

VSO

Configuration Management

Timo Wolf

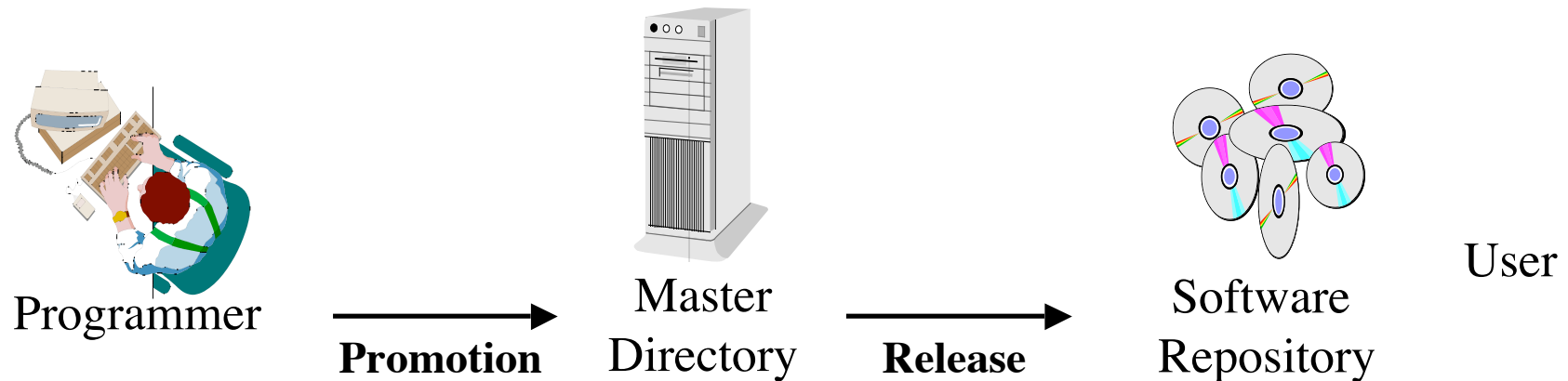
Outline

- ❖ Mapping the IEEE Standard to Subversion (SVN)
- ❖ Introduction to Subversion

- ❖ Subversion exercises
 - ◆ Exercise 1: Create your local Working Directory
 - ◆ Exercise 2: Adding your file to svn
 - ◆ Exercise 3: Check in your changes
 - ◆ Exercise 4: Update your Working Directory
 - ◆ Exercise 5: Managing concurrent changes

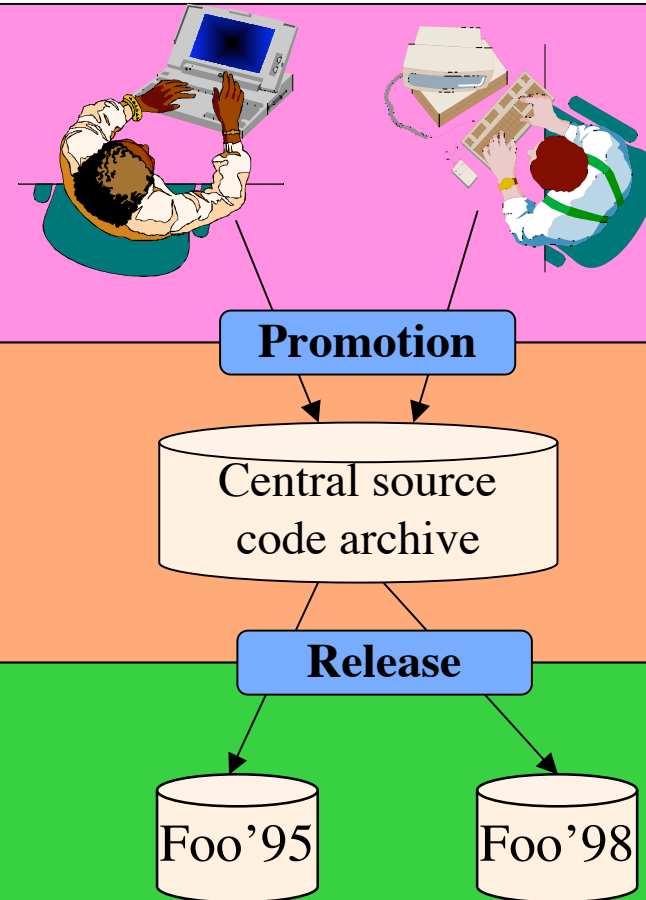
CM:Controlling Changes

- ❖ Two types of controlling change:
 - ◆ **Promotion**: The internal development state of a software is changed.
 - ◆ **Release**: A changed software system is made visible outside the development organization.

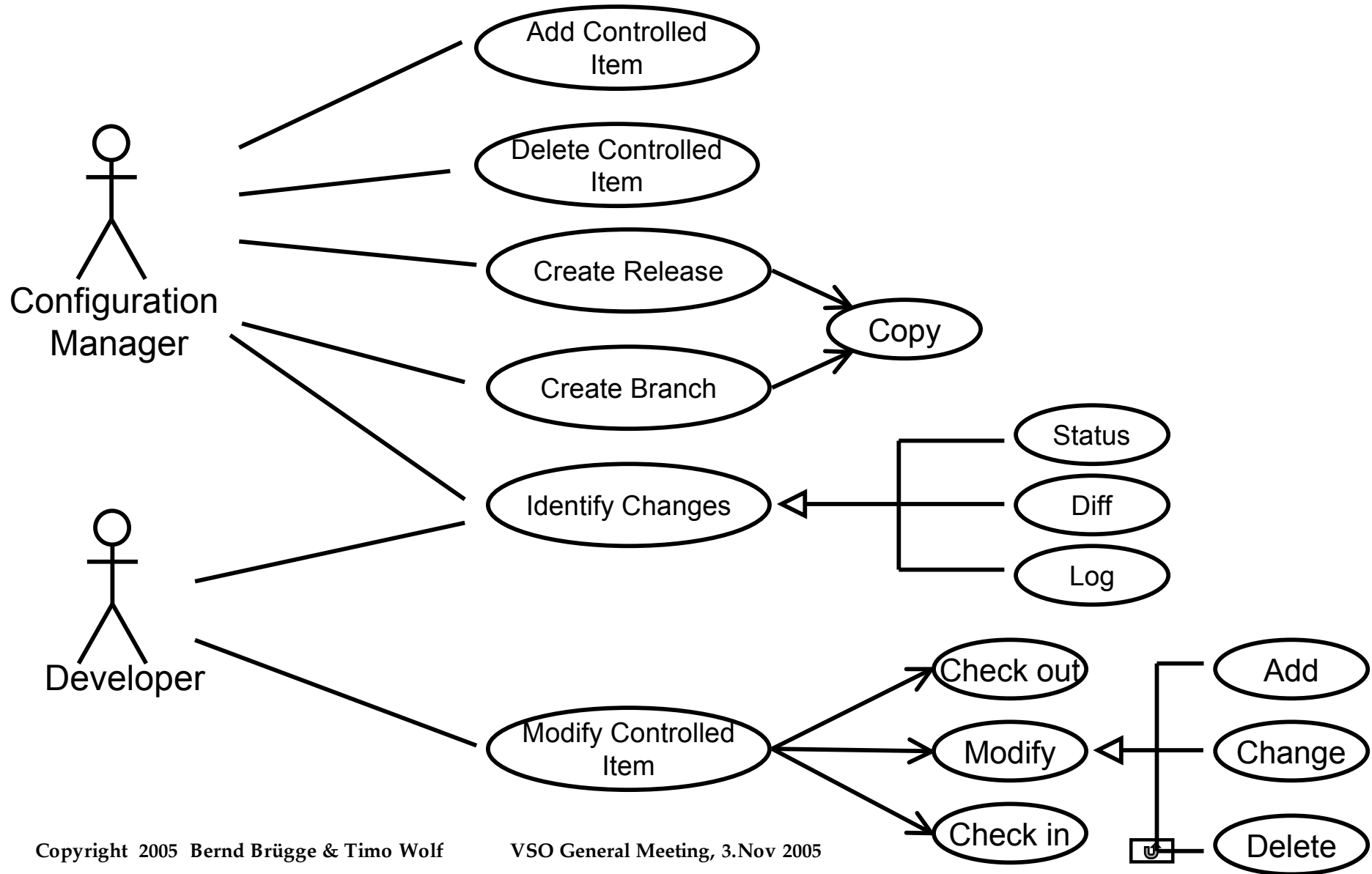


Standard SCM Directories

- ❖ Programmer's Directory
 - ◆ (IEEE Std: "Dynamic Library")
 - ◆ Completely under control of one programmer.
- ❖ Master Directory
 - ◆ (IEEE Std: "Controlled Library")
 - ◆ Central directory of all promotions.
- ❖ Software Repository
 - ◆ (IEEE Std: "Static Library")
 - ◆ Externally released baselines.



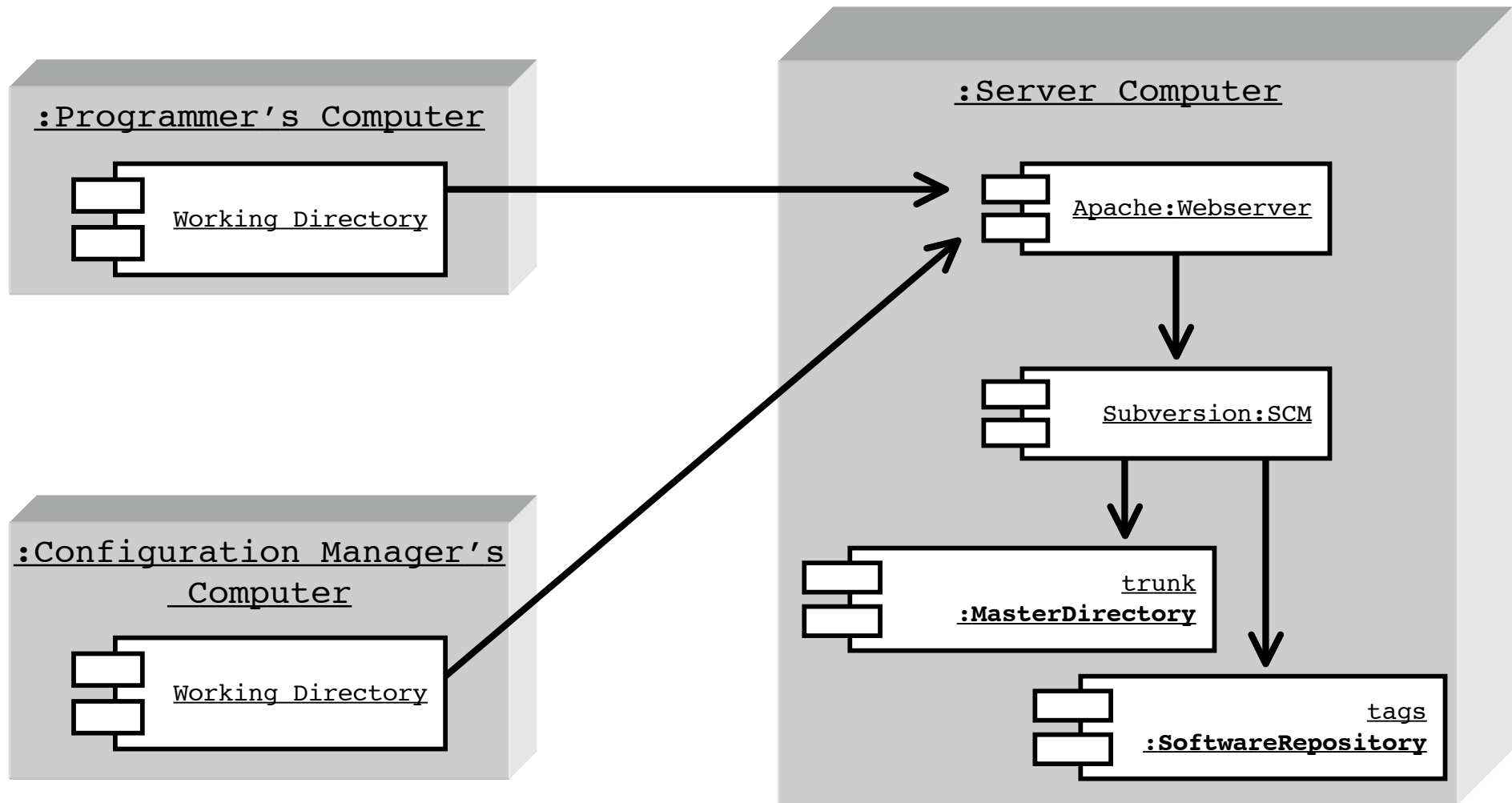
Configuration Management Use Cases



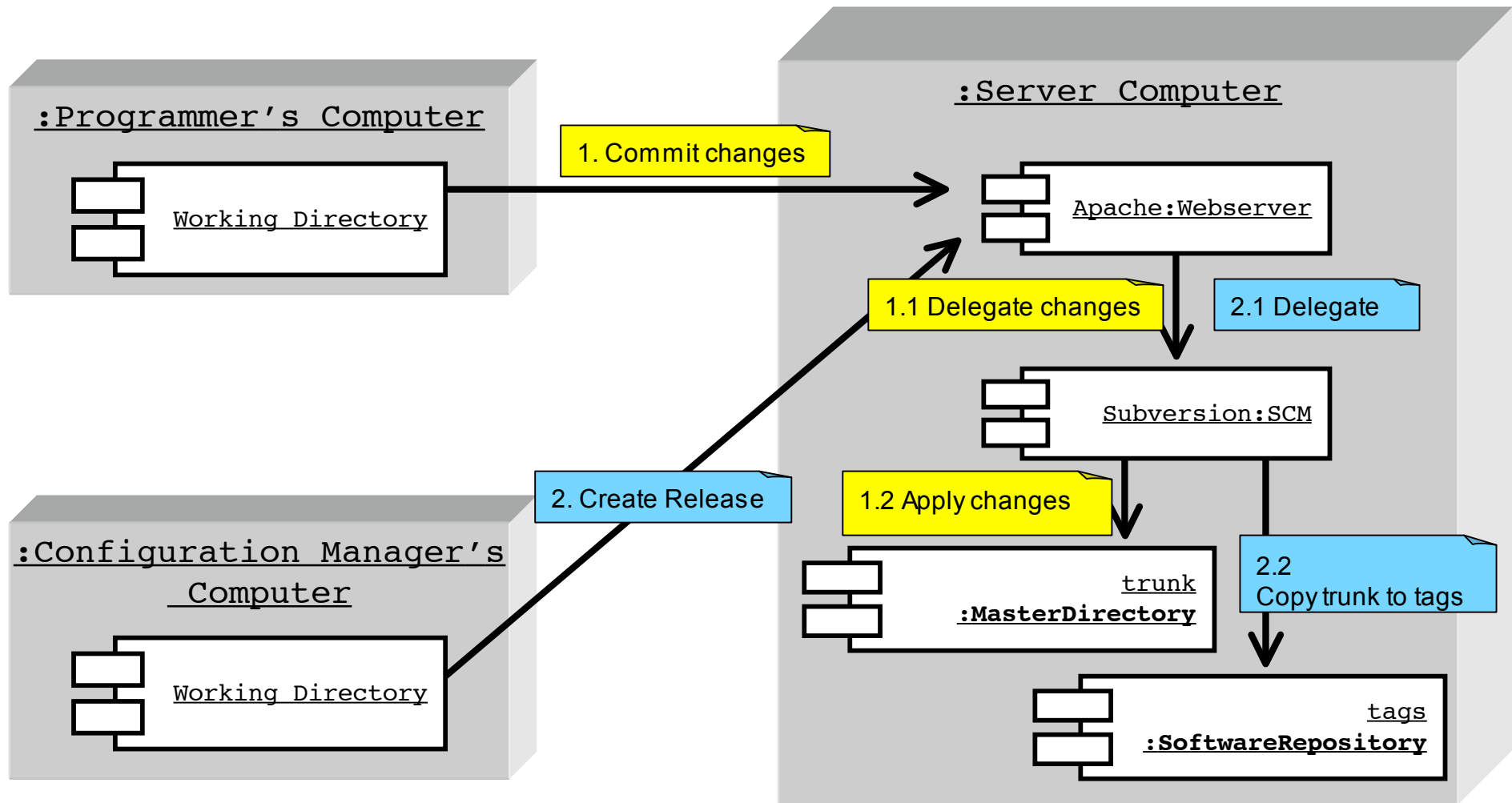
Subversion

- ❖ Subversion is an Open Source SCM
- ❖ Subversion is based on CVS
- ❖ Subversion properties:
 - ◆ Atomic commits
 - ◆ File moves and copies are recognized
 - ◆ Branching and tagging are cheap
 - ◆ Designed for the network
 - ◆ Good binary file handling
 - ◆ Directory versioning
 - ◆ Metadata

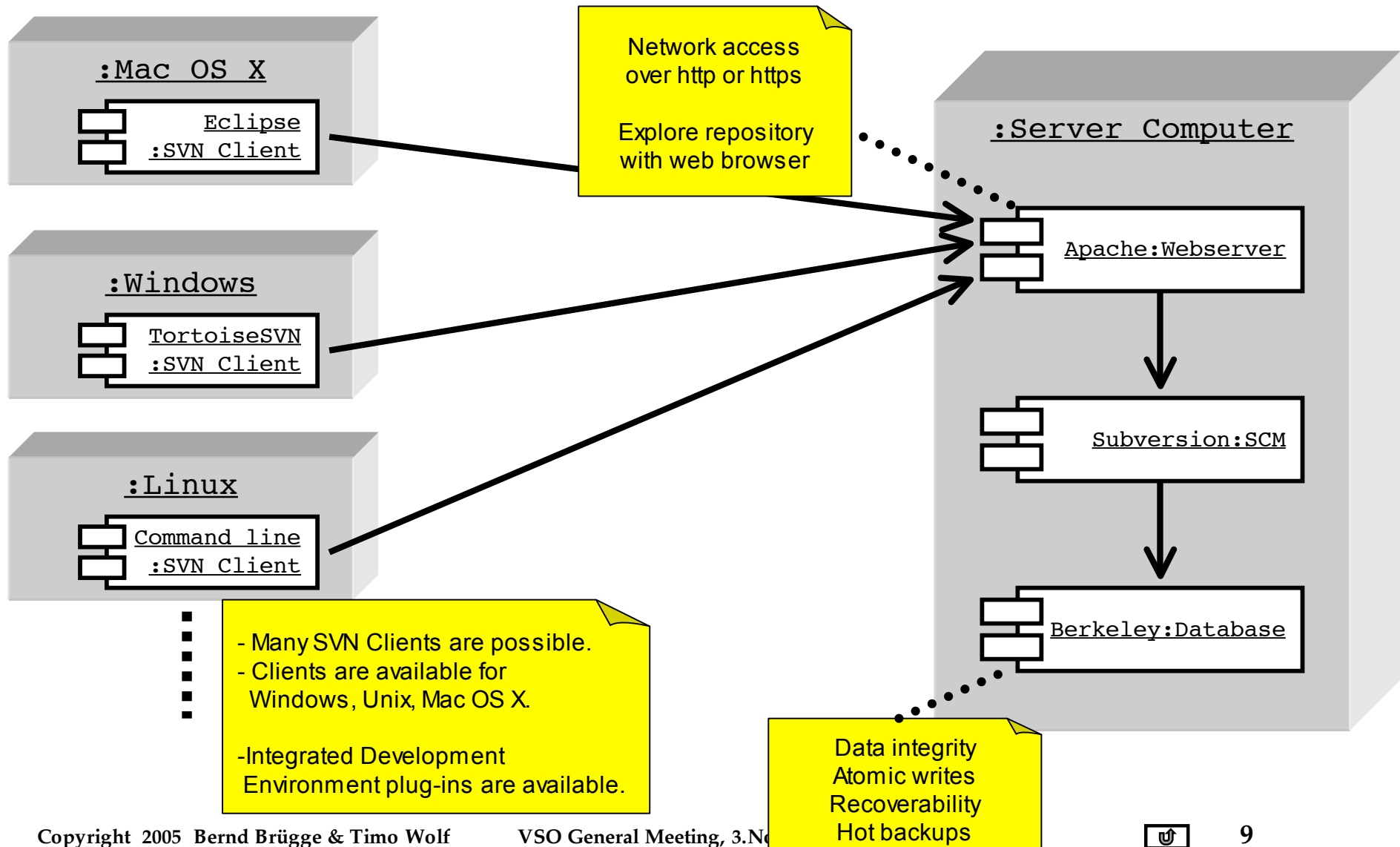
Mapping SCM Directories to Subversion



Event Flow for Use Cases 'Modify Controlled Item' and 'Create Release'



Subversion Hardware Software Mapping



VSO Master Directory

- ❖ URL for Master and Software Repository

<http://www.bruegge.in.tum.de/repos/VSO>

VSO Master Directory Structure

`http://www.bruegge.in.tum.de/repos/VSO
trunk/`

`cal/`

`dev/`

`apm/`

`architecture/`

`audio/`

`orchestra/`

`tracking/`

`ui/`

`video/`

`tutorial/`

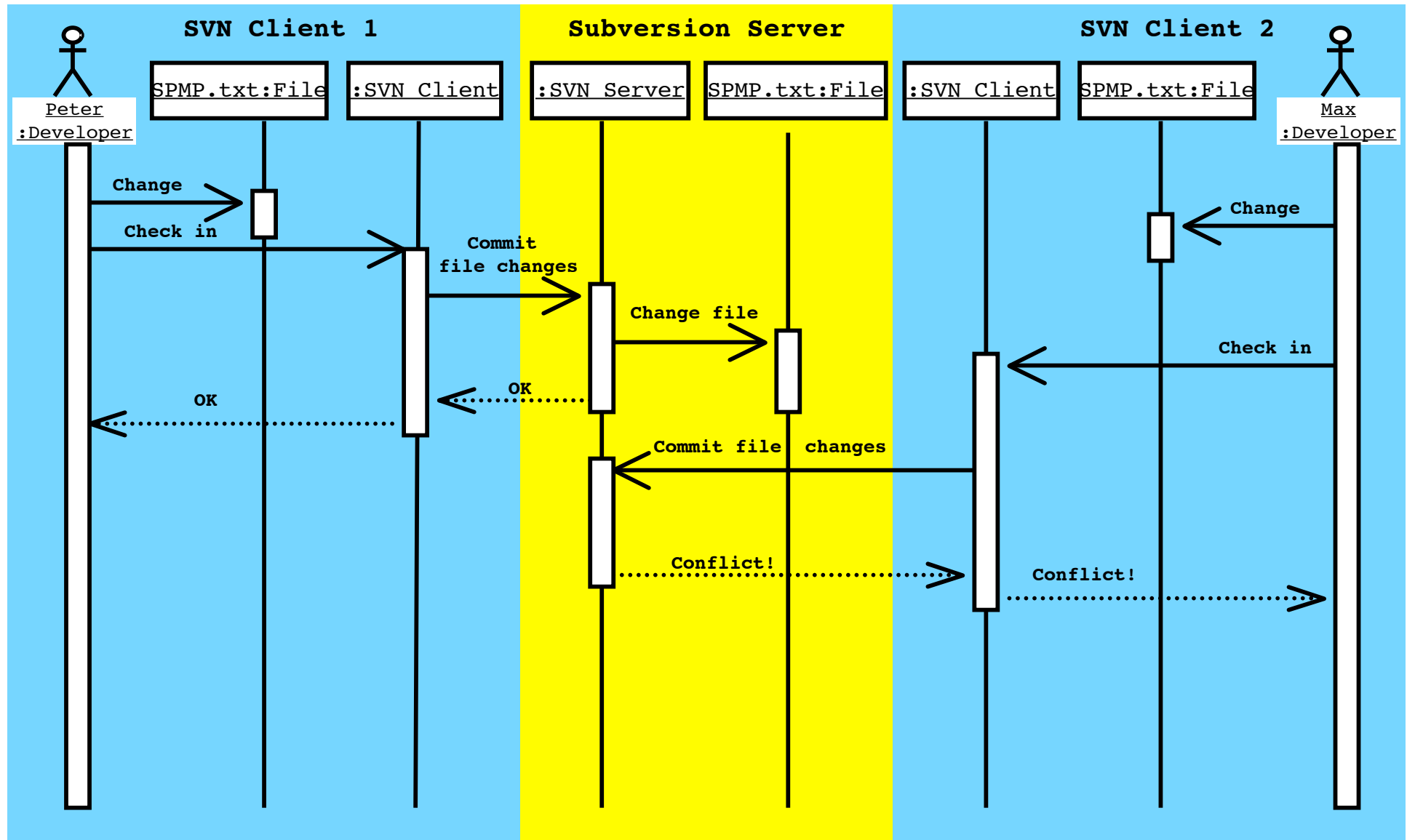
`branches/`

`tags/`

SVN Revision Numbering

- ❖ Small terminology change
 - ◆ **IEEE version => SVN Revision**
- ❖ Global revision number rather than individual numbers
- ❖ Example:
 - ◆ **Allows you to talk about “revision 050204”**
- ❖ Unique identifier for different development states of your system
- ❖ Each revision corresponds to a single check in of a **Controlled Item**
- ❖ Each revision consist of:
 - Number, author, log message, and date**
- ❖ SVN Client View is tree based, similar to names in hierarchical File System
 - ◆ **Directories are also controlled items**

Getting a Revision Conflict



Two ways of resolving conflicts in a text file

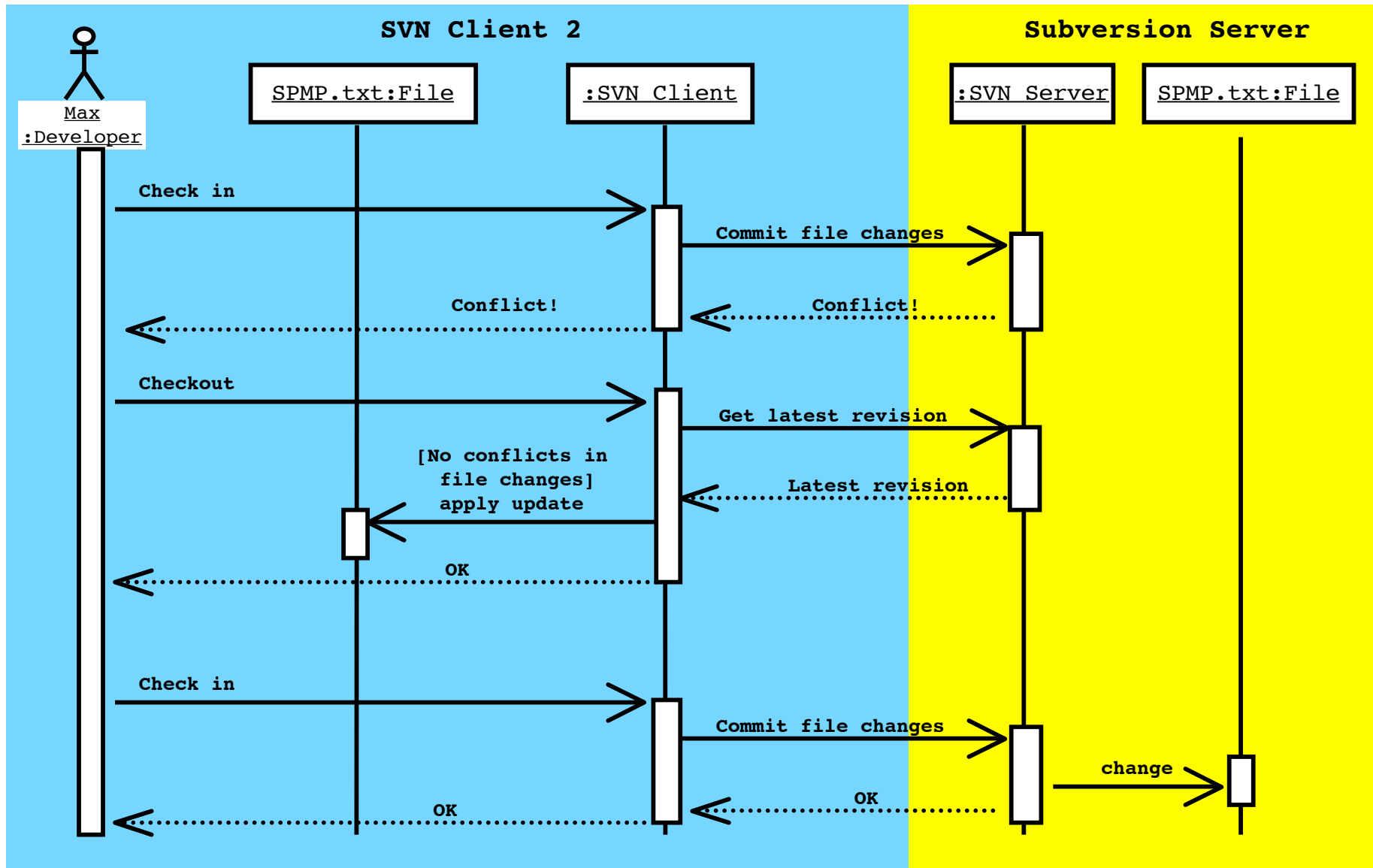
Automated Merge:

- ❖ File changes are in different lines and do not overlap
 - ◆ **Example:**
 - ◆ **Developer Peter changed the first paragraph of the SPMP**
 - ◆ **Developer Max added a new paragraph at the end of the SPMP**
- ⇒ **Subversion merges to two file versions**

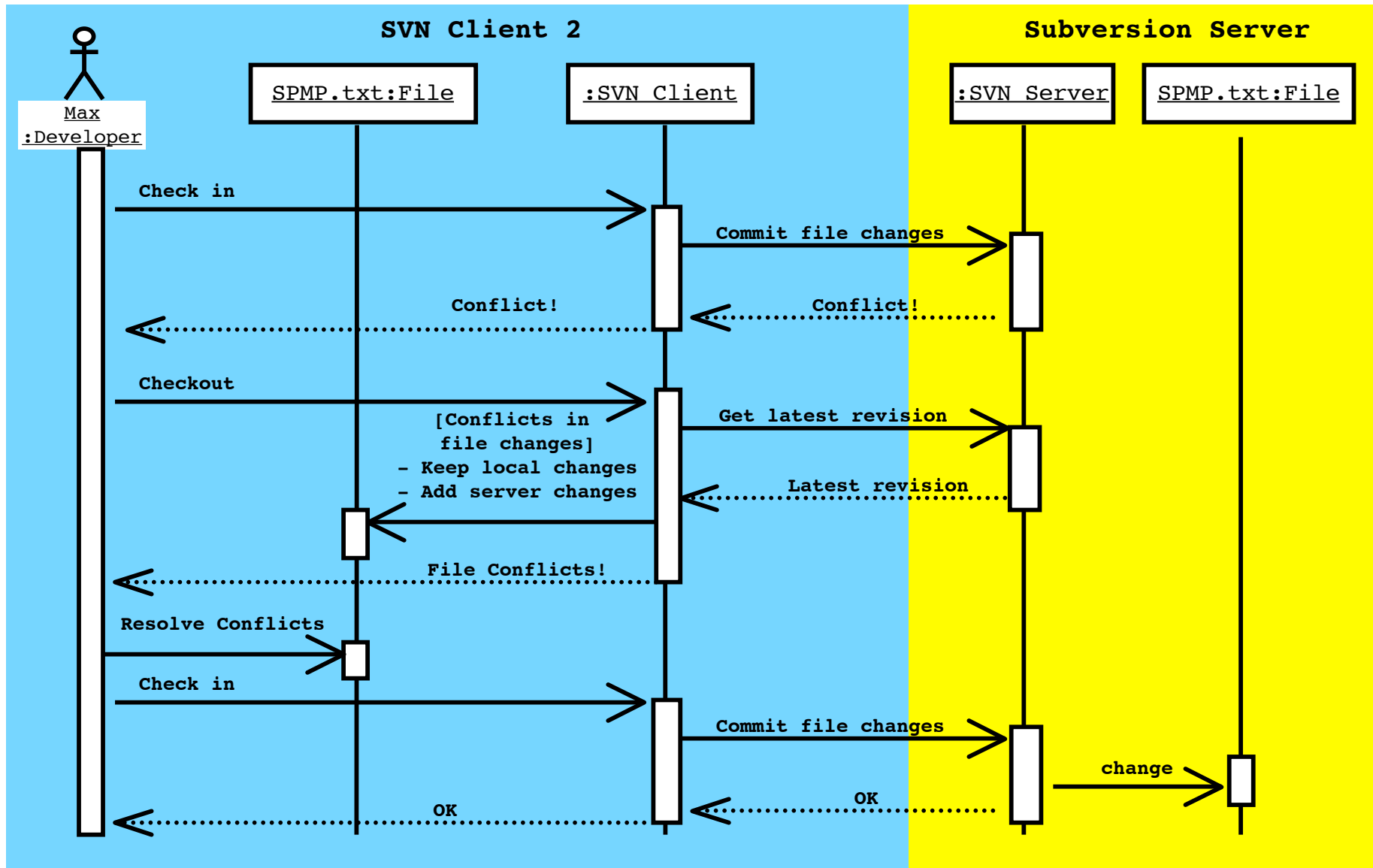
Manual Merge

- ❖ File changes are in the same lines and overlap
 - ◆ **Example:**
 - ◆ **Developer Peter changed the first paragraph of the SPMP**
 - ◆ **Developer Max deleted the first paragraph of the SPMP**
- ⇒ **The merge of the two file versions must be manual**

Resolving Conflicts I: Automated Merge



Resolving Conflicts II: Manual Merge



Extended Subversion Documentation

❖ SVN Book

- ◆ <http://svnbook.red-bean.com/>
- ◆ PDF and HTML Format

❖ Subversion Home page

- ◆ <http://subversion.tigris.org/>