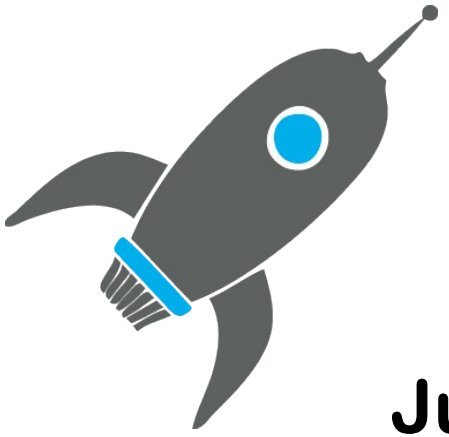
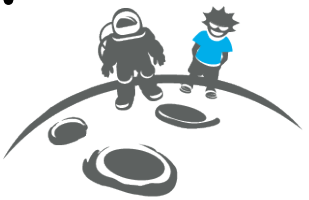
GearConf 2012
Using Subversion and Git Together



June 25th 2012, Dusseldorf



tmate
software
dream driven development

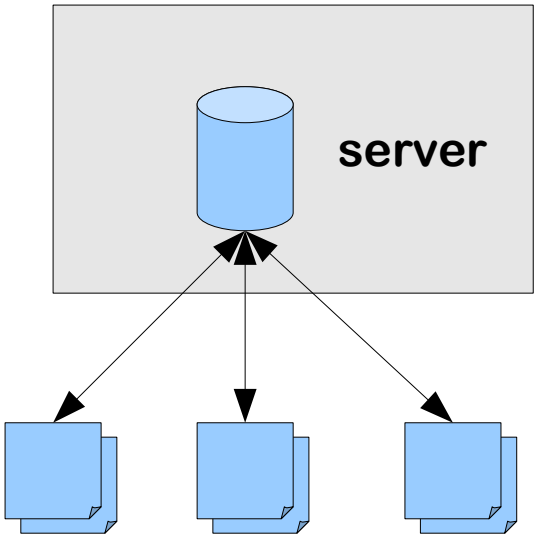
About Us

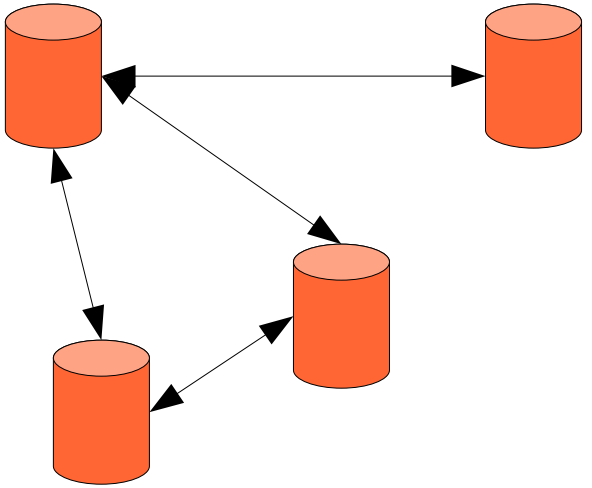
- TMate Software
- We create VCS tools since year 2003
- Products at pre-release stage
  - SubGit (RC, self-hosting for 6 months)
  - HG4J (Beta)
- Our mature products:
  - SvnKit (1.7.4, 30% of the ASF requests)
  - SqlJet (1.1.2)

Subversion and Git

- Subversion:
  - Mature, well known and widely used
  - Centralized
  - Manager-friendly
- Git:
  - Fast growing VCS
  - Distributed
  - Geek-friendly

Subversion and Git





- Single repository at shared centralized server
- All branches and forks are in the same repository

- Many repositories
- Forks
- Delayed commits

Subversion and Git

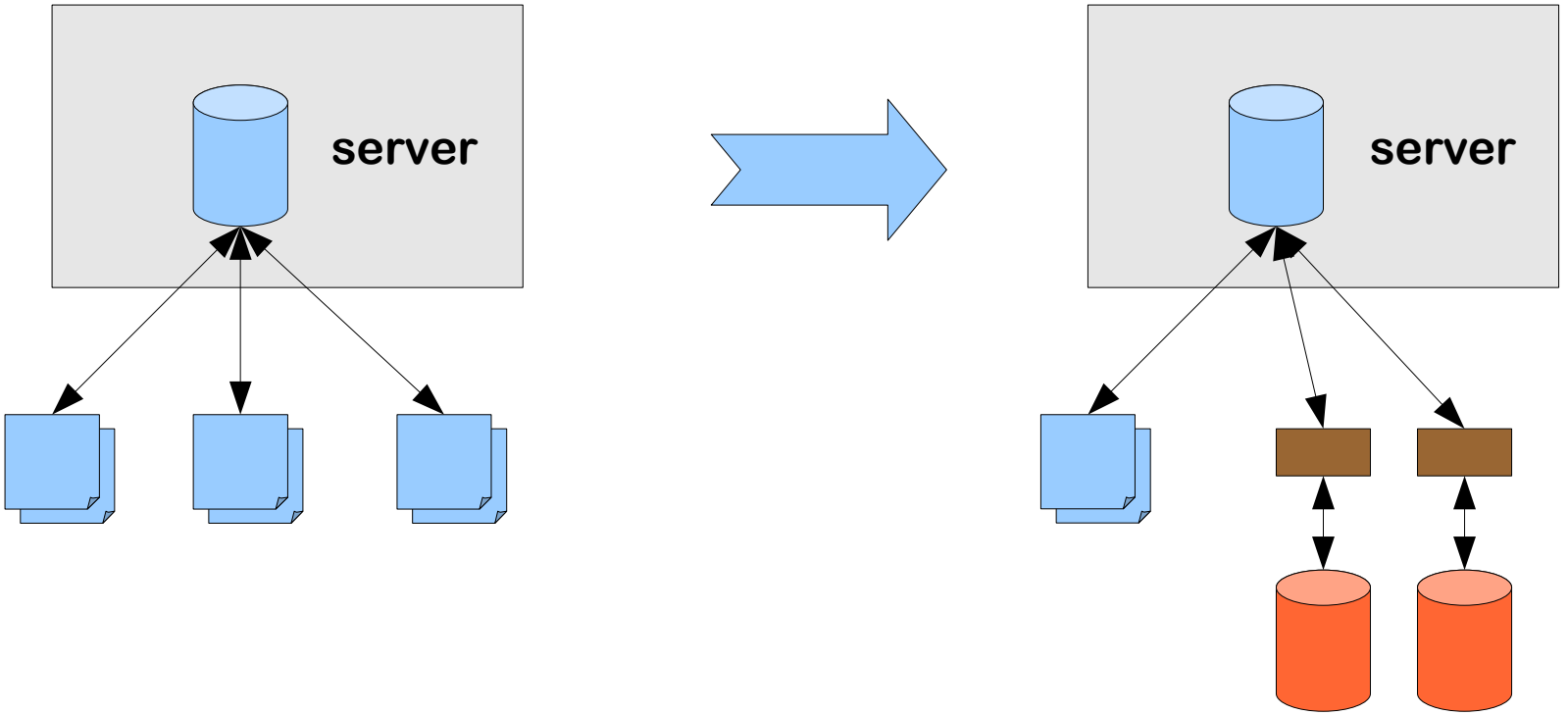
How is it possible to use both?!

Two Approaches

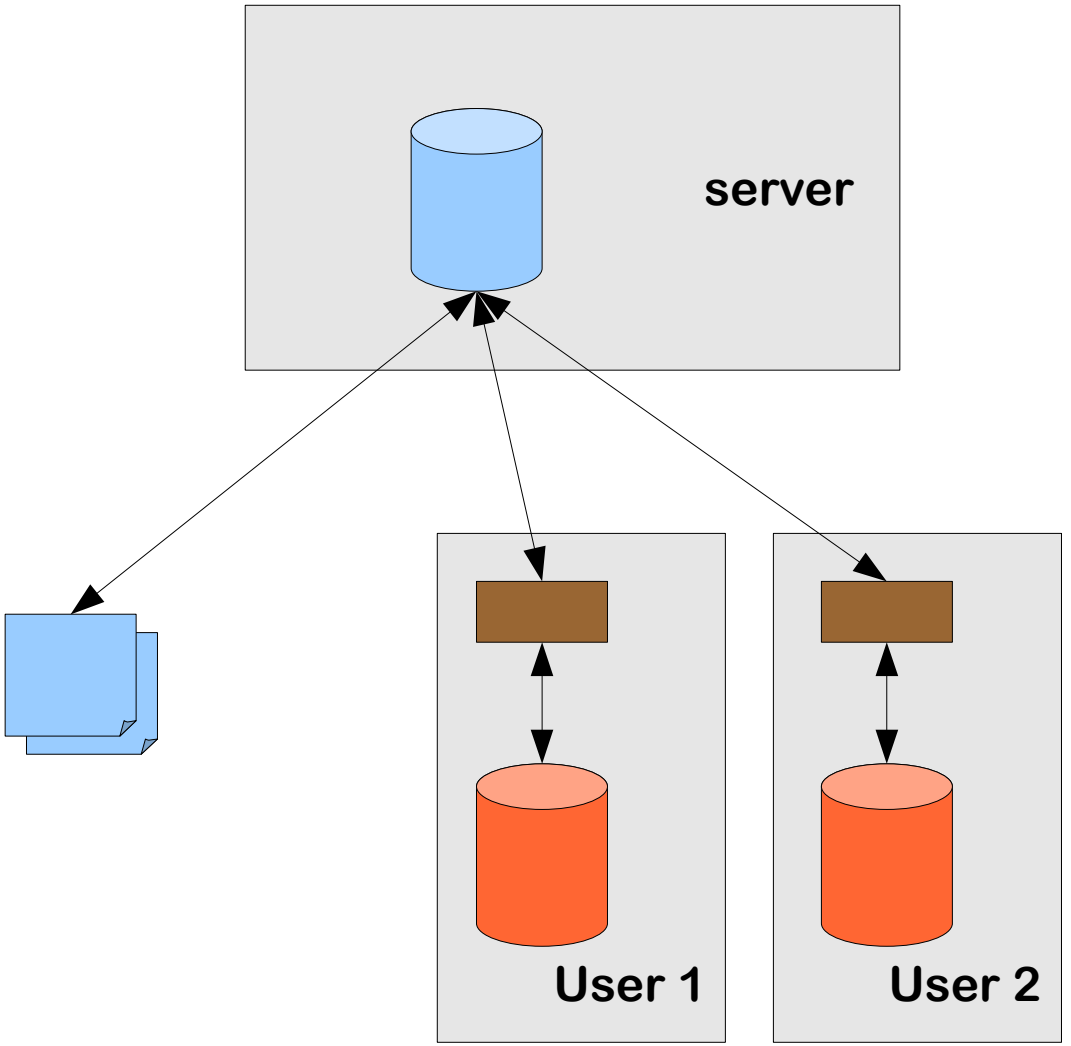
Git-Svn

or SubGit

Git-Svn Approach
overview

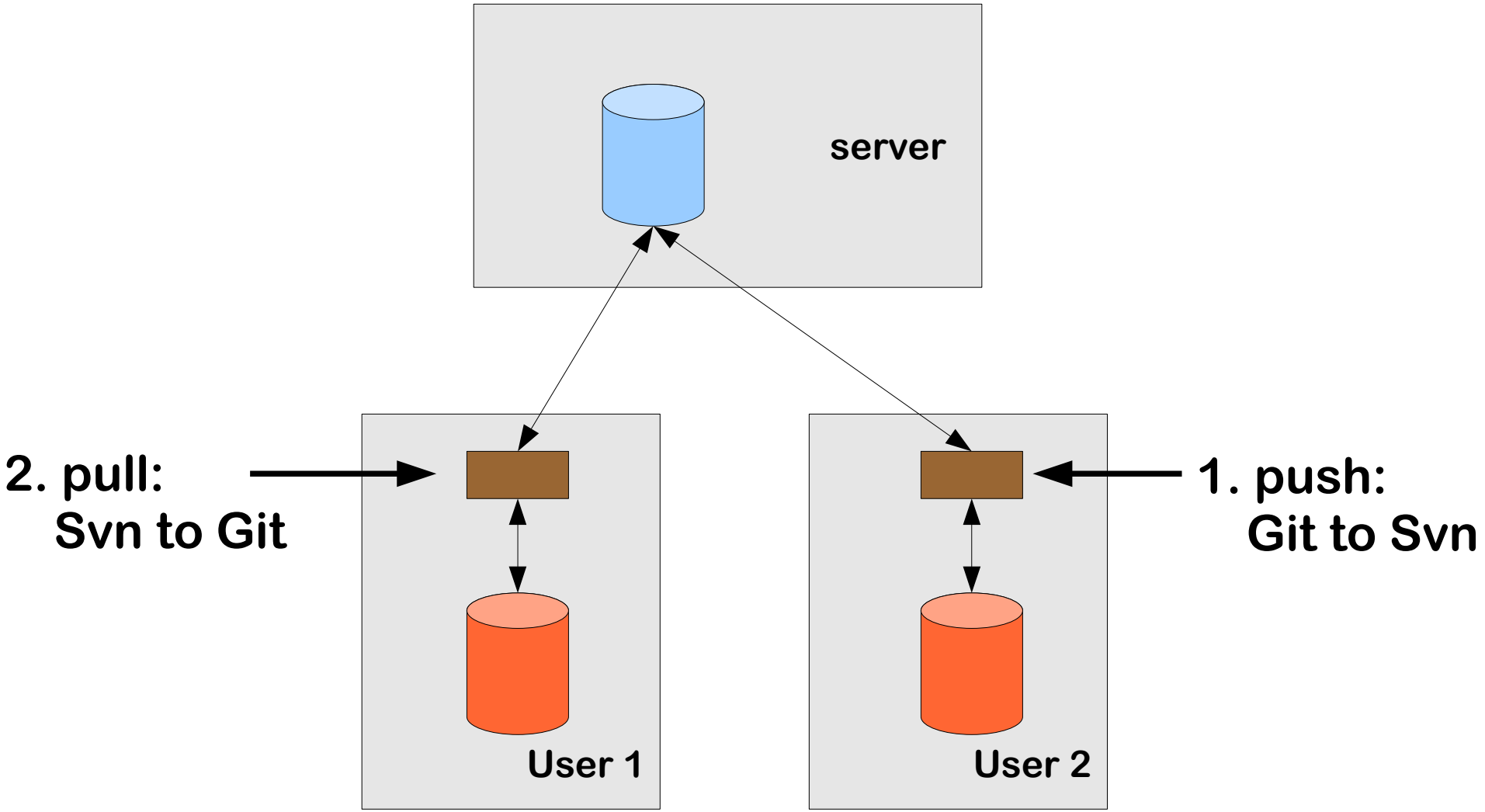


Git-Svn Approach
client-side, partisan deployment

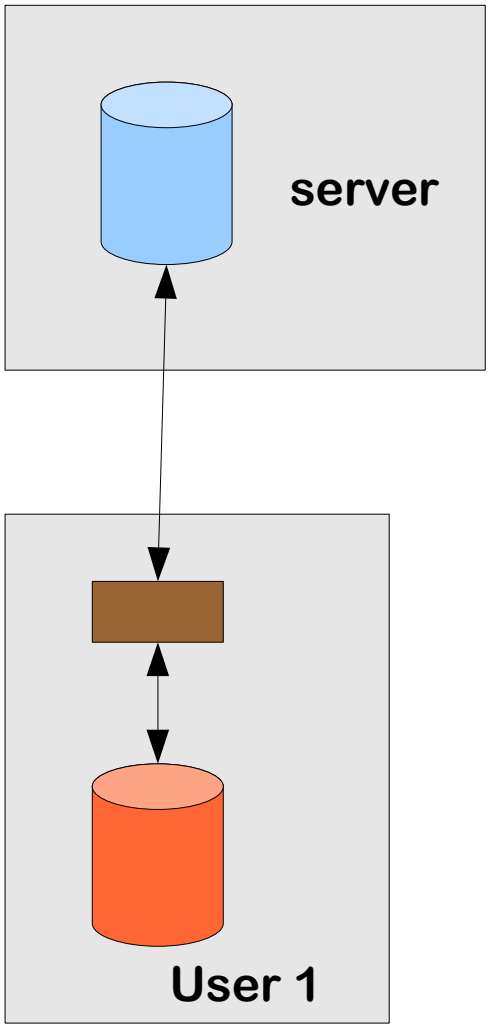


Git-Svn Approach

double translation, resulting in data loss



Git-Svn Approach
custom non-Git workflow



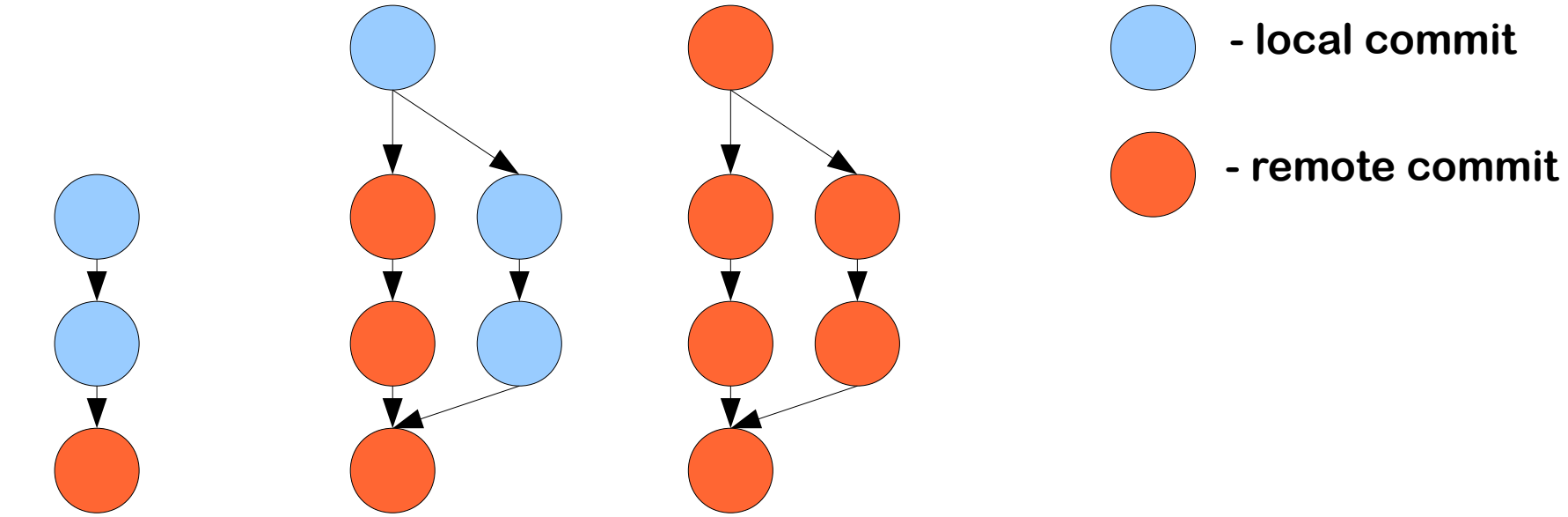
Git-Svn commands:

- `fetch`
- `dcommit`
- `rebase`
- `branch`

...

Git-Svn Approach
custom non-Git workflow

Standard Git push:



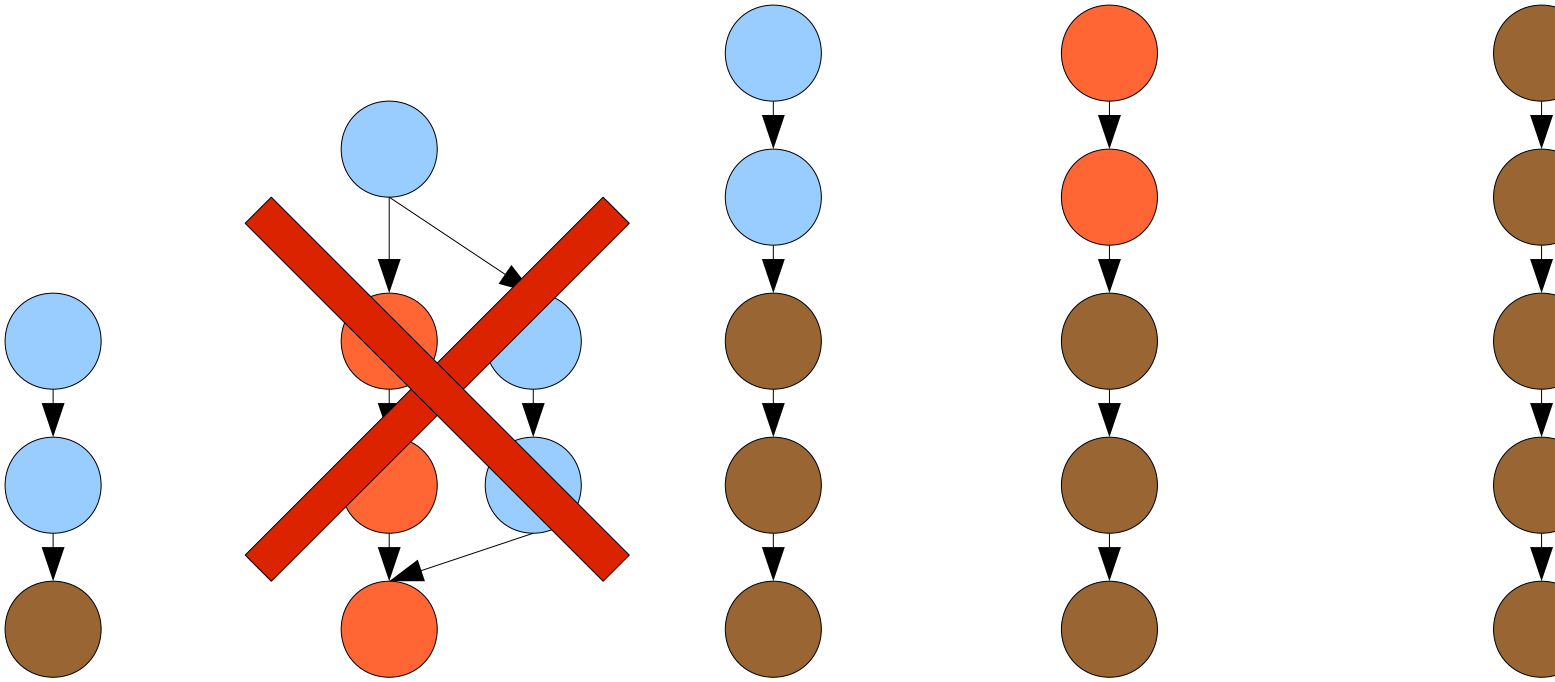
local state

git pull,
merge commit created

git push,
new commits pushed

Git-Svn Approach
custom non-Git workflow

Git-Svn dcommit:



local state

git svn dcommit,
requires rebase first

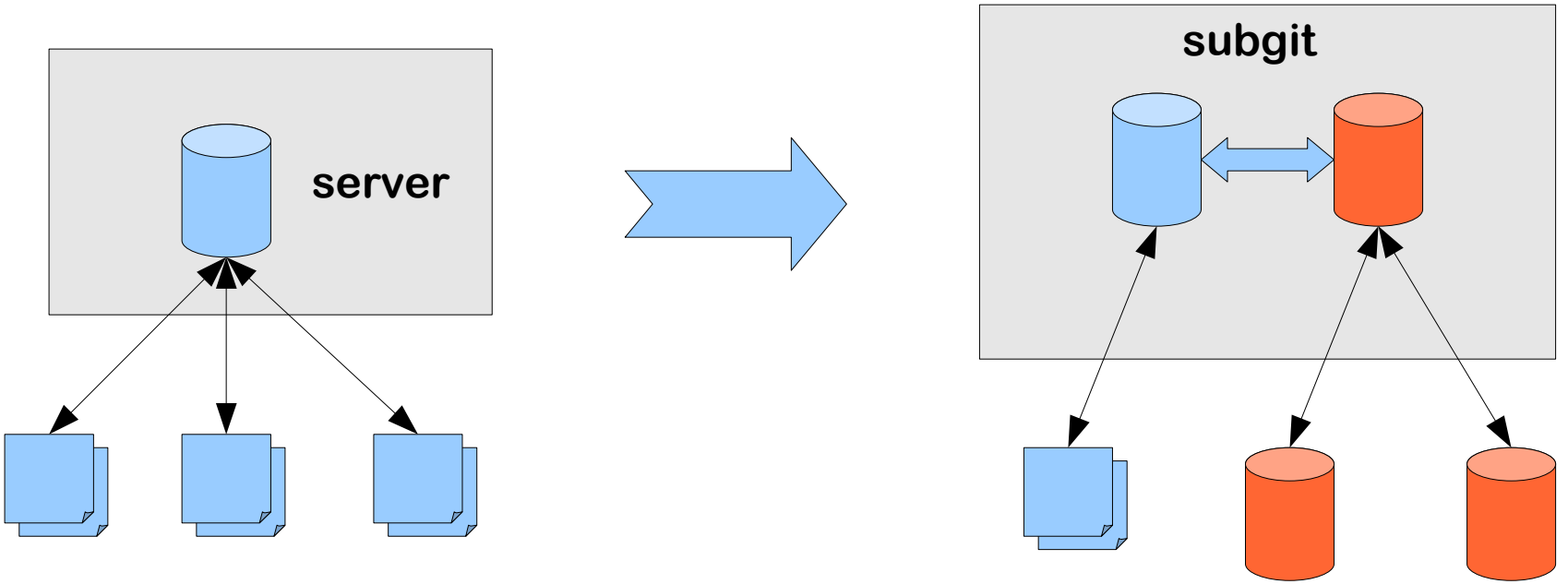
git svn dcommit,
(commit)

git svn dcommit,
(fetchback, sign)

Git-Svn Approach
summary

- Client side
- Each user needs to translate at least part of the repository
- Non-standard workflow
- Part of standard Git package, but...
- ...git-svn is 5000 lines script of perl code

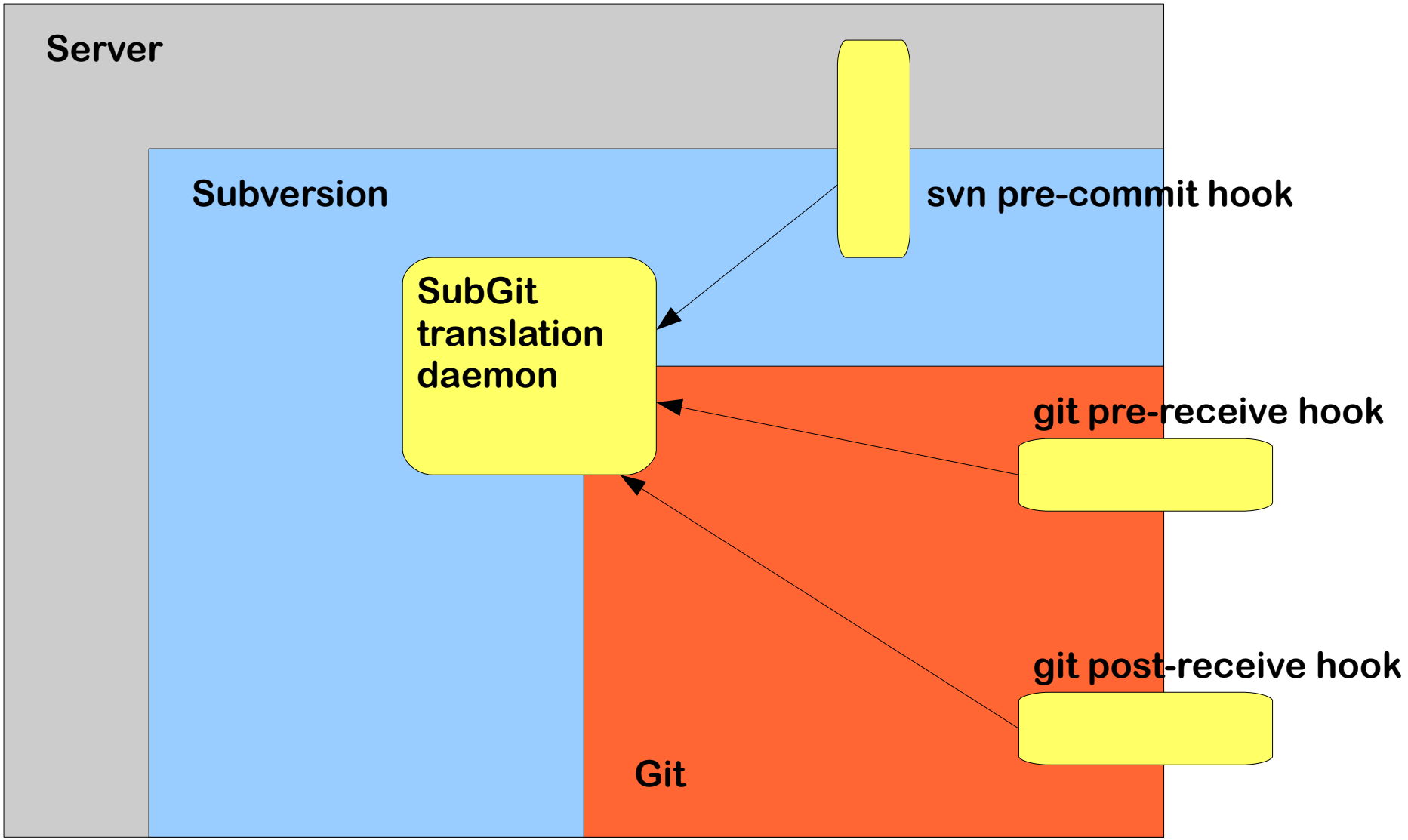
SubGit
overview



SubGit
approach details

- Installed *into* repository
- Uses hooks
- Translation is performed in the background to minimize overhead

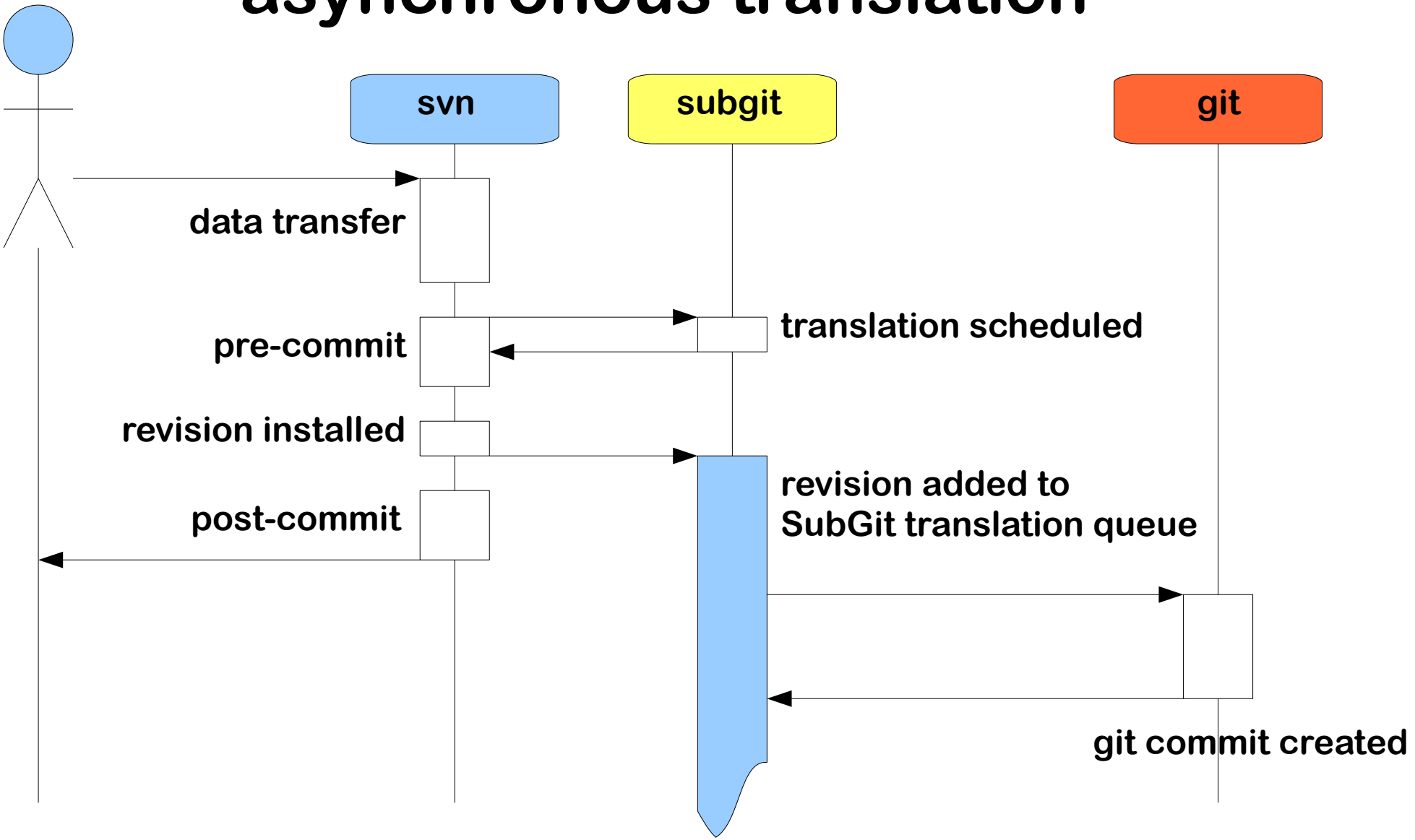
SubGit
default installation



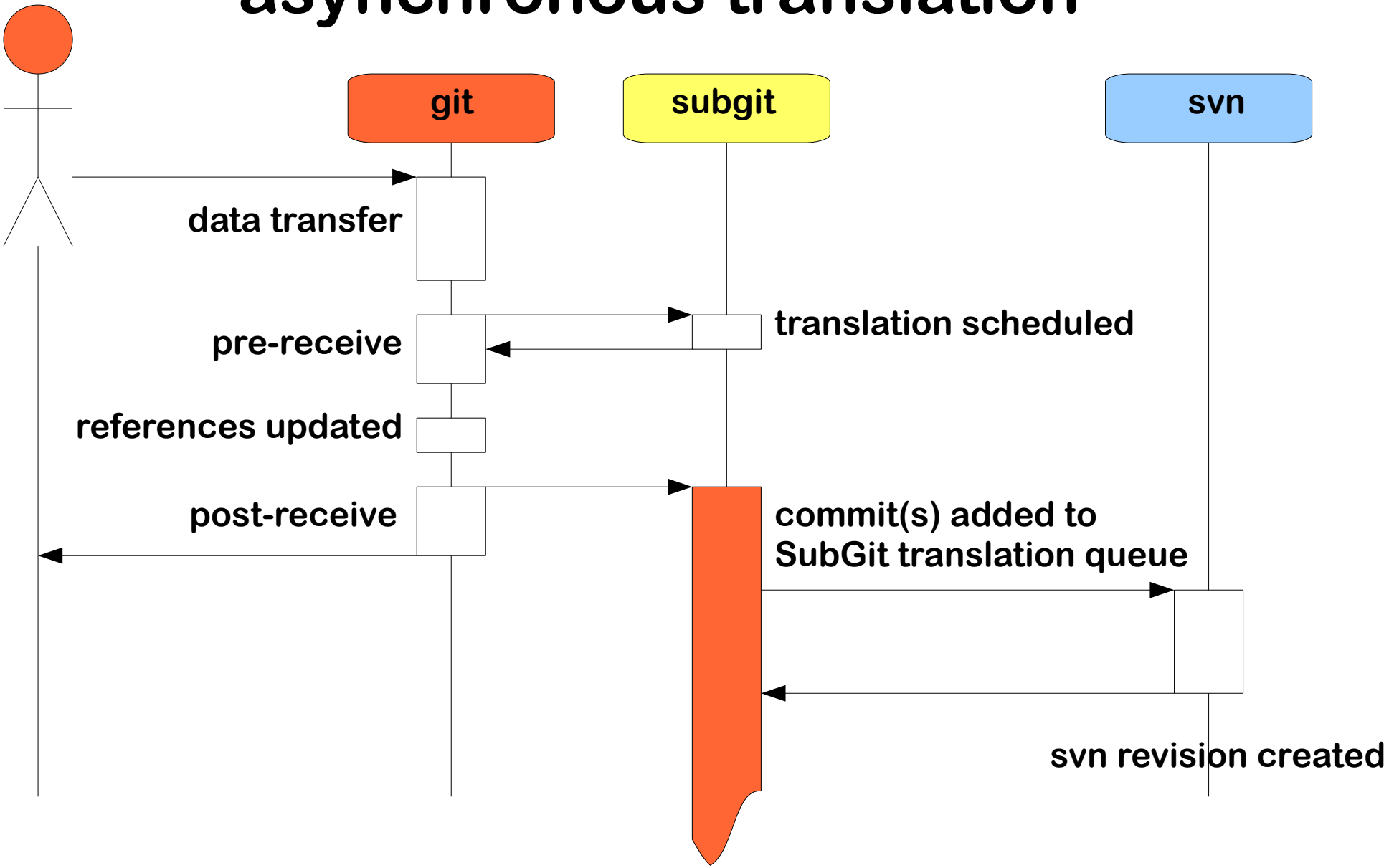
SubGit
Apache configuration example

```
<Location /svn>
    # http://repos.host.com/svn/project
    DAV Svn
    SVNParentPath /var/svn/repos

    Require valid-user
    AuthType Basic
    AuthUserFile /var/svn/repos/passwords
</Location>

<Location /git>
    # http://repos.host.com/git/project
    ScriptAlias /git/ …
    …
    Require valid-user
    AuthType Basic
    AuthUserFile /var/svn/repos/passwords
</Location>
```

SubGit
asynchronous translation

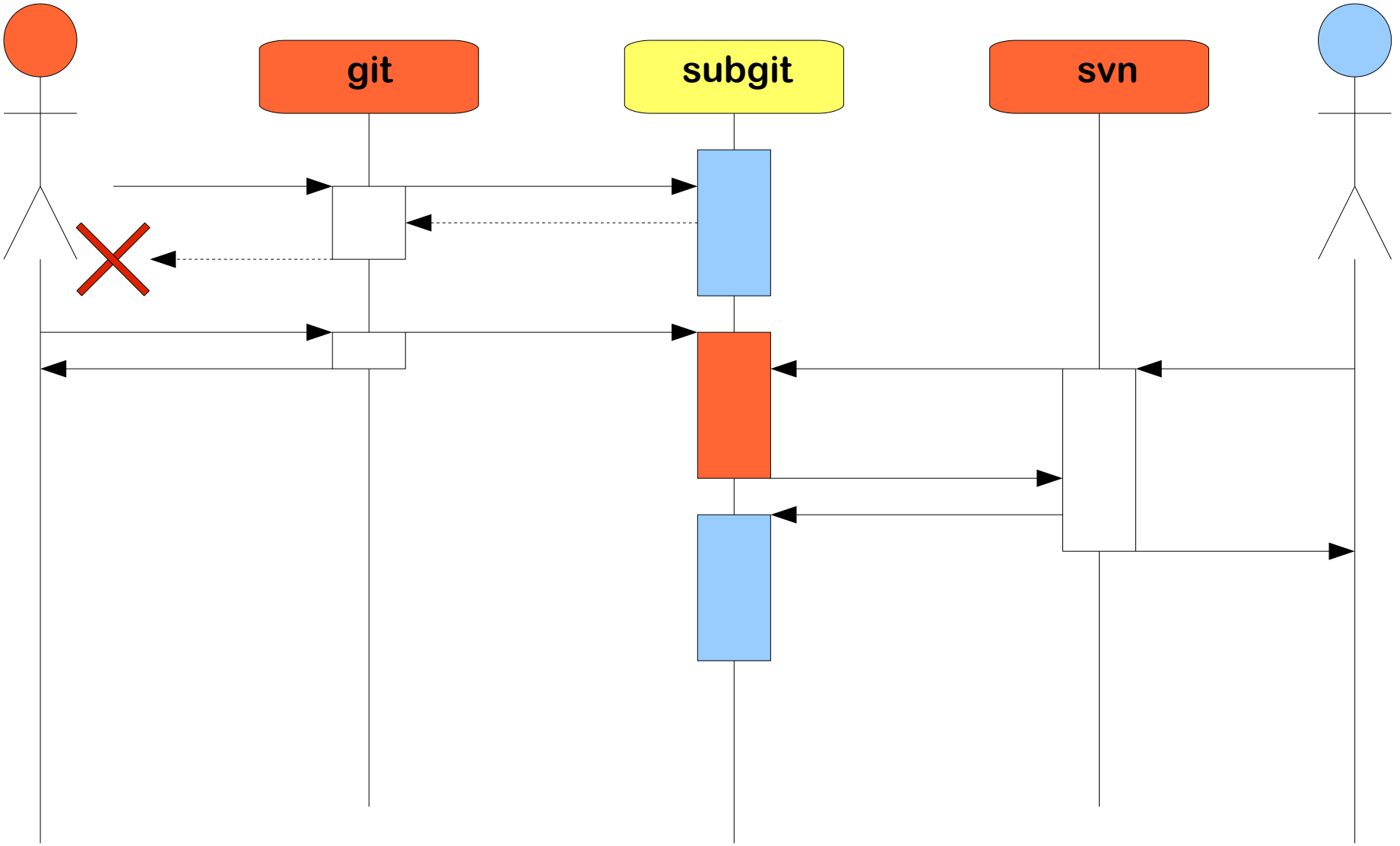


SubGit
asynchronous translation



SubGit

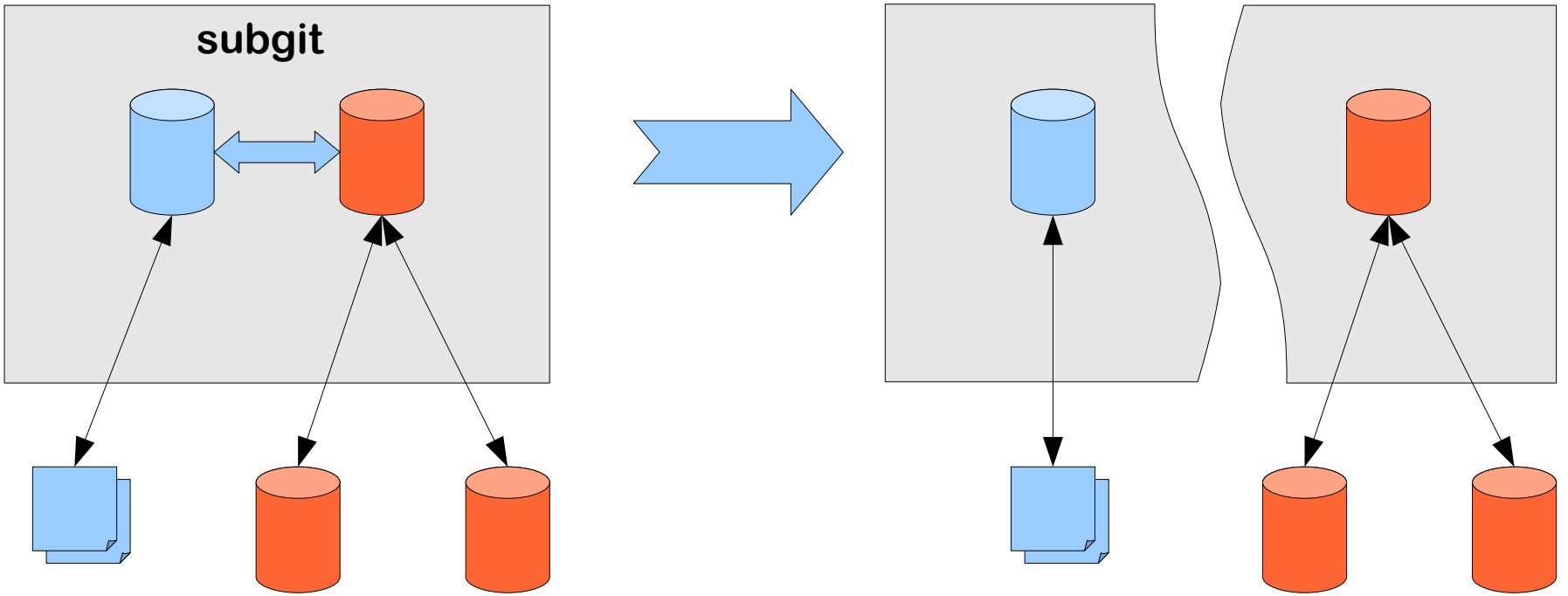
asynchronous translation



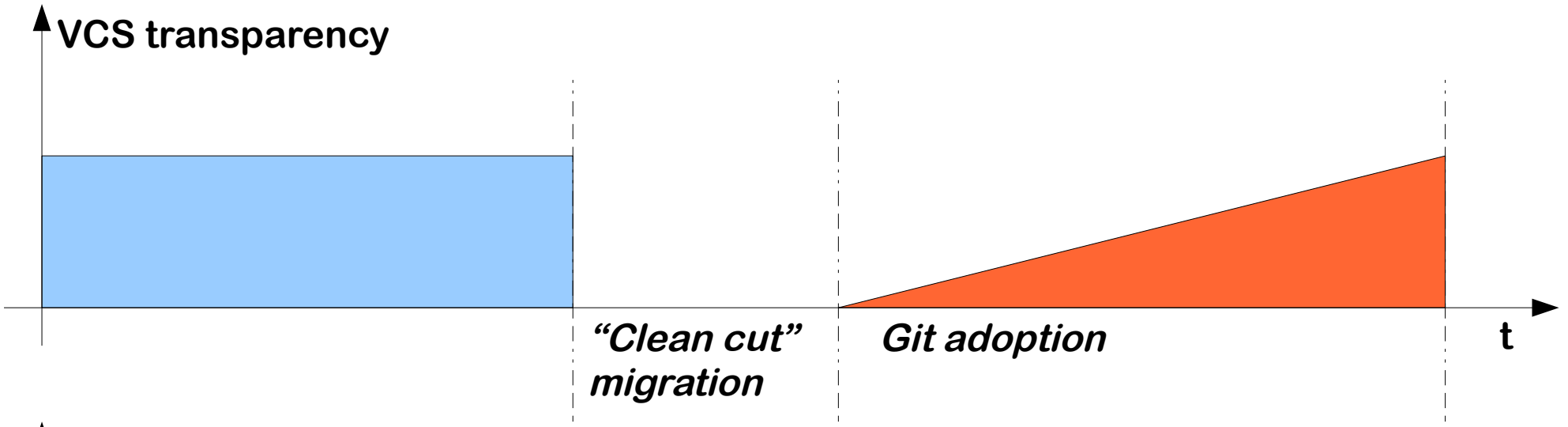
SubGit
summary

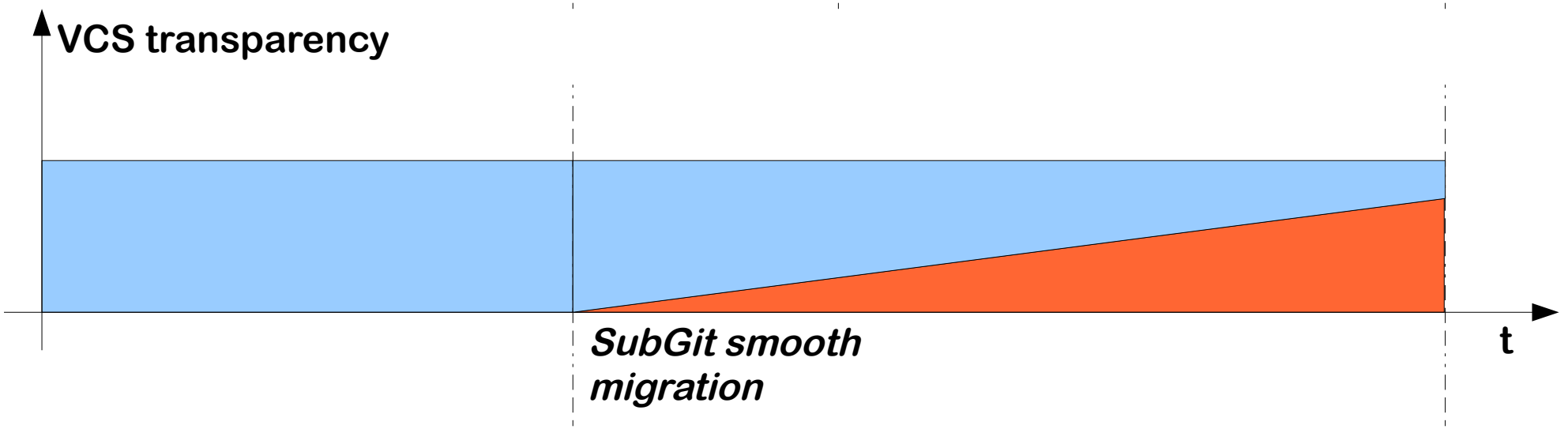
- Server side
- One-time centralized deployment
- Reuse of existing infrastructure
- Pure Git/Subversion experience
- Safe and smooth migration
- Commercial quality tool

SubGit
safe migration



SubGit
smooth migration





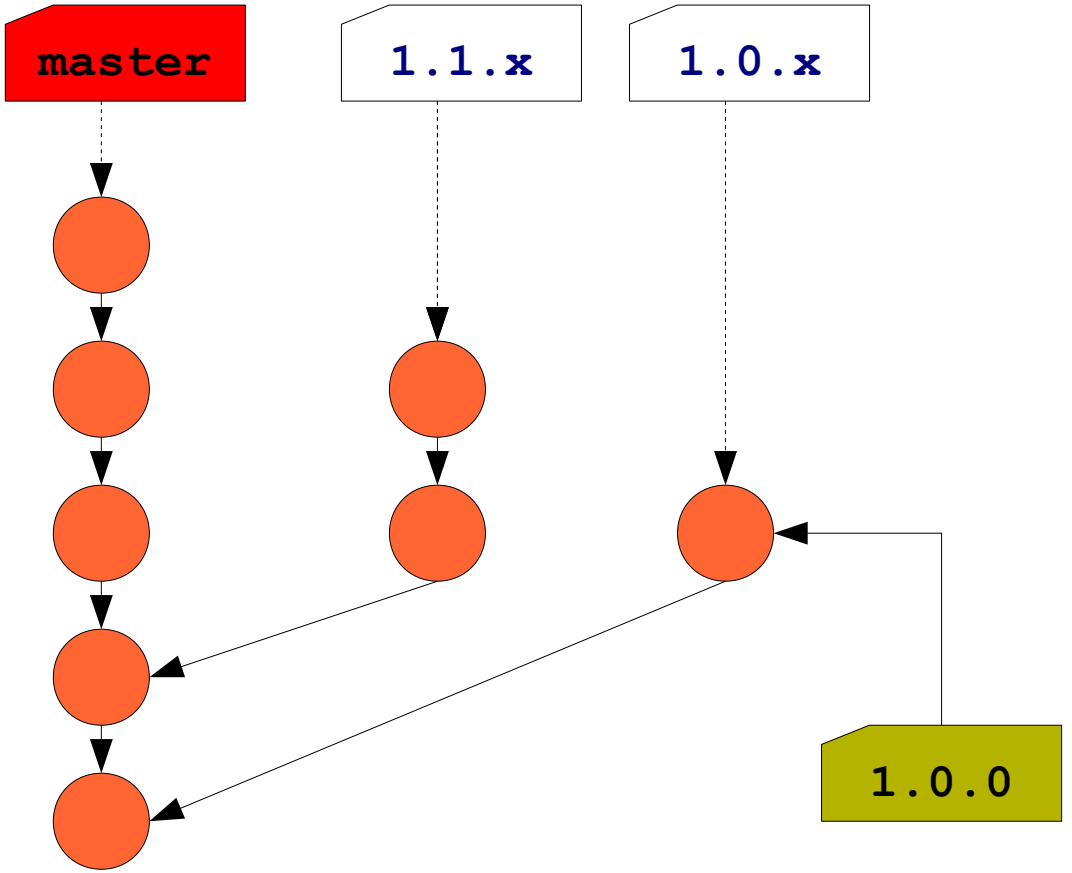
Subversion<=>Git Translation

- Branches and Tags
- Deltas
- Special properties
- Merge tracking info

- References
- Trees
- Special files
- Commit parents

Branches and Tags
overview

```
/trunk

/branches
/branches/1.0.x
/branches/1.1.x

/tags
/tags/1.0.0
```

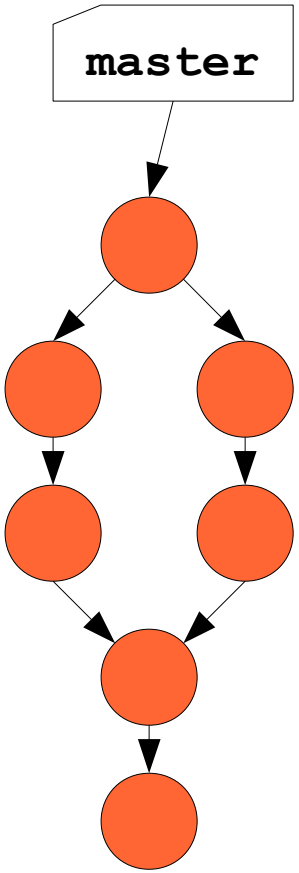



```
--- r10
M /branches/1.0.x/file.txt
```

Branches and Tags
mapping

```
trunk = trunk:refs/heads/master
branches = branches/*:refs/heads/*
tags = tags/*:refs/tags/*

shelves = shelves/*:refs/shelves/*
```

Branches and Tags

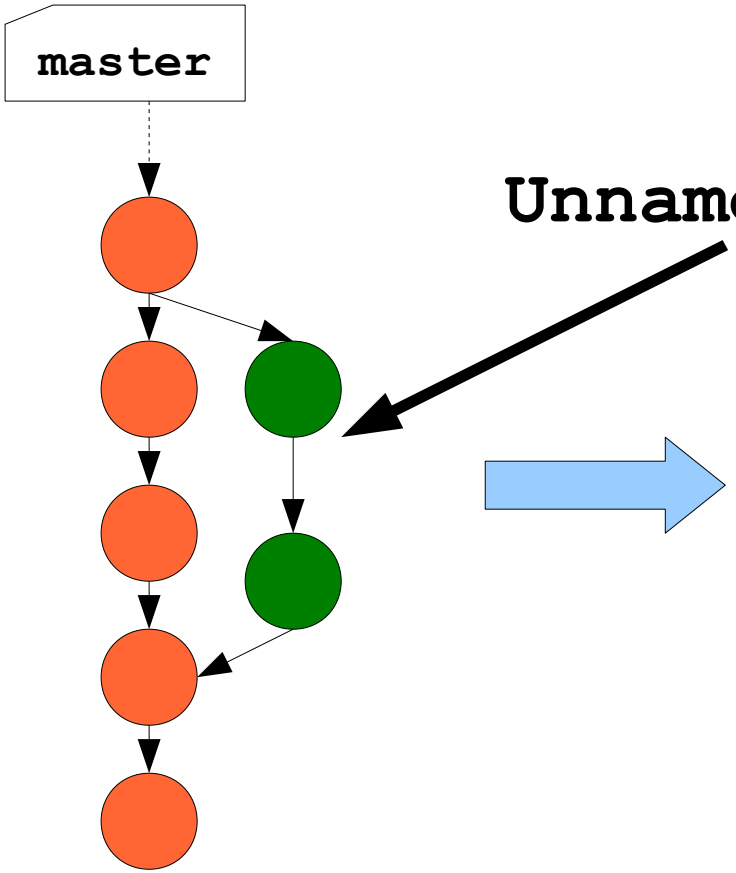


Which history line is a trunk?

And which is a branch?

Heuristics help!

Branches and Tags
shelves

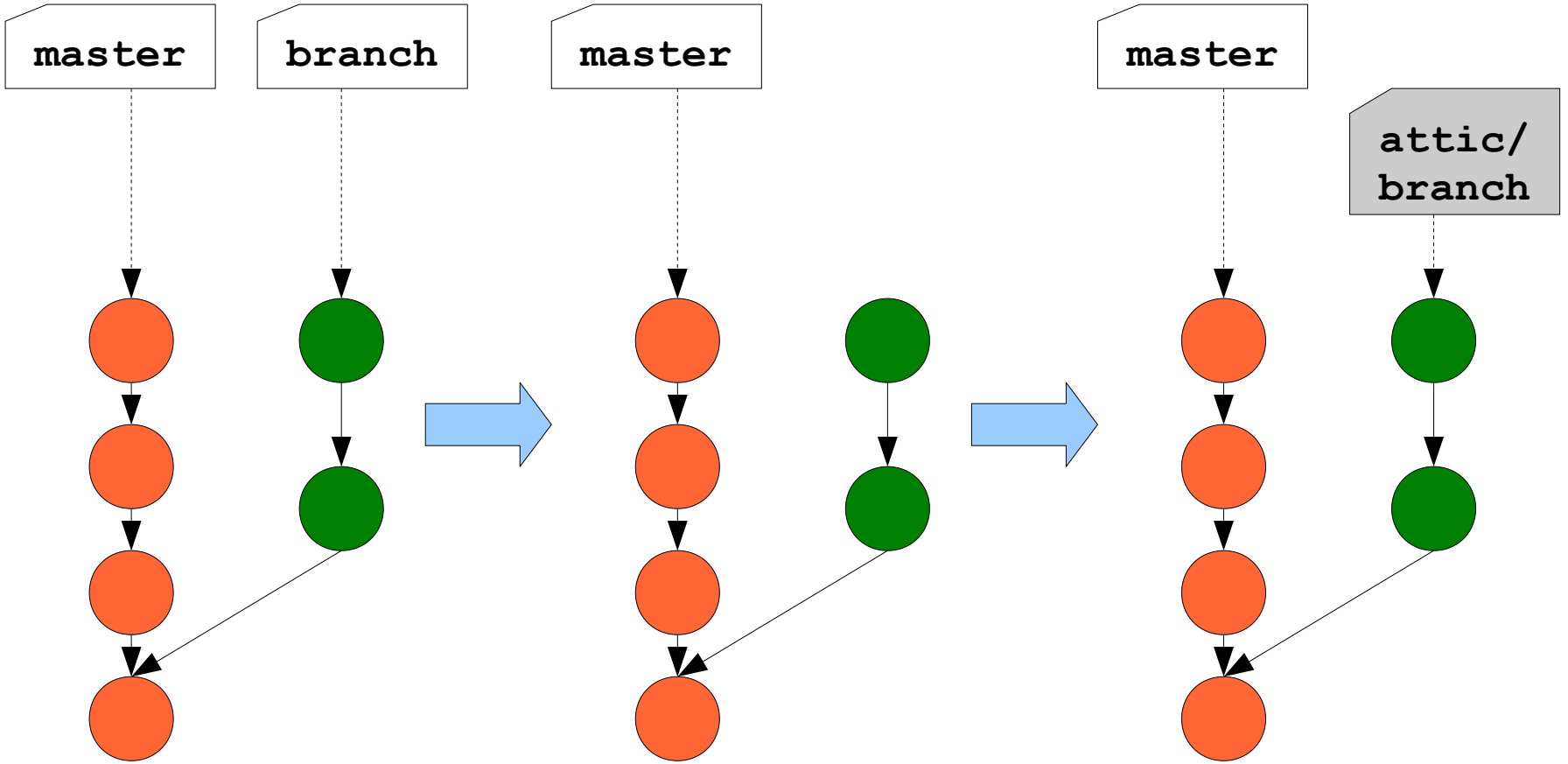


Unnamed Git branch

/shelves/author[_date]

```
--- r20
M /shelves/alex/file.txt
```

Branches and Tags
attic references



Braches and Tags

- From Subversion to Git
  - Single Subversion revision might be translated to multiple Git commits
- From Git to Subversion
  - Heuristics help to detect target branch in complex cases
  - /shelves directory is used to store anonymous branches

Deltas and Trees

- Subversion revision is a delta:

```
--- r20
A /trunk/file.txt
M /trunk/dir/file.txt
A /trunk/dir/copy.txt
        from /trunk/file.txt@r10
D /trunk/dir/old_file.txt
```

- Delta tracks copies
- Git commit is always a full file tree

Deltas and Trees

- From Subversion to Git
  - Relevant part of the delta is applied to the parent commit tree
- From Git to Subversion
  - Renames and copies are detected at translation time
  - Branches and tags creation uses copy operation

Special Properties

- properties:
  - svn:ignore
  - svn:eol-style
  - svn:mime-type
- files:
  - .gitignore
  - .gitattributes

file.txt

```
svn:eol-style=native
```

.gitattributes

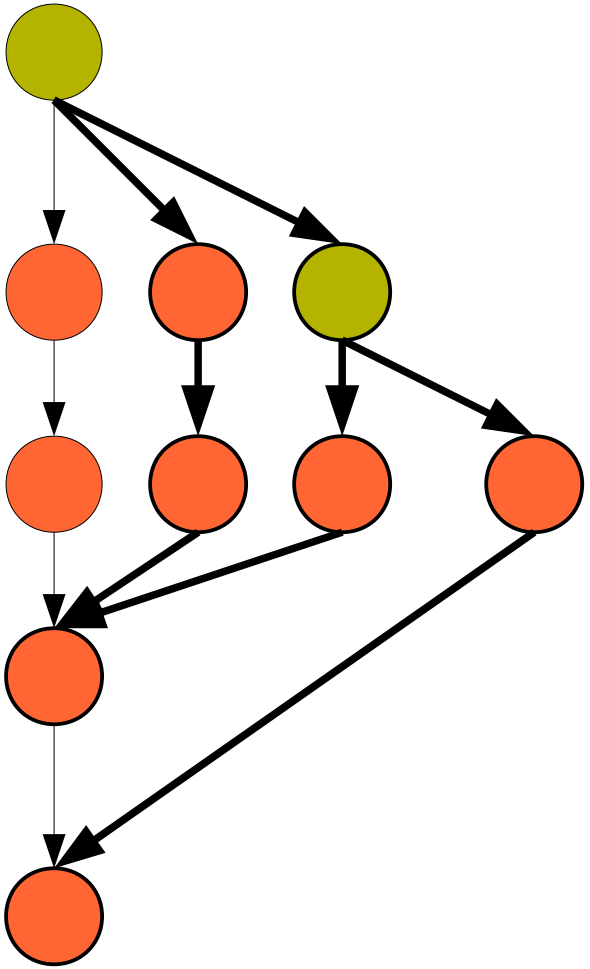
```
* text=auto !eol
```

Special Properties

- From Subversion to Git
  - svn:ignore; svn:mime-type and svn:eol-style are translated
- From Git to Subversion
  - .gitignore and relevant values set in .gitattributes are translated

Merge Tracking

Git merge commits:



In Git merges are tracked by:

- Natural history
- Merge commits parents
- Merge unit is commit

Merge Tracking

- Subversion tracks merges by
  - Natural history
  - Revisions ranges (including cherry-picks)

- Merge unit is file or a directory

```
/trunk@r100
   svn:mergeinfo=/branches/1.0.x:1-100,105

/trunk/file.txt@r100
   svn:mergeinfo=/branches/1.0.x/file.txt:102
```

Merge Tracking

- From Subversion to Git
  - Branch-level merges are translated to merge commits
- From Git to Subversion
  - Merge commits are represented with svn:mergeinfo

Not Yet Translated (in SubGit 1.0)

- Empty directories
- Arbitrary properties
- Revision properties
- Externals references

It Works!

```
$ svnadmin create repos
....
$ subgit install repos
$ git clone repos git-clone
....
$ svn co svn://host/repos wc
....
```

SubGit Target Audience

- Managers who listen to their Geeks
- Hosting providers
- Safe and smooth migration

- Java 1.5 or newer is needed
- Linux, OS X, Windows Server 2003

On SubGit Availabilty

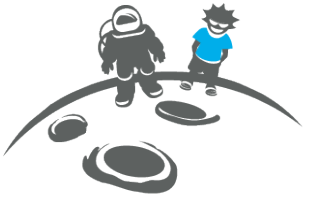
- SubGit 1.0 release: June 2012
- SubGit is free for Open Source and Start-Up Projects
- Closed-source projects will have to register SubGit (purchase a registration key)

- Dedicated technical support for SubGit

Thank you!

- SubGit on Web: http://subgit.com/
- SubGit on Twitter: @subgit
- Page on Google+ (SubGit)
- Blog: http://blog.subgit.com/
  - Coming soon: using Gerrit with Subversion

- Thank you! :)



tmate
software
dream driven development