

How can a team of several people:

- work on the same software,
- possibly on different versions,
- possibly on the same files,
- at the same time,
- from possibly different locations?

SCM & Revision Control

SCM (1/4)

- SCM stands for *Software Configuration Management*
- SCM is aimed at **establishing and maintaining the integrity and control of software packages** (that is, source codes, specifications, documentation, etc.) throughout projects life cycle

SCM (2/4)

- The best solution to **track and control the changes** in software projects to maintain software integrity and traceability
- SCM involves two practices:
 - Establishment of **baselines**
 - **Revision control**

SCM (3/4)

- SCM also:
 - Facilitates team interactions
 - Enforces defect tracking
 - Enforces quality

SCM (4/4)

- What is tracked with SCM?
 - All kind of documents
 - Specifications
 - Analysis and design documents
 - Manuals (administration guide, user guide, etc.)
 - Etc.
 - Source code
 - Potentially, software packages
 - Etc.

SCM Process (1/6)

- Four activities:
 - Configuration identification
 - Configuration control/management
 - Configuration status accounting
 - Configuration audit

SCM Process (2/6)

- Configuration identification:
 - Aimed at understanding the current state of the software
 - Aimed at identifying **configuration items** and **baselines**
 - Can be done in various ways including:
 - Use of formal configuration documentation (checklist, etc.)
 - Use of already-in-place revision control systems
 - Etc.

SCM Process (3/6)

- Configuration control/management:
 - Aimed at implementing a **controlled change process**
 - Defines procedures to support the software life cycle
 - Usually based on workflows
 - Also usually based on revision control systems
 - May be achieved by defining a **control board** which is aimed at approving/rejecting **change requests**

SCM Process (4/6)

- Configuration status accounting:
 - Records and reports all the necessary information on the **status of the development of the software**





















SCM Process (5/6)

- Configuration audit:
 - Aimed at ensuring that the **SCM process is adhered to**
 - Ensures that the configurations contain everything they do have to contain
 - Ensures that the configurations comply with requirements/specs/etc.
 - Generally occurs at **delivery** or **when effecting the change**

SCM Process (6/6)

- Configuration audit:
 - Three types of audits:
 - **Physical** audit: Ensures that identified configuration is the same as the actual configuration
 - **Functional** audit: Ensures that functional description maps to the actual functions (primarily done via testing)
 - **Process** audit: Ensures that the defined process is consistently followed

Workflow – Example (1/3)

Key ↕	T	Summary	Original Estimate	Reporter	Impacted Release/s
XNET-2284		Intégration 2.5.1-SAC			
XNET-2283		If a BLOB/CLOB column name is "value", then the generated setValue() method of the corresponding managed bean is wrong			2.5.0
XNET-2281		generated application crashes when trying to display a float value			2.5.1
XNET-2280		When generating an application from a XAM file, CLOB types are generated as "blob" instead of "clob"			2.5.1
XNET-2273		"title" attribute on a <h:graphicImage> tag is not correctly generated	1 hour		2.4.3 2.5.0
XNET-2243		Move MyFaces from 1.1.6 to 1.1.7 and Tomahawk from 1.1.7 to 1.1.8	2 hours		2.5.0
XNET-2239		If the menu bar of an Xnet Web application is large, menu items may become unclickable	2 hours		2.4.3 2.5.0
XNET-2218		Project Management			
XNET-2203		Using the toString() representation of a bean to track the instance in XnetListDataModel is unsafe and error prone	2 days		2.5.0
XNET-2201		mail-1.4.jar is corrupted			2.5.0
XNET-2199		XnetLoadBundle does not work with rich:tab component	2 days		2.5.0
XNET-2185		Missing 'f' in facesFragmentListEditableColumn.vm			2.5.0
XNET-2184		Jenia compatibility fix for Tomcat	3 hours		2.5.0
XNET-2183		ResourceLocator Tomcat compatibility			2.5.0
XNET-2181		XPSHome : better error message for missing mapping			2.5.0
XNET-2177		UserManagementProcess can't be overridden through the META-INF/services mechanism			2.4.3 2.5.0
XNET-2175		Model.xml should allow to specify a default sort order	1 day		
XNET-2174		SimpleBatchManager directly stops after it starts			2.5.0
XNET-2148		The images of the XnetPager component (prev, next, etc.) are not displayed when visualizing a portlet using Internet Explorer, it works with Firefox	1 day, 4 hours		2.5.0
XNET-2145		Incorporate Datatable Faces component into Xnet	4 days		2.4.3

Workflow – Example (2/3)

[Xnet](#)

→ XPSHome : better error message for missing mapping

Created: 13/Mar/09 04:21 PM Updated: 27/Jul/09 05:07 PM

Component/s:	Foundation
Impacted Release/s:	2.5.0
Implementation Release/s:	2.5.1
Security Level:	Public Issue (Public Issues are viewable by Managers, Members, Submitters and Viewers.)


Time Tracking:	Unknown
----------------	---------

Total estimate:	Total Original Estimate : Unknown	Total Estimate to Complete (ETC) : Unknown
-----------------	--------------------------------------	---

Project Context:	Xnet
------------------	------

ER Type:	Technical Evolution
----------	---------------------

ER Description:	<p>If XnetClassMapping can't find the mapping description for a particular field : throw new RuntimeException("No field named " + name + " for " + getBeanClass()); I think</p> <p>throw new RuntimeException("No mapping for field named " + name + " for " + getBeanClass());</p> <p>would give more precise information.</p>
-----------------	---

Requestor:	
------------	---

Requestor Priority:	Medium
---------------------	--------

Priority:	Medium
-----------	--------

User Field 3:	IVDCN
---------------	-------

Workflow – Example (3/3)

All Comments Work Log Change History Transitions ClearCase Change Set	
Change by [redacted] - 13/Mar/09 04:24 PM	
Field	Original Value
Assignee	
Change by [redacted] - 13/Mar/09 05:35 PM	
eESM ticket #	{html} <p>Ticket Number : Ticket number 1</p> <p>Ticket Number : Ticket number 2</p> {html}
Change by [redacted] - 13/Mar/09 05:38 PM	
Assignee	[redacted]
Change by [redacted] - 13/Mar/09 05:44 PM	
Status	Submitted [1]
Implementation Release/s	
Change by [redacted] - 13/Mar/09 05:44 PM	
Status	Evaluated [5]
Change by [redacted] - 13/Mar/09 05:44 PM	
Status	Evaluation Approved [10001]
Change by [redacted] - 13/Mar/09 05:45 PM	
Status	Implemented [10017]
Change by [redacted] - 17/Mar/09 09:34 AM	
Assignee	[redacted]

Configuration Item (1/2)

- Sometime referred as a *work product*
- Base unit of a SCM system
- Corresponds to **what is actually changed**
- Changing a configuration item creates a new **revision** of this configuration item

Configuration Item (2/2)

- Examples:
 - Specifications
 - Analysis and design documents
 - Manuals
 - Source code

Baseline (1/2)

- Allows the marking of significant states within a series of parallel changes on configuration item
- Generally marks **approval status** and, thus, defines characteristics for the baselined configuration items
- May also simply mark **noticeable progression** (alpha, milestone, etc.)

Baseline (2/2)

- Marks a set of approved configuration items
- Allows retrieving **old sets of consistent configuration items** (that is, to potentially return to an earlier state of the configuration items)

Revision Control (1/2)

- Also known as:
 - *Version control*
 - *Source control*
- Used to track and control changes on configuration items:
 - **Who** did the change
 - **When** it was done
 - **What** has been changed

Revision Control (2/2)

- Used to **baseline** configuration items
 - Who?
 - When?
- Changes are often referred to as **revisions**
- Revisions can be:
 - **Compared**
 - **Restored**
 - **Merged**
- Implemented into **Revision Control Systems**

RCS

- Two types of revision control systems:
 - Centralized
 - Distributed

Centralized RCS (1/3)

- Revision control functions are located in one place
- Has to manage concurrent access to the same configuration items through:
 - File locking
 - File merging

Centralized RCS (2/3)

- File locking:
 - User gets the locks
 - User modifies the configuration item(s)
 - User commits his changes and unlocks the configuration item(s)
 - Efficient in case merging would be too difficult (models, etc.)

Centralized RCS (3/3)

- Version merging:
 - The first user committing his changes has nothing to do
 - Other users committing their changes have to merge their changes with the previously committed ones

Distributed RCS

- No central repository (not so sure...)
- P2P approach
- Each user has a *working copy* of the repository

Centralized RCS Terminology (1/7)

- Baseline
 - Sometimes referred as *label* or *tag*
 - **Approved revision** (based on a significant status) of a set of configuration items
- Branch
 - Branches may be created from time to time to allow working on **several copies** of the same set of configuration items, **independently from each other**

Centralized RCS Terminology (2/7)

- Change/diff
 - Sometimes referred to as a *commit*
 - Represents a change brought to a configuration item (or to a set of configuration items)
- Checkout
 - Action of **creating a working copy** from the repository
- Checkout (IBM Rational ClearCase)
 - Action of **reserving** (locking) **a configuration item** in order to modify it

Centralized RCS Terminology (3/7)

- Commit/checkin
 - Action of **applying changes** from the working copy to the repository
- Conflict
 - Occurs when a configuration item has been changed by several users at the same time
 - Has to be resolved through a *merge*

Centralized RCS Terminology (4/7)

- Head
 - The **latest revision** of the repository
- Import
 - Action of **adding a new set of configuration items** from the working copy to the repository
- Lock
 - Action of **locking a file to prevent concurrent modifications**

Centralized RCS Terminology (5/7)

- Merge
 - Consists in **applying two changes** brought on the same configuration item
 - May happen:
 - When updating the working copy from the repository
 - When applying a change from one branch to the trunk (or to another branch)
- Repository
 - Central store of configuration items, changes, baselines, etc.

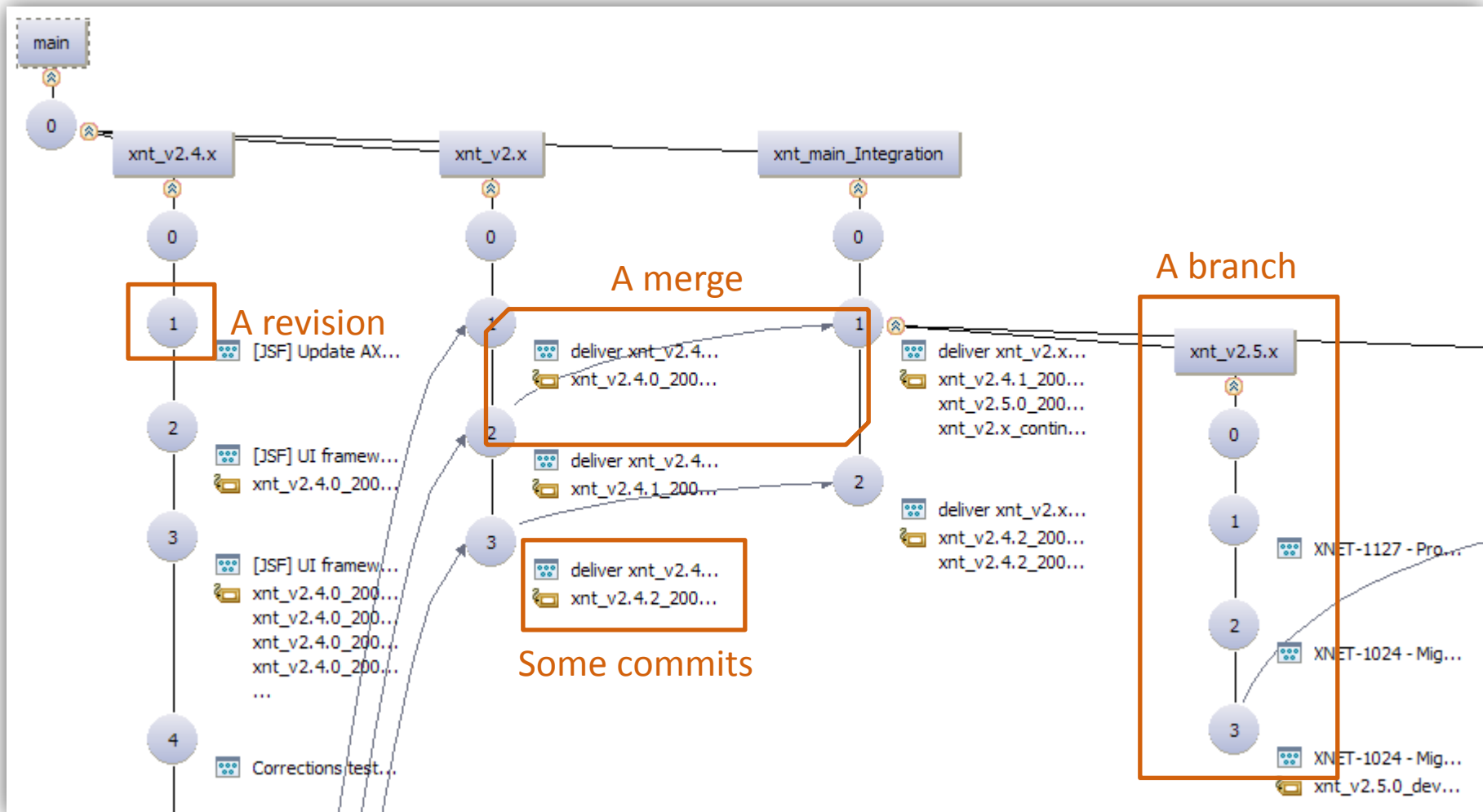
Centralized RCS Terminology (6/7)

- Trunk
 - Can be seen as the main branch (although it is not a branch)
- Update
 - Action of applying changes (not necessarily the latest ones) from the repository to the working copy

Centralized RCS Terminology (7/7)

- Working copy
 - Local copy of a set of configuration items from the repository
 - Place where configuration items are actually changed before being committed to the repository

Centralized RCS – Example (1/3)



Centralized RCS – Example (2/3)

Problems	Javadoc	Declaration	Search	Console	Properties	Styles	Call Hierarchy	Servers	ClearCase Version Tree	ClearCase History	X
History: XnetManagedBean.java											
Date	User	Name	Version	Event Kind	Comn						
Oct 23, 2008 5:42:19 PM		/xnt/xnt/xnt_core/xnetFacesUI/src/com...	/main/xnt_main_Integra...	create							
Sep 12, 2008 9:51:03 AM		/xnt/xnt/xnt_core/xnetFacesUI/src/com...	/main/xnt_main_Integra...	create							
Jun 20, 2008 10:15:51 AM		/xnt/xnt/xnt_core/xnetFacesUI/src/com...	/main/xnt_main_Integra...	create							
May 27, 2008 3:37:24 PM		/xnt/xnt/xnt_core/xnetFacesUI/src/com...	/main/xnt_main_Integra...	create							
Nov 15, 2007 9:39:48 AM		/xnt/xnt/xnt_core/xnetFacesUI/src/com...	/main/xnt_main_Integra...	create							
Oct 30, 2007 3:08:05 PM		/xnt/xnt/xnt_core/xnetFacesUI/src/com...	/main/xnt_main_Integra...	create							
Sep 14, 2007 10:11:19 AM		/xnt/xnt/xnt_core/xnetFacesUI/src/com...	/main/xnt_main_Integra...	create							
Aug 28, 2007 9:33:33 AM		/xnt/xnt/xnt_core/xnetFacesUI/src/com...	/main/xnt_main_Integra...	create							
Aug 27, 2007 2:57:16 PM		/xnt/xnt/xnt_core/xnetFacesUI/src/com...	/main/xnt_main_Integra...	create							
Aug 27, 2007 2:57:16 PM		/xnt/xnt/xnt_core/xnetFacesUI/src/com...	/main/xnt_main_Integra...	create							
Jul 31, 2007 11:59:04 AM		/xnt/xnt/xnt_core/xnetFacesUI/src/com...	/main/xnt_main_Integra...	create							
Jul 18, 2007 6:22:41 PM		/xnt/xnt/xnt_core/xnetFacesUI/src/com...	/main/xnt_main_Integra...	create							
Jul 13, 2007 4:38:49 PM		/xnt/xnt/xnt_core/xnetFacesUI/src/com...	/main/xnt_main_Integra...	create							
Jun 27, 2007 6:39:57 PM		/xnt/xnt/xnt_core/xnetFacesUI/src/com...	/main/xnt_main_Integra...	create							
Jun 27, 2007 6:39:57 PM		/xnt/xnt/xnt_core/xnetFacesUI/src/com...	/main/xnt_main_Integra...	create							
Jan 24, 2007 2:16:05 PM		/xnt/xnt/xnt_core/xnetFacesUI/src/com...	/main/xnt_main_Integra...	create							
Jan 23, 2007 3:34:06 PM		/xnt/xnt/xnt_core/xnetFacesUI/src/com...	/main/xnt_v2.x/3	create							
Dec 14, 2006 11:53:11 AM		/xnt/xnt/xnt_core/xnetFacesUI/src/com...	/main/xnt_v2.4.x/22	create							
Aug 10, 2006 4:03:54 PM		/xnt/xnt/xnt_core/xnetFacesUI/src/com...	/main/xnt_main_Integra...	create							
Aug 9, 2006 9:32:21 PM		/xnt/xnt/xnt_core/xnetFacesUI/src/com...	/main/xnt_main_Integra...	create							
Aug 9, 2006 9:32:21 PM		/xnt/xnt/xnt_core/xnetFacesUI/src/com...	/main/xnt_main_Integra...	create							
Aug 9, 2006 9:35:17 AM		/xnt/xnt/xnt_core/xnetFacesUI/src/com...	/main/xnt_v2.x/2	create							
Jun 15, 2006 11:36:32 AM		/xnt/xnt/xnt_core/xnetFacesUI/src/com...	/main/xnt_v2.4.x/21	create							
May 12, 2006 9:45:28 AM		/xnt/xnt/xnt_core/xnetFacesUI/src/com...	/main/xnt_v2.4.x/20	create							

Some revisions

Centralized RCS – Example (3/3)

```
/**
 * Initializes the given managed bean for use in a "select one" configuration
 * @param listManagedBeanName
 */
public XnetListManagedBean initSelectOnePopup(String listManagedBeanName) {
    XnetListManagedBean listManagedBean = (XnetListManagedBean) FacesUtils.findManagedBeanByName(listManagedBeanName);
    listManagedBean.reset();
    listManagedBean.setParentManagedBean(this);
    return listManagedBean;
}
```

```
public List suggest(Class beanClass, String start) {
    if (start == null || start.equals("")) {
        return new ArrayList();
    }
    try {
        String componentId = (String)FacesContext.getCurrentInstance().getExternalContext().getRequestParameterMap().get("affected");
        UIComponent comp = FacesUtils.findComponentInRootByFullyQualifiedId(componentId);
        Map attributes = comp.getAttributes();
        return SuggestProcess.getInstance().suggest(beanClass, (String) attributes.get("attributeName"), start.toUpperCase());
    } catch (BusinessProcessException e) {
        FacesUtils.addDefaultErrorMessage(FacesContext.getCurrentInstance(), e);
        return null;
    }
}
```

```
/**
 * Method to override in order to reset all search criteria relations
 */
public void customReset() {
}
```

A change/diff

Some examples of RCS (1/4)

- CVS
 - Stands for *Concurrent Version System*
 - Centralized RCS
 - GPL license
 - <http://www.nongnu.org/cvs/>
 - Is progressively replaced by Subversion

Some examples of RCS (2/4)

- Subversion
 - Also known as SVN
 - Centralized RCS
 - APL license
 - <http://subversion.apache.org/>
 - Is now an Apache project
 - Is (has?) been widely used as a CVS replacement

Some examples of RCS (3/4)

- Git
 - GPL license
 - Distributed RCS
 - <http://git-scm.com/>
 - Used as the RCS of the Linux kernel
 - Is gaining momentum over Subversion

Some examples of RCS (4/4)

- IBM Rational ClearCase
 - Proprietary software
 - <http://www.ibm.com/software/awdtools/clearcase>
- IBM Rational Team Concert
 - Proprietary software based on said open foundations
 - <http://www.ibm.com/software/awdtools/rtc/>
 - <http://www.ibm.com/software/rational/jazz/>
 - Extends RCS to collaborative development