



NIDays

GRAPHICAL SYSTEM DESIGN
CONFERENCE

11 Février 2014 | Paris

nidays.fr

Les outils qui vous veulent du bien

Plus de temps pour développer en LabVIEW



Luc DESRUELLE



Certifié LabVIEW Architect (CLA), MESULOG

La société MESULOG en bref

- Activité : Développement logiciel test et mesure
- Compétences : **LabVIEW** (Windows, RT, DSC, PDA, FPGA),
TestStand
VeriStand
- Localisation : Grenoble (Moirans, 38)
- Partenaire National Instruments (2001)
- Développeurs certifiés LabVIEW et TestStand



La société MESULOG en bref

www.mesulog.fr

- Réalisations

- Présentations techniques
 - LabVIEW
 - TestStand



Réalizations



Société -> Présentations techniques
www.mesulog.fr/presentations_techniques


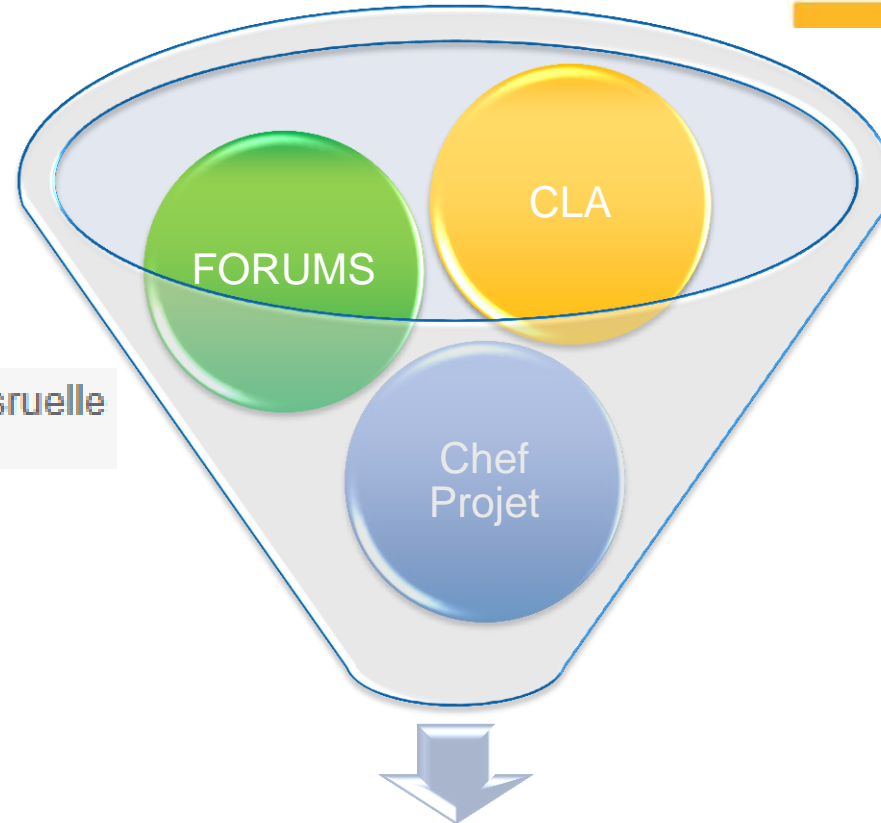
La société MESULOG en bref

- Ils nous ont fait confiance :

- ALSTOM Grid
- AREVA NP
- 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)
- ONERA
- PECHINEY
- RADIALL
- RENAULT
- STMicroelectronics
- THALES Alenia Space
- THALES LCD
- ...



Le présentateur



Luc_Desruelle
Active Participant

Luc DESRUELLE

Le constat

Le constat « amiable » de la gestion de projet



Exemples d'erreurs : bombe à retardement

Grosses erreurs autour du développement logiciel (?)

- ✓ Ne pas savoir où trouver de l'aide
- ✓ Pas de logiciel de gestion de version
- ✓ Pas de centralisation de l'information
- ✓ Pas d'**analyse** ni de **test** du code source
- ✓ Attendre la fin du projet pour générer l'exécutable
- ✓ Pas de gestion de la machine de développement
- ✓ *Autres ?*



Le constat

Gestion projet :
Juste avec

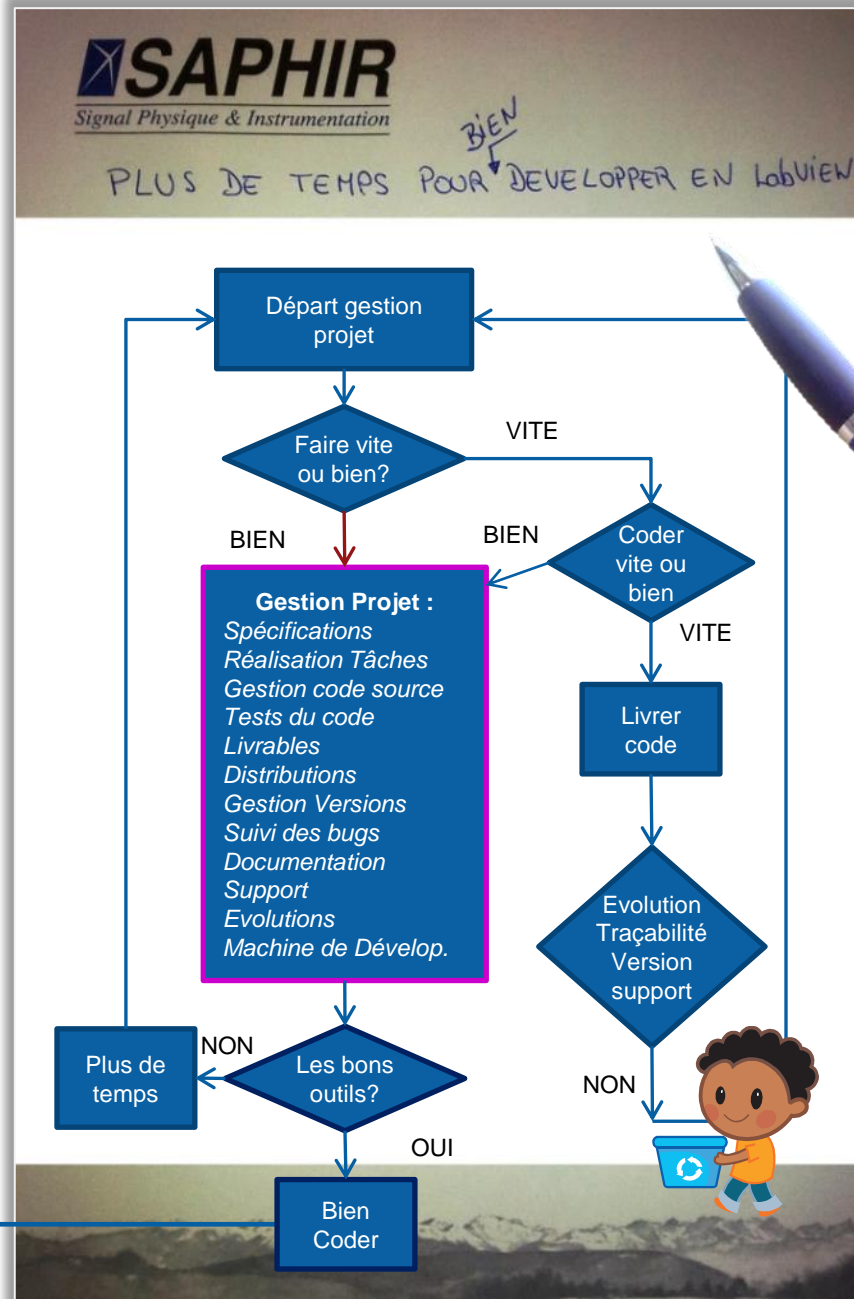
➤ un bloc notes

et

➤ un stylo MESULOG ?

Autres présentations

www.mesulog.fr/presentations_techniques
Programmation avancée sous LabVIEW



Du temps pour coder

Notre métier : chef de projet – développeur - passionné...

Mais... Combien de **temps** passons-nous à **coder** ?

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



Le but : trouver du temps pour coder

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



Sommaire

- **Où trouver de l'aide ?**
- **Gestion de projet :**
 - L'indispensable : logiciel de gestion de code source
 - La caisse à outils de gestion de projet
 - Forge (?)
- **Outils de développement**
 - Logiciels d'analyse et de test de National Instruments
- **Outils complémentaires**
 - Autres outils *de la caisse à outils*
 - Zoom sur : Machine Virtuelle

Où trouver de l'aide ?



Où trouver de l'aide ?

- 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 dans région : Cf LUGE

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

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

- Des articles sur le site de NI + LAVA

- « Le logiciel ne se limite pas au code » *de l'équipe JKI*



LUGE – Rencontre développeurs Rhône-Alpes

❑ Via la Communauté Francophone

- <https://decibel.ni.com/content/community/regional/france/labview?view=all#/?tagSet=1850>

❑ En région : communauté LUGE (Rhône-Alpes)

- <https://decibel.ni.com/content/community/regional/france/labview?view=all#/?tagSet=1780>
- Rencontre - Echange entre développeurs



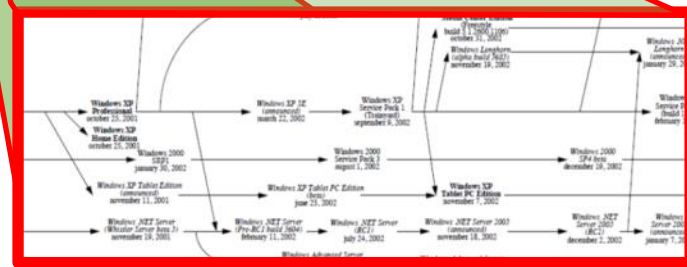
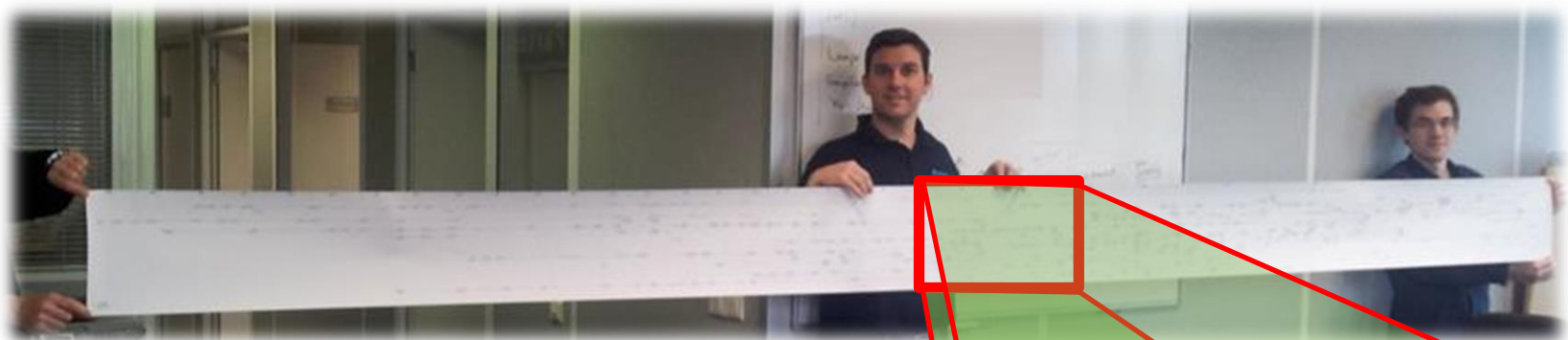
Gestion de projet



L'indispensable : Logiciel de gestion de code source

Qui se cache derrière ce « *Graphique de révision* »?

(Lien entre l'ensemble des versions d'un logiciel)



Logiciel de gestion de code source

- Le contrôle du code source est l'un des aspects les plus importants du développement logiciel
- Simple à mettre en œuvre
- Mais aussi trop négligé

Qui utilise un logiciel de gestion code source ?

Lequel ?

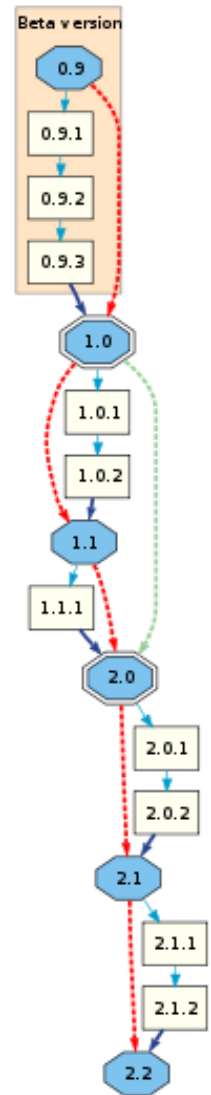
- Mais quelle utilité ??

Définition : Version VS Révision

http://fr.wikipedia.org/wiki/Version_d%27un_logiciel

- Les logiciels évoluent = **versions** du logiciel
- Chaque étape d'avancement = **modifications de fichiers (code, vi,...) = révision** de fichiers
- « **révision** » d'un fichier VS la « **version** » d'un logiciel.

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



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 travailler sur plusieurs sites ?
- Comment faire des tests et les conserver ?
- Comment revenir sur une ancienne version ?
-

Logiciel Gestion de code source

- 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 le développement multi-développeurs (gestion de conflits)

... beaucoup d'autres fonctionnalités

SCC : histoire

- 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) |
| Client-serveur | Logiciels libres | CVS (1990) CVSNT (1992) SVN (2000) |
| | Logiciels propriétaires | Rational ClearCase (1992) CCC/Harvest (Années 70) CMVC (1994) Visual SourceSafe (1994) Perforce (1995) 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) Mercurial (2005) Bazaar (2005) Fossil (2007) Veracity (2011) |
| | Logiciels propriétaires | BitKeeper (1998) Plastic SCM (2007) |

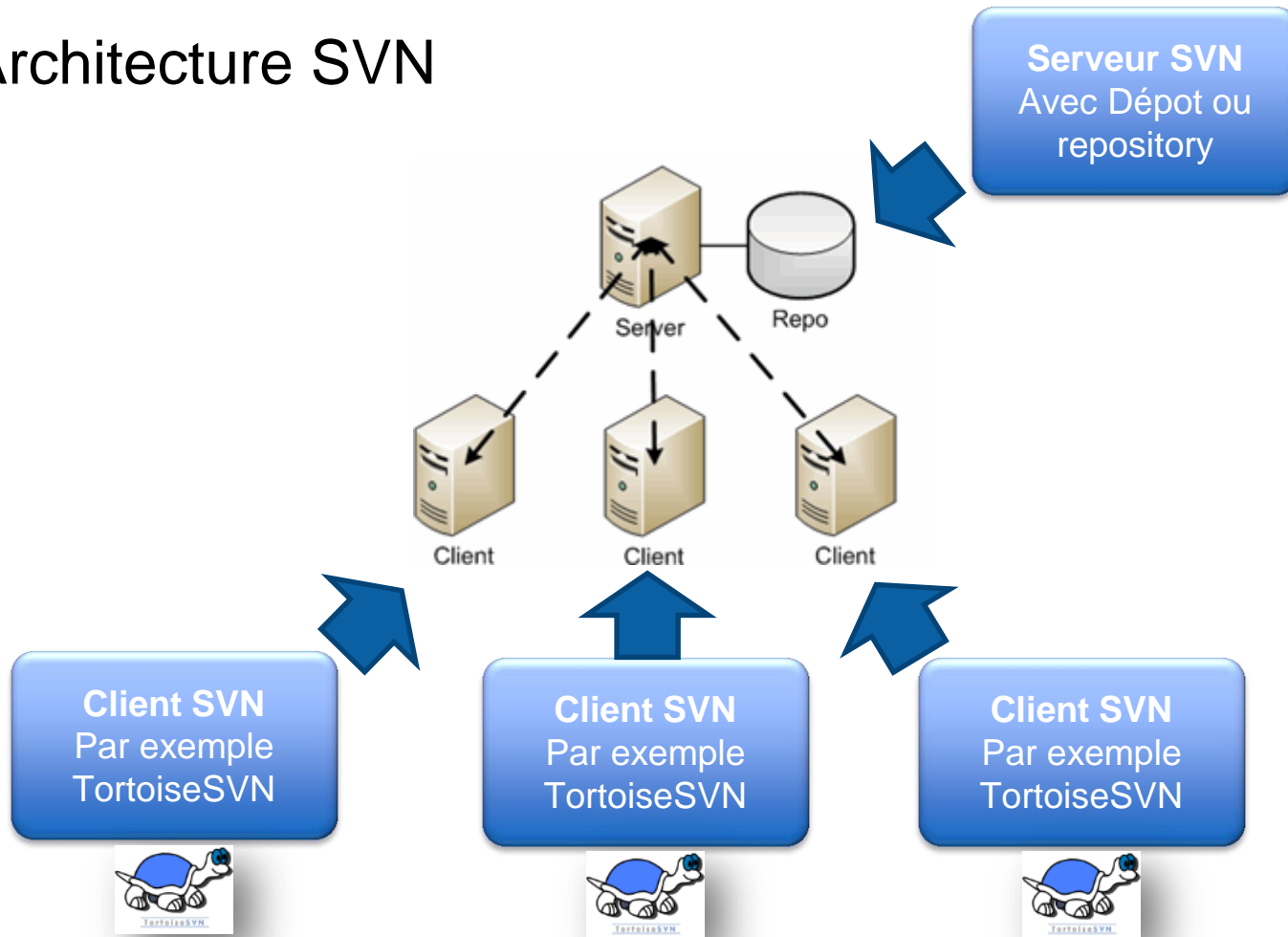
Zoom sur Subversion



- **Subversion** (en abrégé **SVN**)
- Logiciel Libre
- Mode client-serveur, avec :
 - Un Serveur unique :
 - les fichiers constituant la référence (le « dépôt » ou « référentiel »)
 - **un logiciel serveur** Subversion tournant en 'tâche de fond'
 - Des postes clients :
 - les fichiers copiés depuis le serveur, éventuellement modifiés localement depuis,
 - **un logiciel client** permettant la synchronisation entre chaque client et le serveur de référence

Architecture SCC Client - Serveur

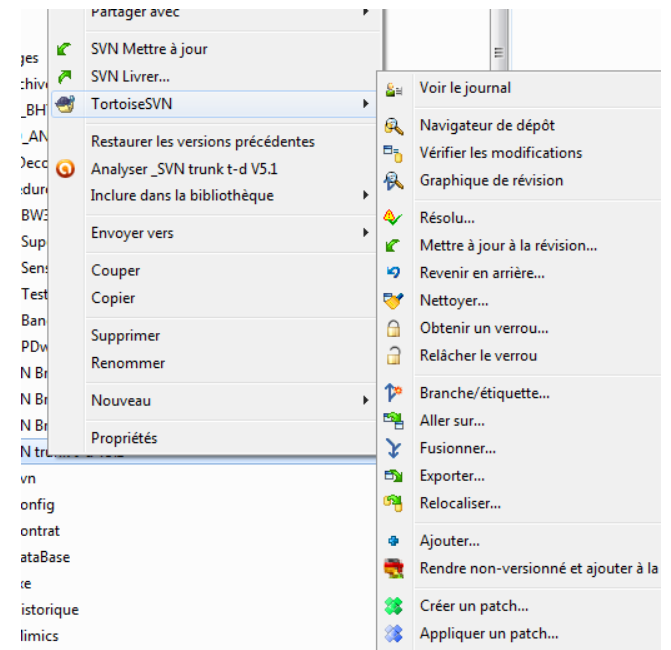
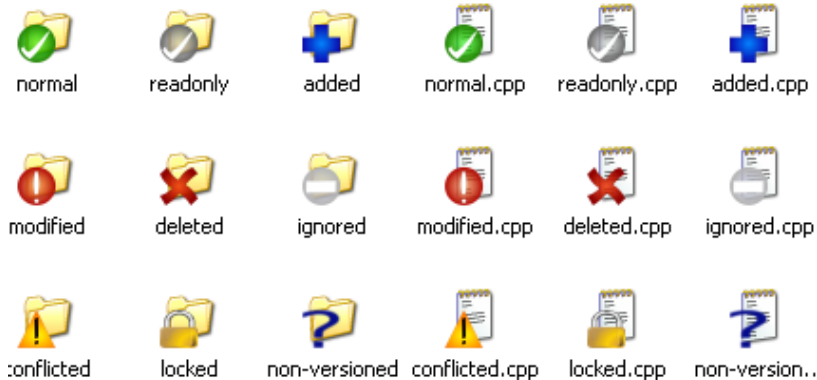
- Architecture SVN



Un logiciel client SVN : TortoiseSVN

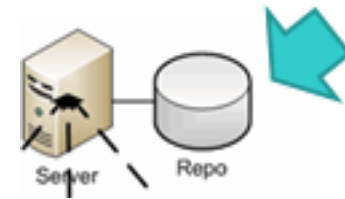


- Son nom vient de l'anglais *Tortoise*, « Tortue »
- Logiciel Libre
- S'intègre à l'explorateur de Windows :
 - Superposition d'icône aux répertoires et fichiers permettant de visualiser instantanément l'état
 - « Menu contextuel » des actions



SVN : créer le dépôt

- Création du dépôt sur le serveur



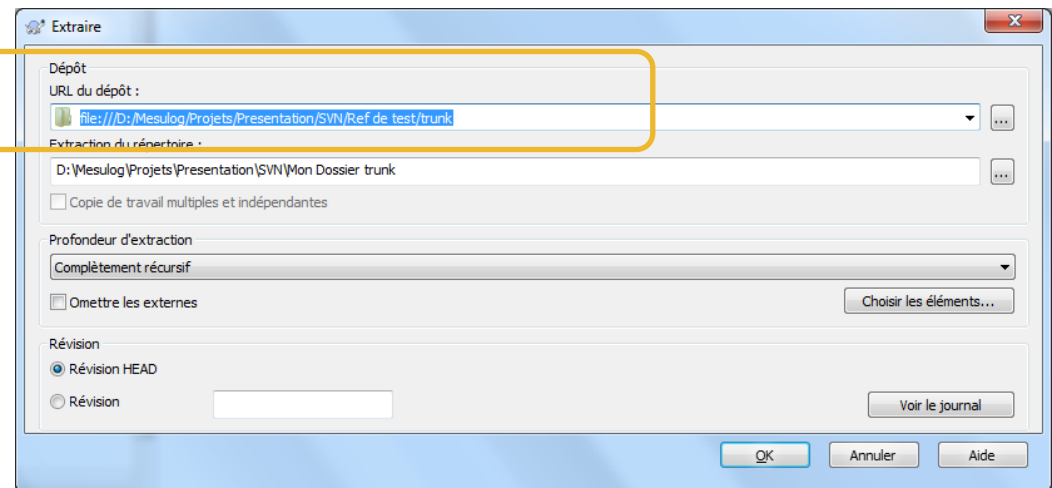
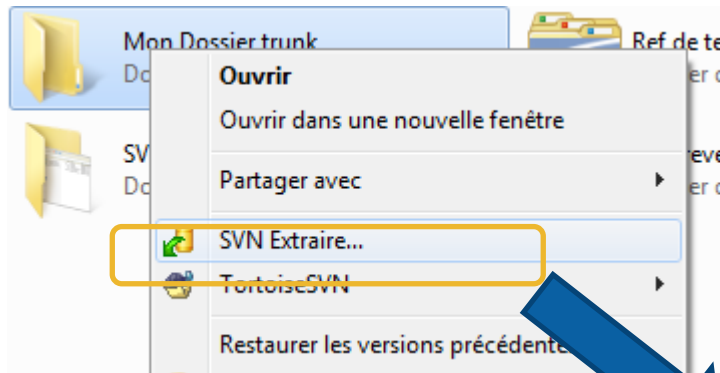
Chemin du dépôt :

- trunk
- tags
- branches

| Fichier | Extension | Révision | Auteur | Taille | Date | Verrou |
|----------|-----------|----------|--------|--------|---------------------|--------|
| branches | | 1 | ld | | 28/06/2013 17:34:03 | |
| tags | | 1 | ld | | 28/06/2013 17:34:03 | |
| trunk | | 1 | ld | | 28/06/2013 17:34:03 | |

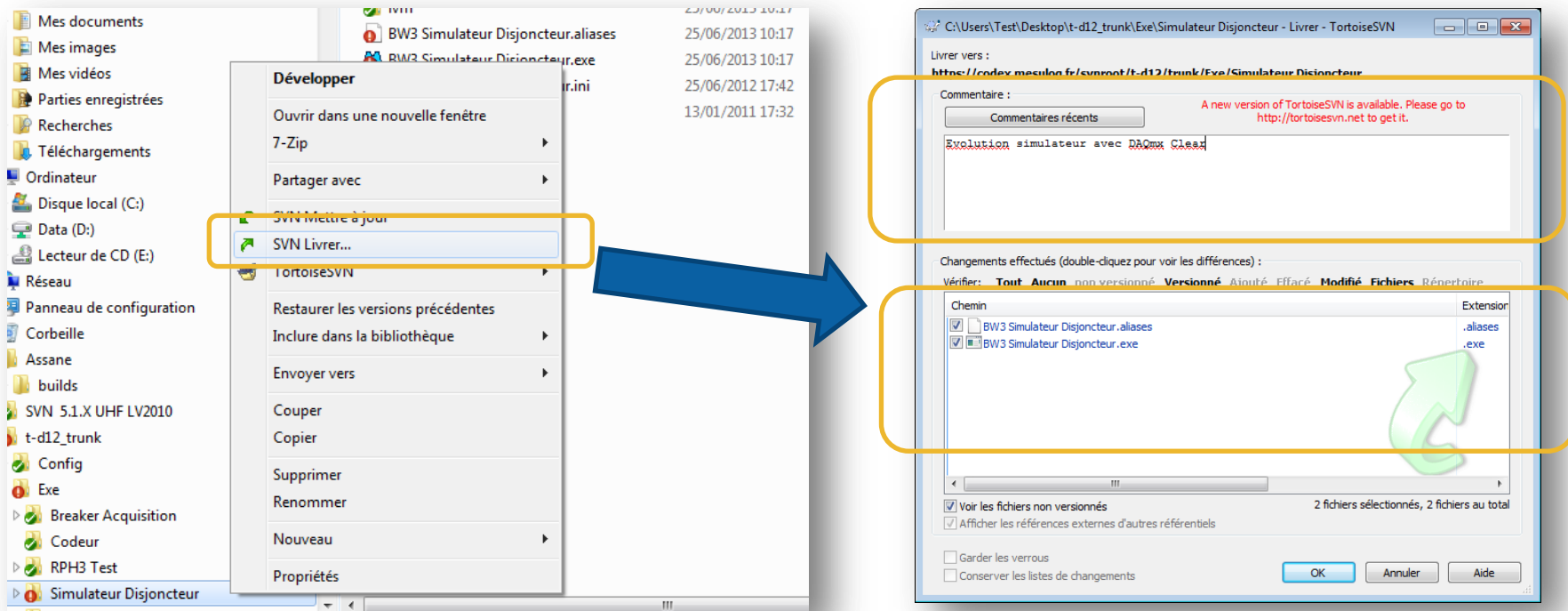
SVN : extraire le dépôt

- Exemple : Extraire (*extract*) sur un client



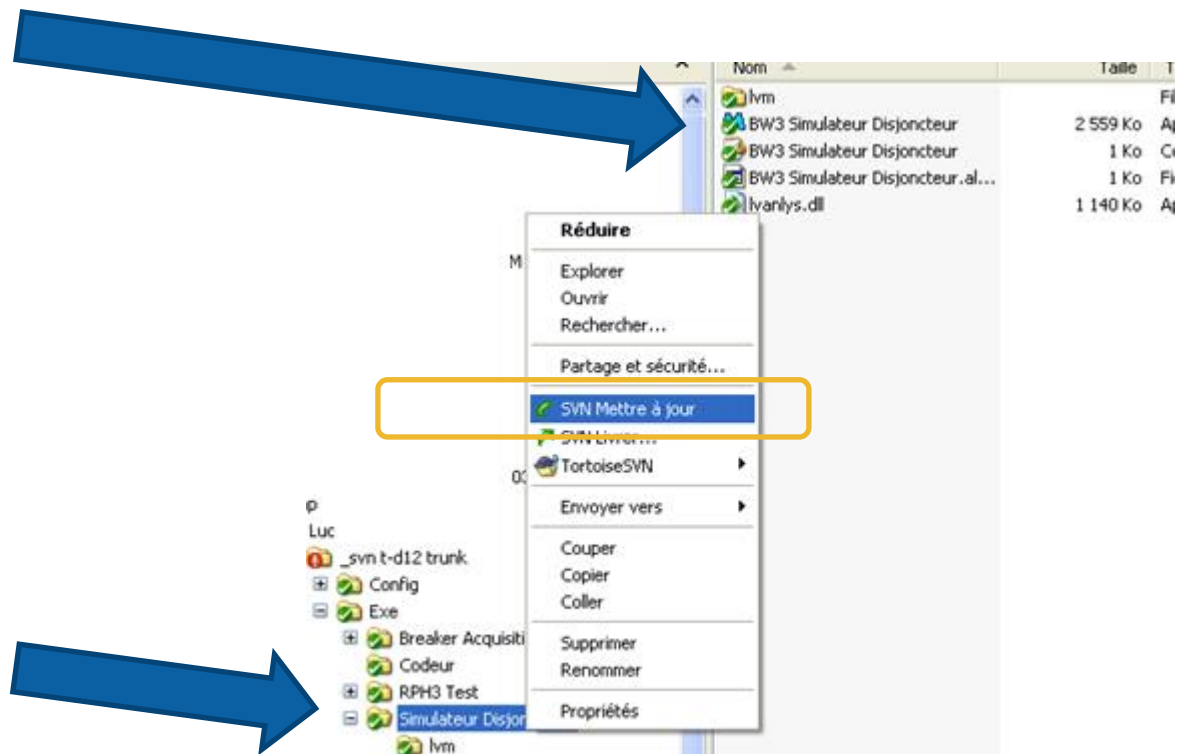
SVN : livrer

- Exemple : Livraison (*commit*) par un des clients



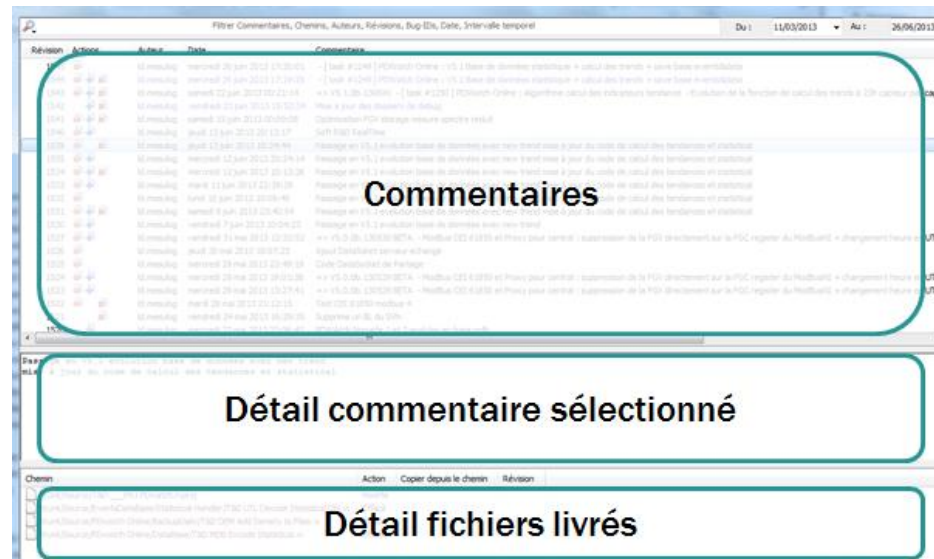
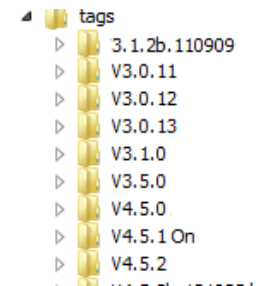
SVN : mettre à jour

- Exemple : Mise à jour (*update*) par un autre client



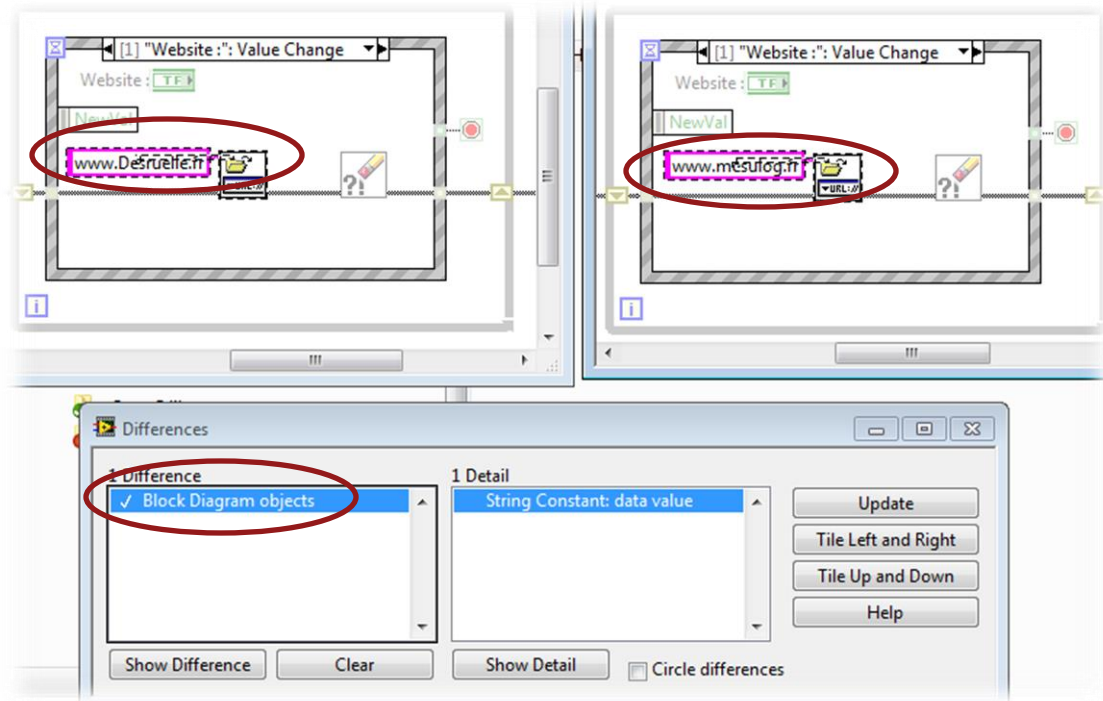
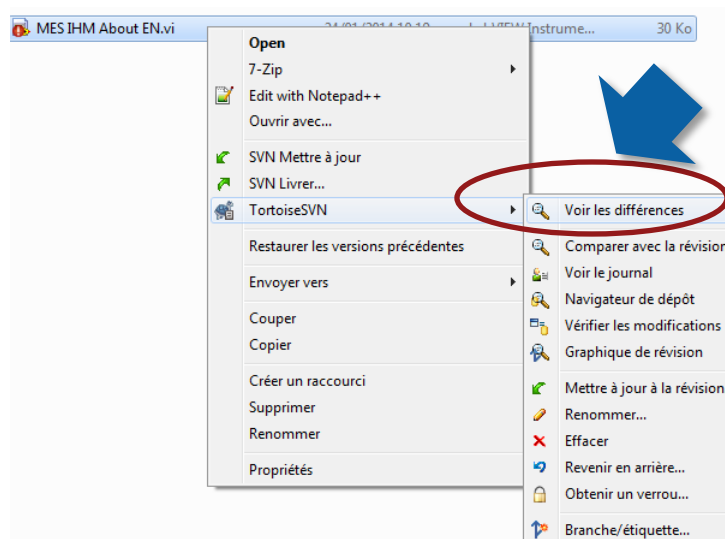
SVN : étiquette & journal

- Etiquette (*tag*) : retrouver une version
- Journal :



SVN : voir les différences

- Permet de comparer les différences entre 2 révisions d'un vi's



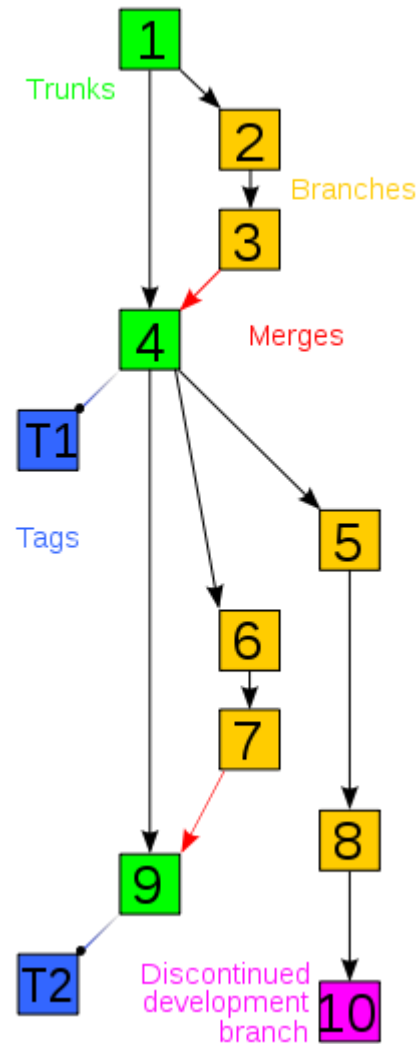
Pour utiliser LabVIEW compare via TortoiseSVN :

Ajouter : TortoiseSVN -> Configuration -> Programmes externes -> programme pour comparer les différentes révisions de fichiers

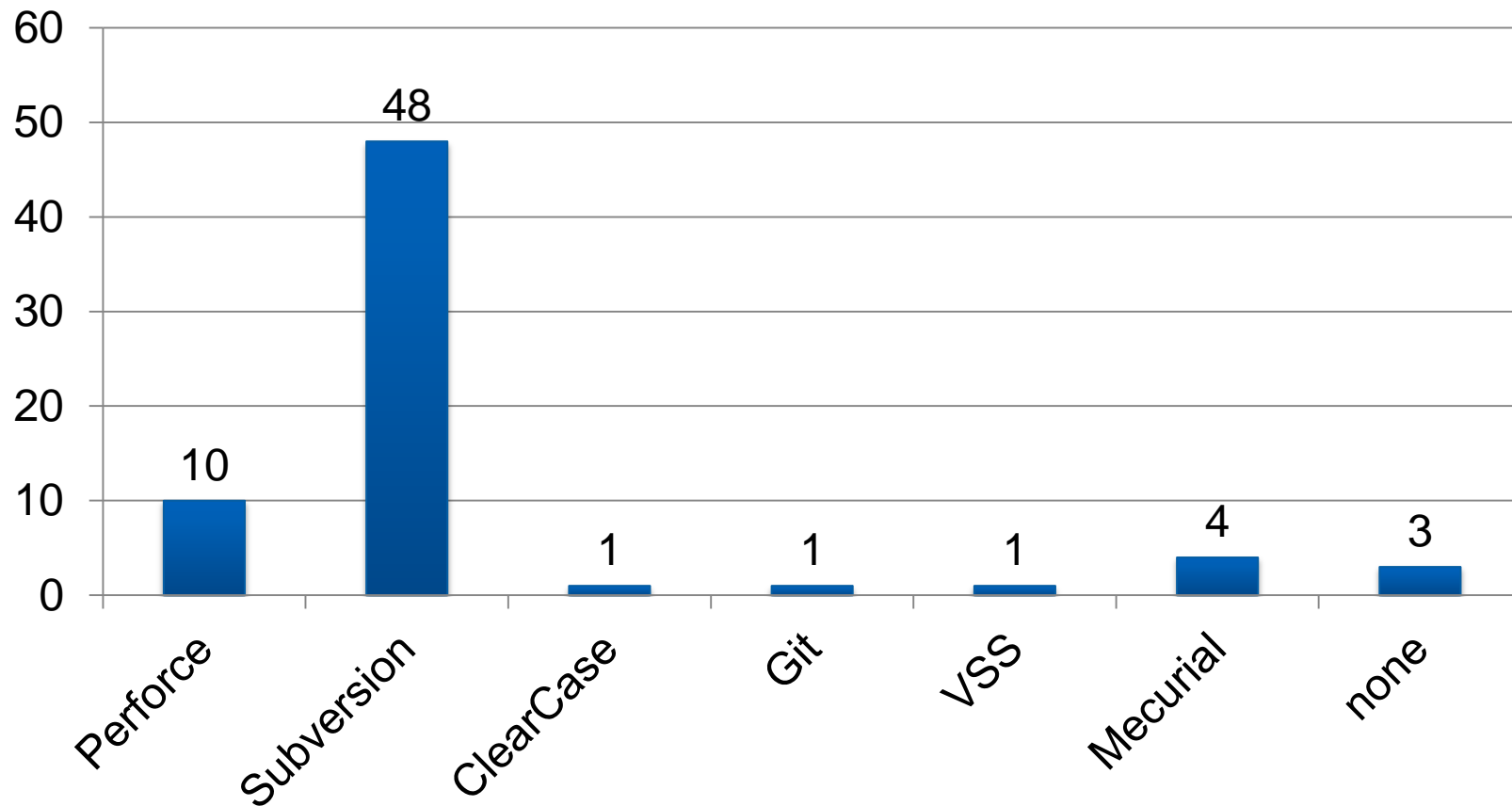
"C:\Program Files\National Instruments\Shared\LabVIEW Compare\LVCcompare.exe" %mine %base -nobdcosm -nobdpos

SVN : arborescence

- Trunk : tronc
- branches
- tags : étiquettes



Popularités des logiciels SCC



Source: 2010 ni.com/largeapps survey

Inconvénient de SVN : pas intégré par NI LabVIEW.... mais

Native LabVIEW Integration

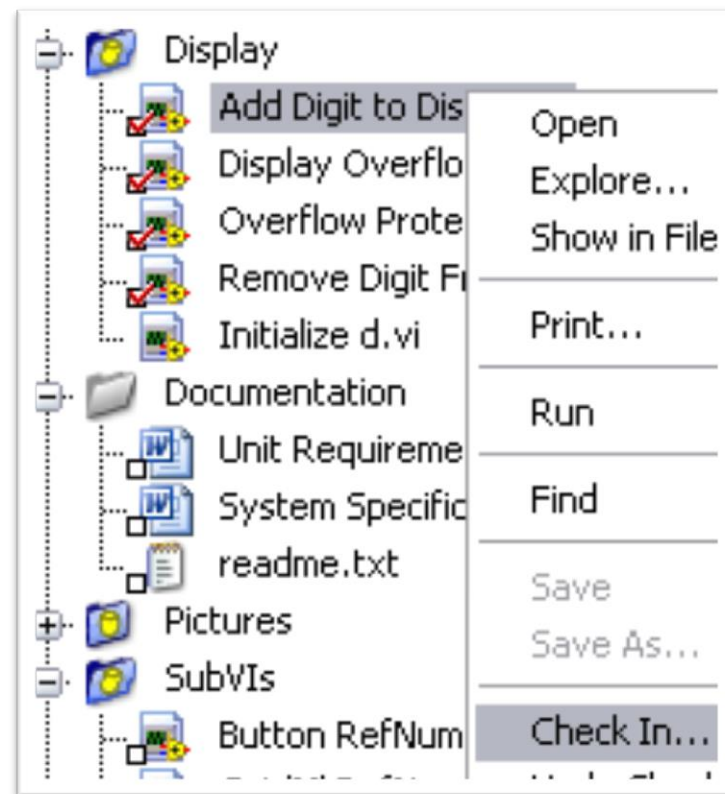
- Perforce

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 : TSVN Toolkit
- Mercurial



Pour aller plus loin

- Gestion utilisateurs – mot de passe
- Accès distant svn, http, https
- Branches, tags
- Merge
- Lock
-

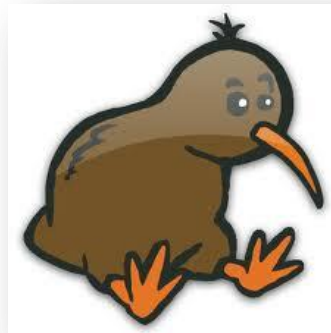
Caisse à outil de gestion de projets

- Bug tracker : outil de gestion de bug ou anomalie
 - Identifier, diffuser, sauvegarder
 - Associer à une version de code
 - Correctif code et distribution



- Quelques exemples

- FogBugz
- Mantis
- Bugzilla
- Tuleap



Caisse à outil de gestion de projets

• Diffusion de la documentation

- Box (box.com)
- Google Drive
- Wiki
- Tuleap



• Diffusion des « livrables »

- Serveur FTP
- Tuleap

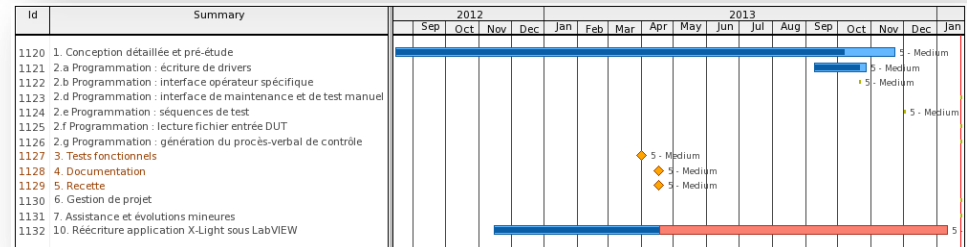
RAMSES TestBench [Créer une version]

V1.7 2013-10-01

| Nom de fichier | Taille (Ko) | D/L | Arch. | Type | Date | Md5sum du fichier | Propriétaire |
|-----------------------------|-------------|-----|-----------------|-------------|------------|----------------------------------|---------------------|
| RAMSES_TestBench_V1.7.0.zip | 17 805 | 3 | i386 Windows | Binary .zip | 2013-10-01 | 40c761a6cce560dca0c18a5481ac2bac | Jean-Louis Schricke |

• Gestion des tâches

- MS Project
- Trello
- Tuleap



• Intégration Continue

- Jenkins

Derniers Builds ANP2 Build

- Dernier Build : # 58
- Dernier succès : # 58
- Dernier échec : #

Tendance :

Forge (?)

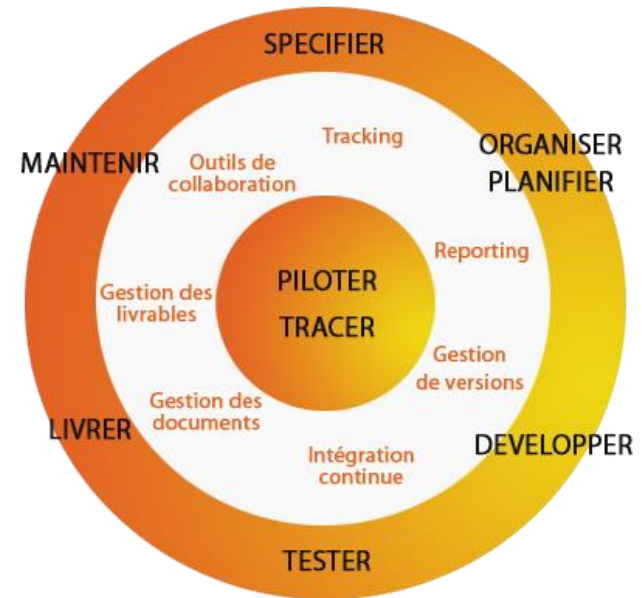
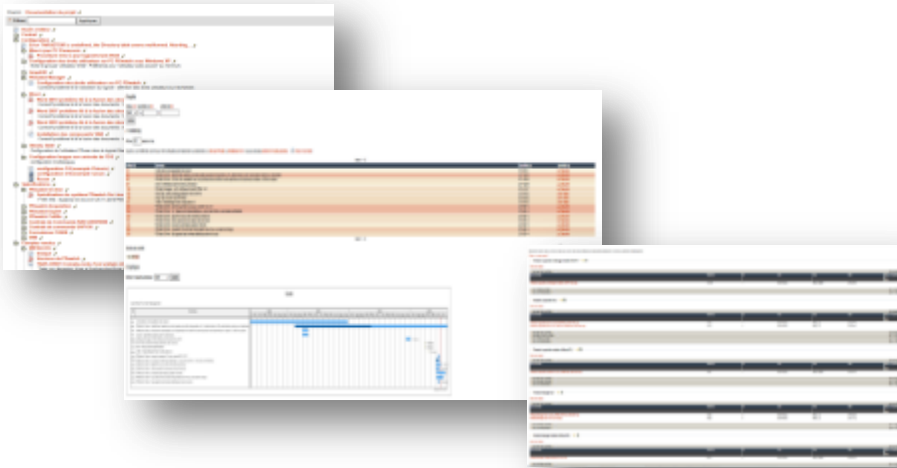
Une forge !



Késako ?

Forge

- Beaucoup de logiciels MAIS logiciels séparés....
- Forge ?
 - Système qui gère « les développements collaboratif de logiciel »
 - Système simple et efficace
 - Regroupe de nombreux outils
- Chez MESULOG : Tuleap



- **Mettre à disposition de tous les acteurs autorisés, l'ensemble des données d'un projet :**
 - Documentation (doc #ID)
 - Code source (gestion sous Subversion, https, svn #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.

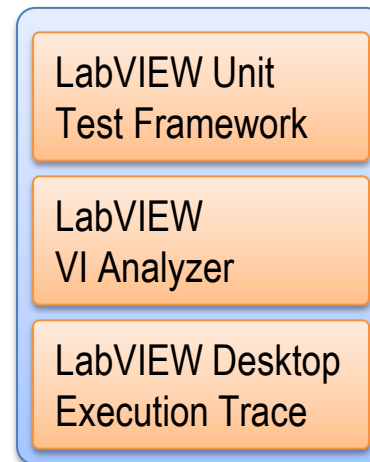
« *La Suite ALM 100% Open-Source* »

Outils de développement

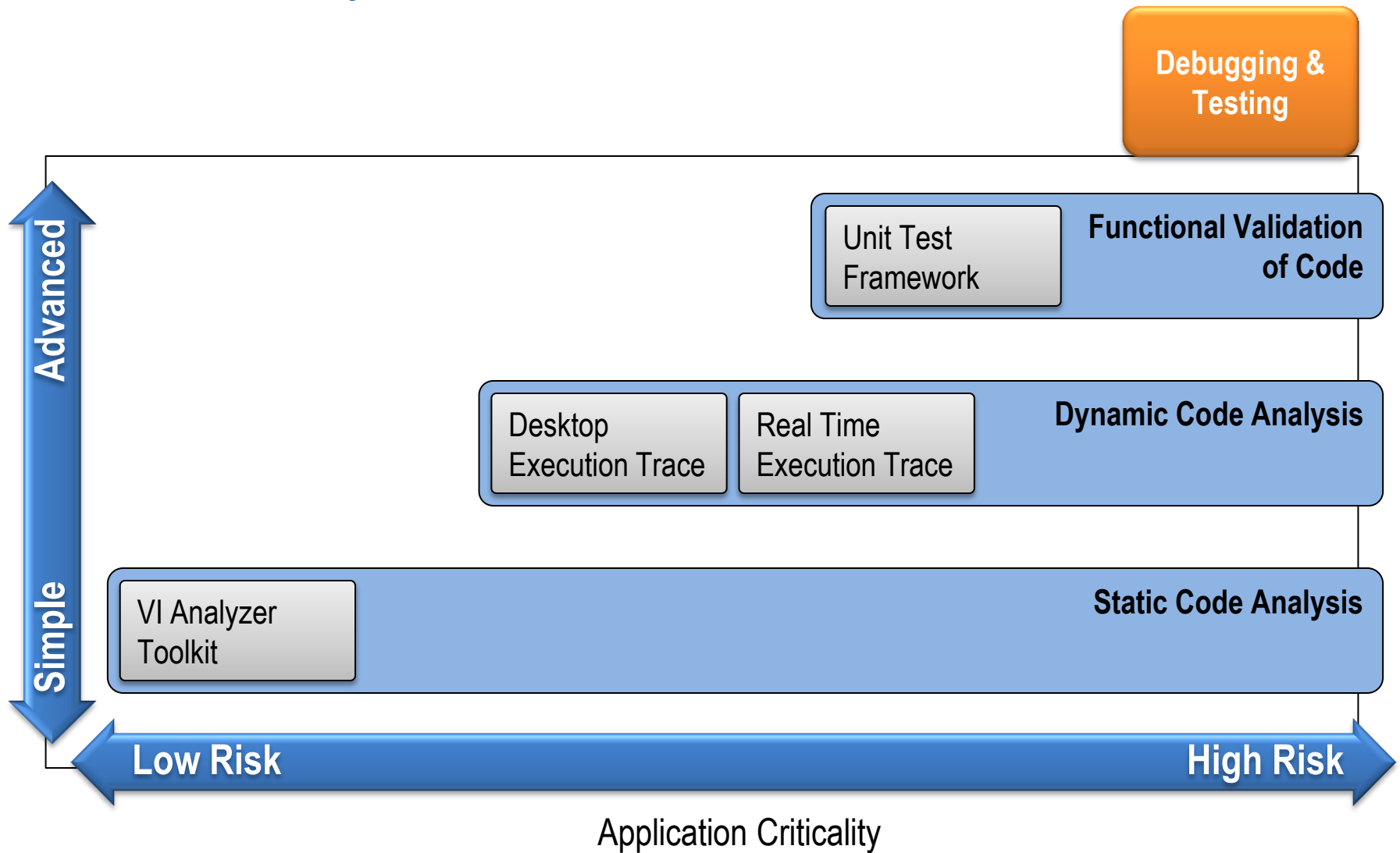
Logiciels de Test et d'Analyse du code



By National Instruments

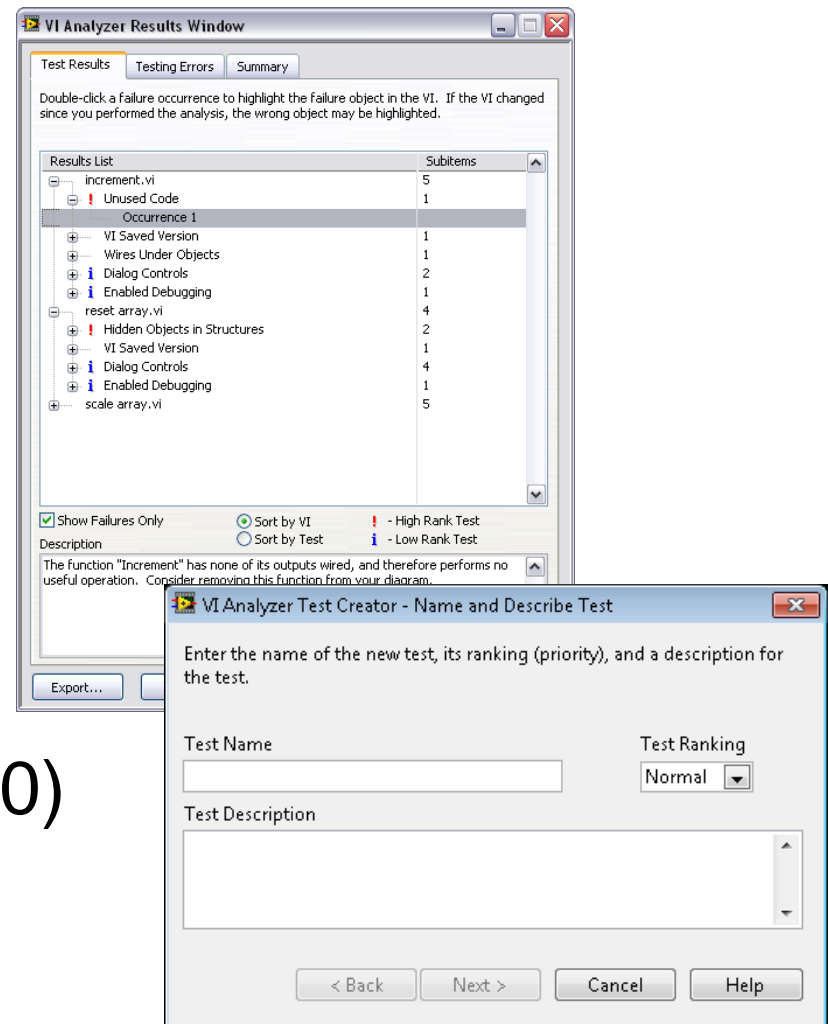


Test et Analyse du code



VI Analyser

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



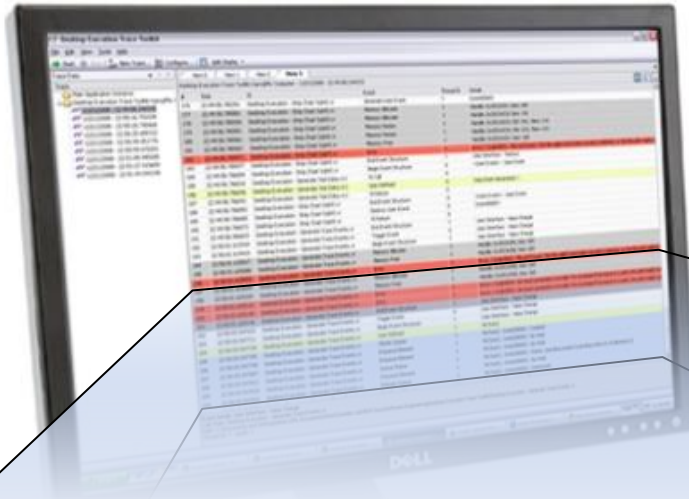
VI Analyser : sur la communauté

- Fonctions de test

- 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

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

Desktop Execution Trace Toolkit

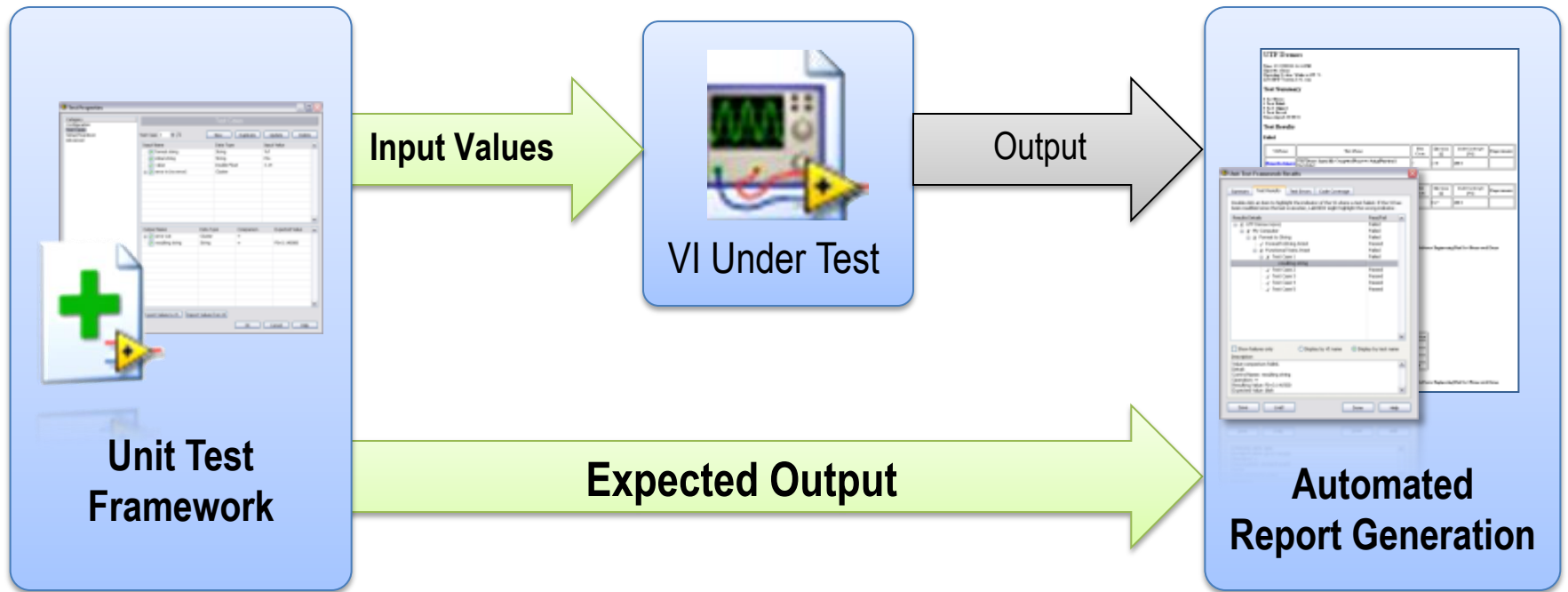


Trace During Run-Time:

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

| | | | |
|--------------------------|-----------------|---|--|
| 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 |

LabVIEW Unit Test Framework



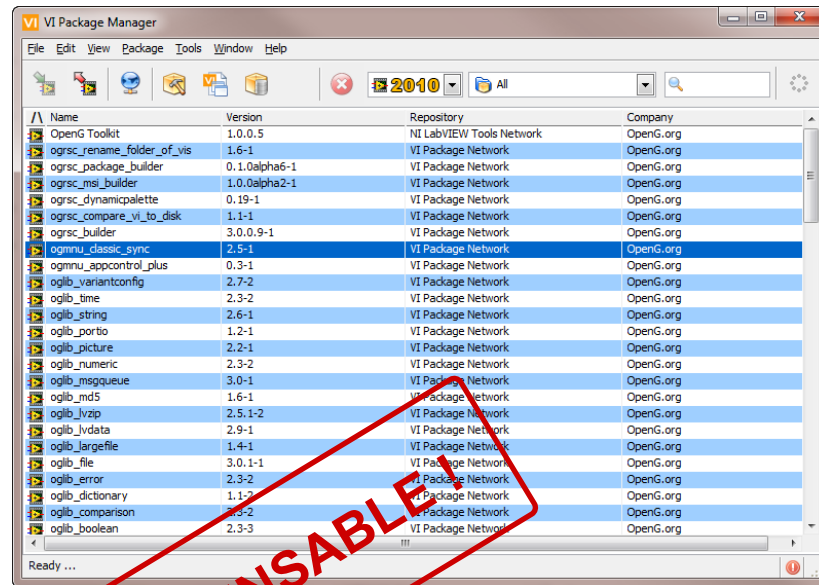
Test vector = Input value(s) + Expected output(s)

Outils complémentaires

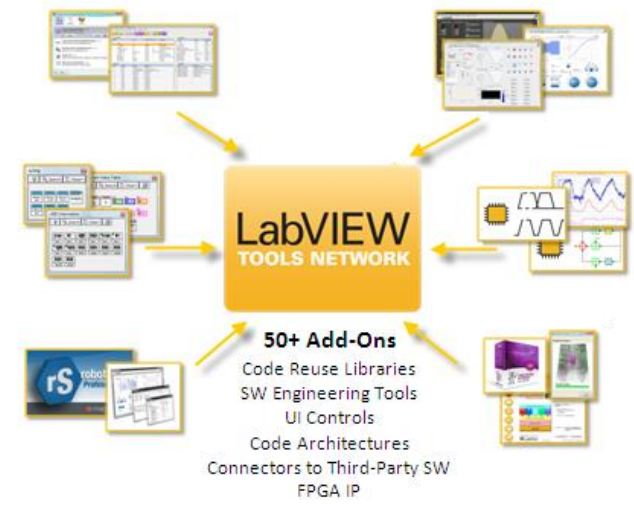
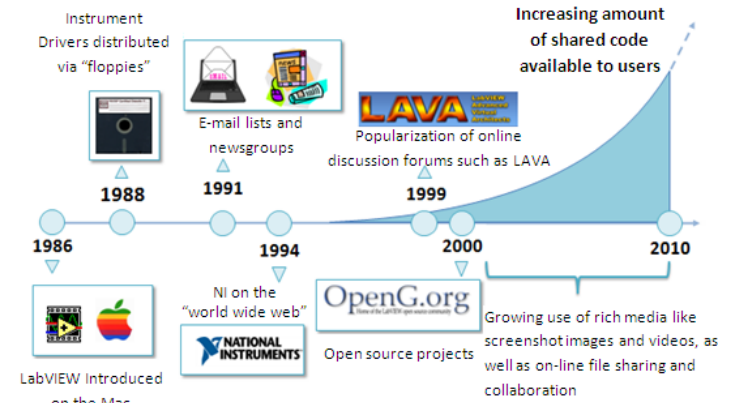


VI Package Manager & LabVIEW Tools Network

- Intégré dans LabVIEW depuis LV2013
- <http://www.ni.com/labview-tools-network/>



INDISPENSABLE



Bibliothèques OpenG

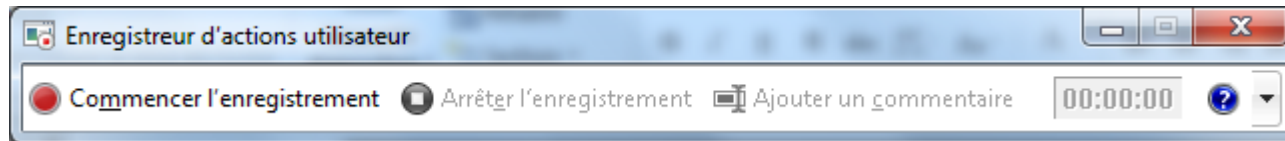
- Palette de VI's pour LabVIEW
- Des centaines de VIs en open source
 - la manipulation de tableaux
 - la manipulation de chaînes de caractères
 - la gestion de fichiers et Zip
 - l'implémentation MD5 digest
 - la gestion des erreurs
 - la manipulation de données variant
 - ...



PSR de Windows ???

- **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 » :



Autolt

- Automatiser des tâches système
- Simuler des actions utilisateurs : un script *pilote* une application en lieu et place d'un contrôle manuel
- Automatiser une application LabVIEW via ActiveX
- Réaliser une démonstration



Intégration continue : Jenkins

- Suivre la santé de votre projet
- **L'intégration continue** vérifier 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.



Autres

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

Les VOTRES ?

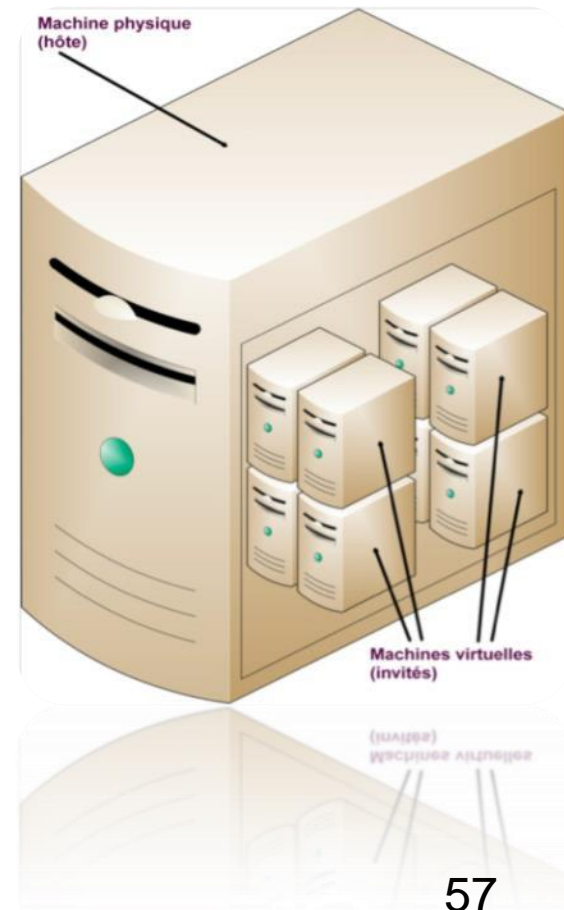
Zoom sur : Machine Virtuelle



Machine Virtuelle : définition

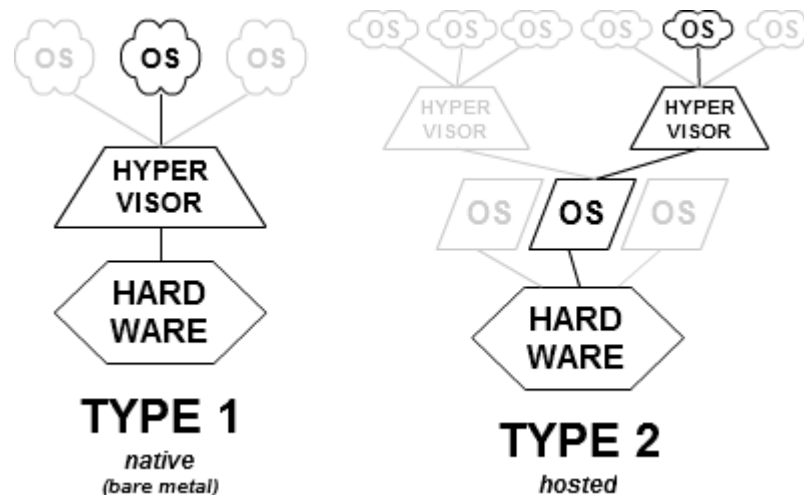
- *But : Virtualiser la plate-forme de développement*
- **Virtualisation** : simuler l'existence de plusieurs machines informatiques sur une seule.

- ❑ **Machine virtuelle** illusion d'un appareil informatique créée par un logiciel d'émulation.
- ❑ Simule la présence des ressources matérielles et logicielles (mémoire, processeur, disque dur, système d'exploitation, pilotes)



Machine Virtuelle : définition

- **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

Machine Virtuelle : Exemples

- **Virtualisation des serveurs** est le modèle le plus courant. plusieurs serveurs virtuels sur un serveur physique
- OS moderne simule une machine qui exécute MS-DOS;
- **Machine de développement** : une machine virtuelle par plate-forme de développement (PDA, Windows, linux,..)



Machine Virtuelle pour le développement

• Les avantages

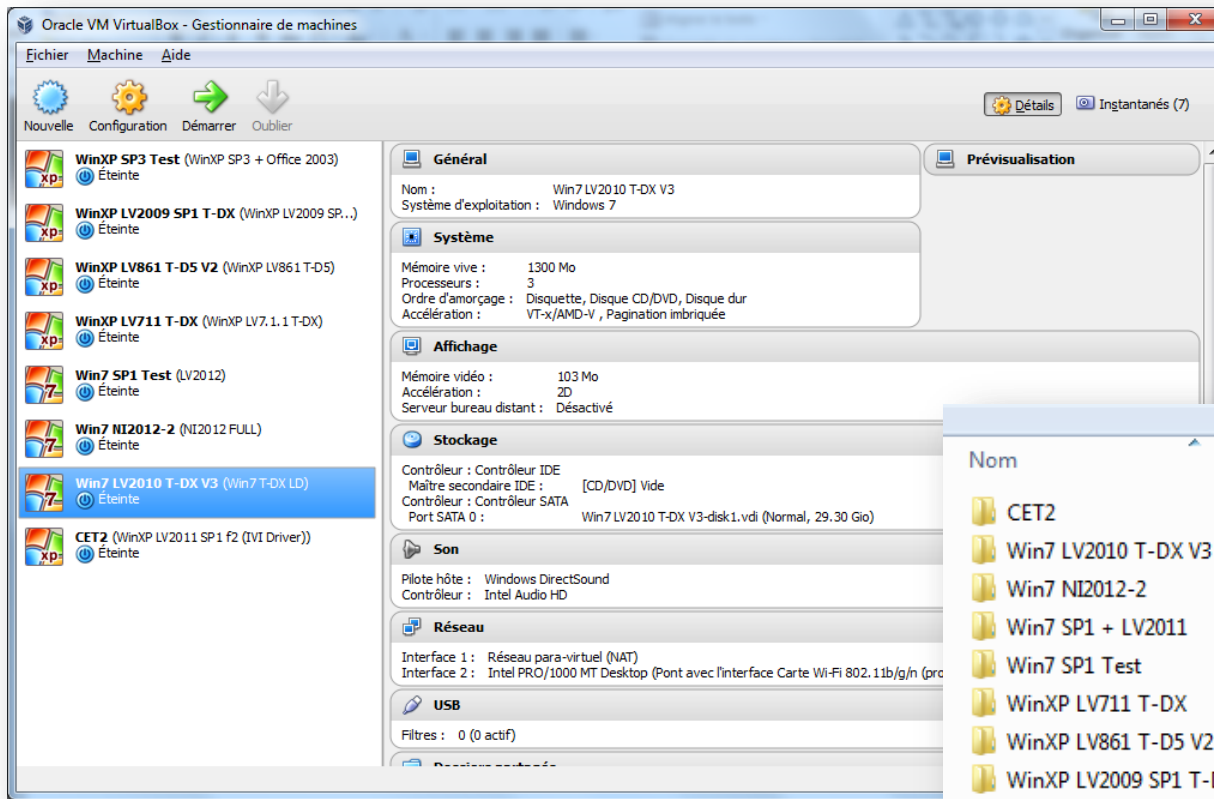
- Sécurisation des plates-formes de développement (quelques fichiers à copier) : sauvegarde – retour en arrière possible.
- Installation, déploiement et migration facile des machines virtuelles d'une machine physique à une autre (copier – coller)
- Une machine virtuelle par version de LabVIEW : plus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

• Les inconvénients

- Lenteur réseau, port série, USB,...
- 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

Oracle VM VirtualBox

- Logiciel libre par Oracle (rachat de Sun)



| Nom | Modifié le |
|-----------------------|------------------|
| CET2 | 17/05/2013 21:00 |
| Win7 LV2010 T-DX V3 | 26/06/2013 19:00 |
| Win7 NI2012-2 | 03/04/2013 22:22 |
| Win7 SP1 + LV2011 | 11/03/2013 22:22 |
| Win7 SP1 Test | 10/05/2013 22:22 |
| WinXP LV711 T-DX | 09/03/2013 11:22 |
| WinXP LV861 T-D5 V2 | 22/05/2012 16:33 |
| WinXP LV2009 SP1 T-DX | 20/03/2013 21:33 |
| WinXP SP3 Test | 09/05/2012 10:00 |

Podium



Liens

- fr.wikipedia.org/wiki/TortoiseSVN
- fr.wikipedia.org/wiki/Version_d%27un_logiciel
- tuleap.net/?utm_source=forge&utm_medium=forge&utm_campaign=forge
- www.enalean.com/produits/tuleap/
- fr.wikipedia.org/wiki/Machine_virtuelle
- [fr.wikipedia.org/wiki/Jenkins_\(informatique\)](http://fr.wikipedia.org/wiki/Jenkins_(informatique))
- ni.com/largeapps

Questions



Restez **connecté** avant et après NIDays



ni.com/**communaute-francophone**



facebook.com/**nifrance**



twitter.com/**nifrance**



youtube.com/**nifrance**