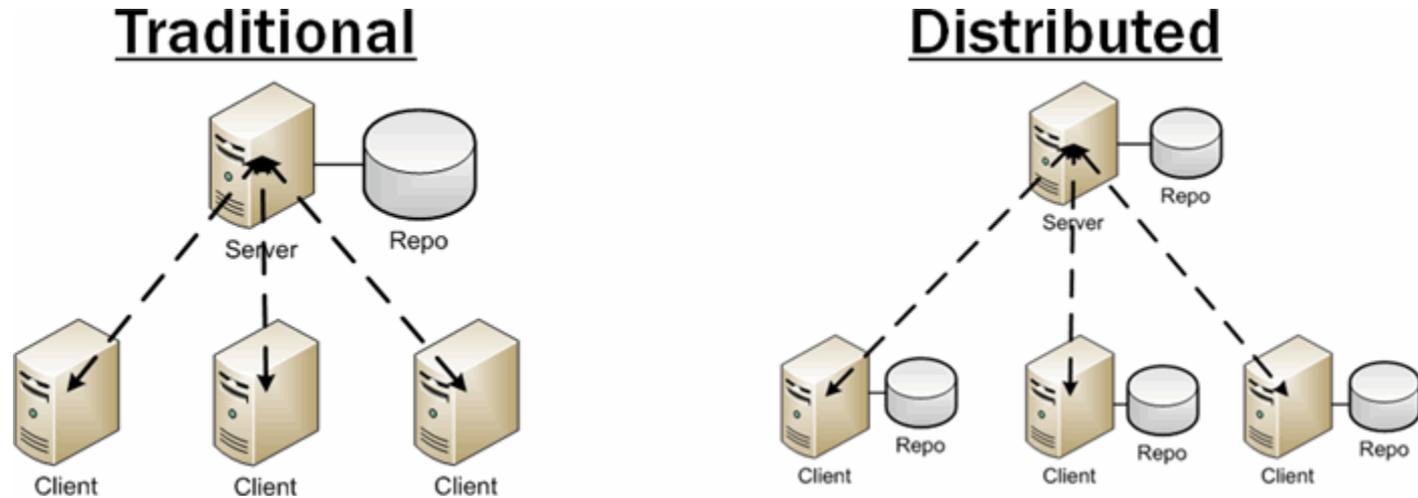


## Graphical Differencing

The screenshot displays the LabVIEW DIFF graphical differencing tool. It compares two versions of a LabVIEW block diagram, 'Main.vi'. The differences are highlighted with red circles. A 'Differences' panel at the bottom lists 31 differences, with 'Block Diagram objects' checked. A callout box contains a checklist of features:

- Provides a checklist of changes
- Useful for peer reviews
- Available via command-line

## ❑ Base Centralisée VS décentralisée



- ❑ **Gestion de versions centralisée** : un seul dépôt des versions qui fait référence.
  - Simplifie la gestion des versions
  - mais est contraignant pour certains usages comme le travail sans connexion au réseau.
  
- ❑ **Gestion de versions décentralisée** plusieurs dépôts.
  - Permet à chacun de travailler à son rythme, de façon désynchronisée des autres, puis d'offrir un moyen à ces développeurs de s'échanger leur travaux respectifs.
  - Ce système est très utilisé par les logiciels libres.

- ❑ Avantages de la gestion décentralisée :
  - Contributeurs peuvent travailler sans être connecté au gestionnaire de version ;
  - Participation à un projet sans nécessiter les permissions par un responsable du projet (les droits de commit/soumission peuvent donc être donnés après avoir démontré son travail et non pas avant) ;
  - la plupart des opérations sont plus rapides car réalisées en local (sans accès réseau) ;
  - Permet le travail privé pour réaliser des brouillons sans devoir publier ses modifications et gêner les autres contributeurs ;
  - Permet toutefois de garder un dépôt de référence contenant les versions livrées d'un projet.

- ❑ Désavantage de la gestion décentralisée :
  - cloner un dépôt est plus long que récupérer une version pour une gestion de version décentralisée car tout l'historique est copié ;
  - il n'y a pas de système de lock (ce qui peut poser des problèmes pour des données binaires qui ne se fusionnent pas).

- ❑ L'intégration continue vérifie à chaque modification de code source que le résultat des modifications ne produit pas de régression dans l'application développée.
- ❑ Modifications de code -> Compiler et tester votre application.
- ❑ Si erreur avertir les développeurs afin qu'ils puissent tout de suite corriger le problème.
- ❑ Suivre la santé de votre projet

