



# Subversion Replication via Apache proxy



# Subversion Replication via Apache proxy

## Preface

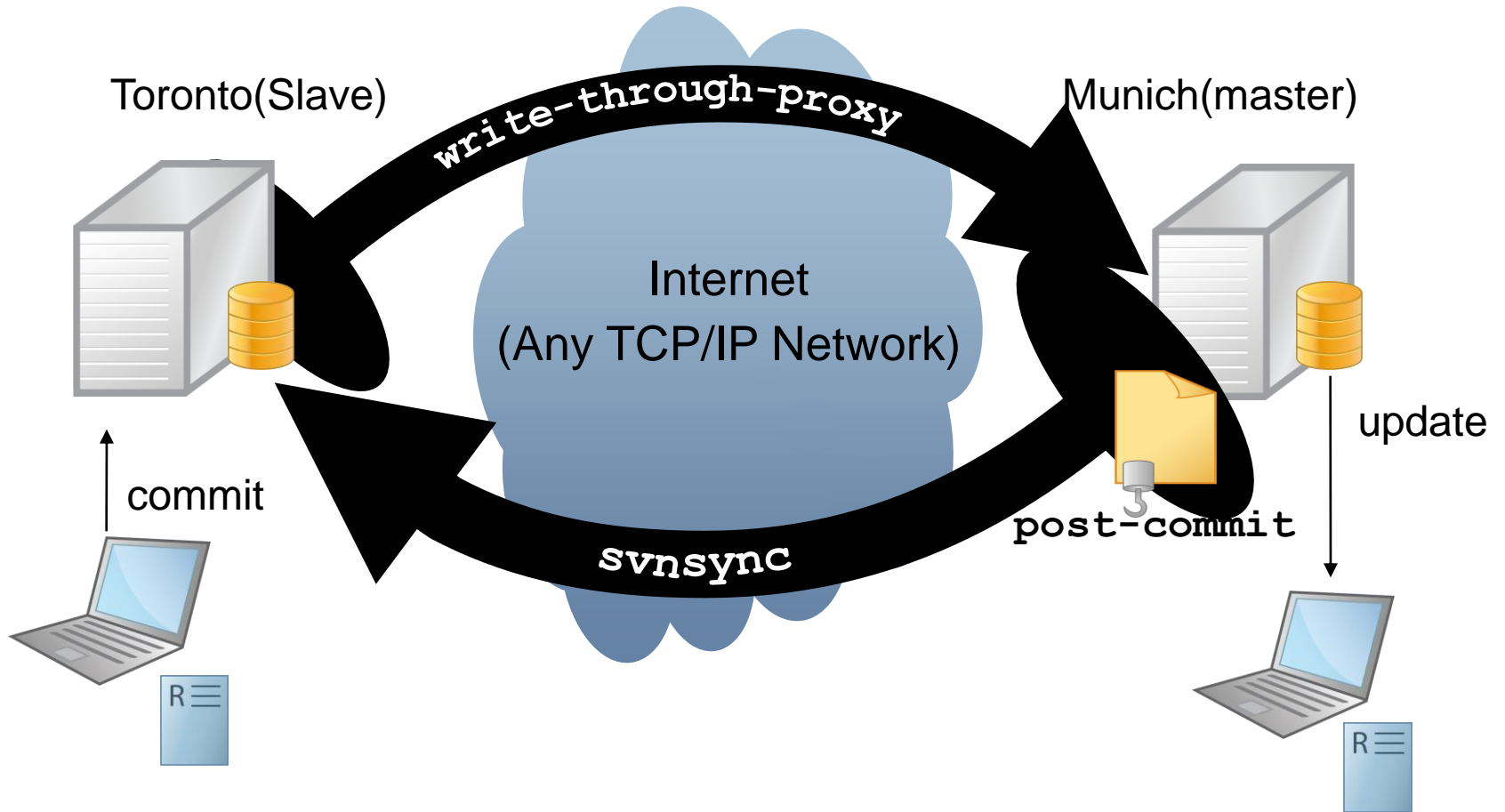
quite new feature

not subversions core functionality

master/slave, not real replication



# Subversion Replication via Apache proxy Architecture





# Subversion Replication via Apache proxy

## Read-only alternatives

Sometimes only actual read access is needed

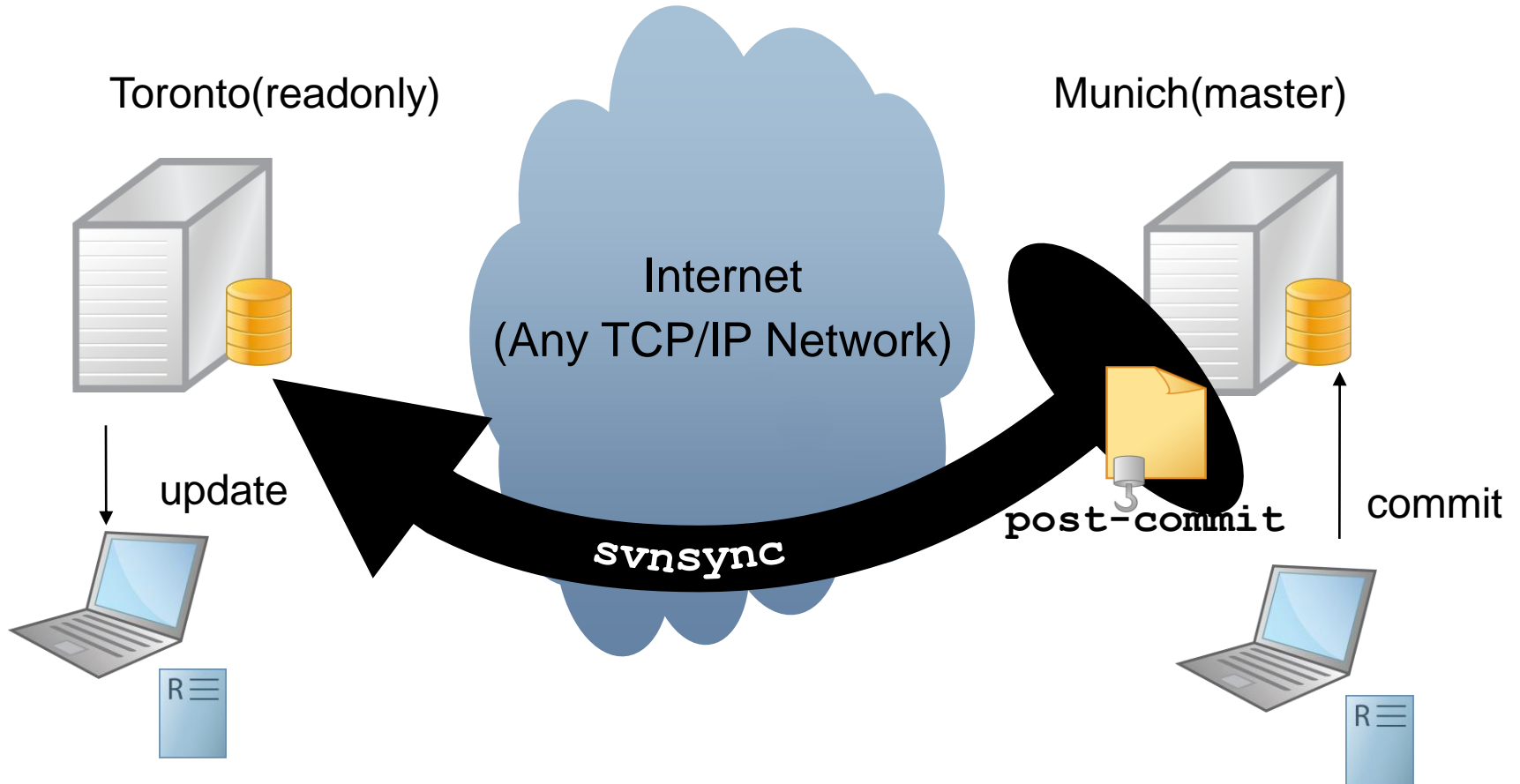
no demand for roundtrip

eg. 3<sup>rd</sup> party tools (apache repo, svn repo..)



# Subversion Replication via Apache proxy

## Read-only alternatives:architecture





# Subversion Replication via Apache proxy

## Read-only alternatives 1: svnsync

enable the `pre-revprop-change` hook in `<DEST_URL>`

```
svnsync init <DEST_URL> <SRC_URL>
```

will initialize repository `<DEST_URL>` by writing sync data into properties in REV 0

```
svnsync sync <DEST_URL>
```

will transfer all pending revisions to `<DEST_URL>`

**Note:** works also without physical access to `<DEST_URL>`



# Subversion Replication via Apache proxy

## Read-only alternatives 2: dump/load

svnsync limitations:

- Replication with a subset of directories will not work with svnsync
- additional work is not allowed in destination via svnsync

workaround:

dump the appropriate part of the source repository, loading it into destination

**LIMITATIONS:**

you need [file://](#) access to both repositories

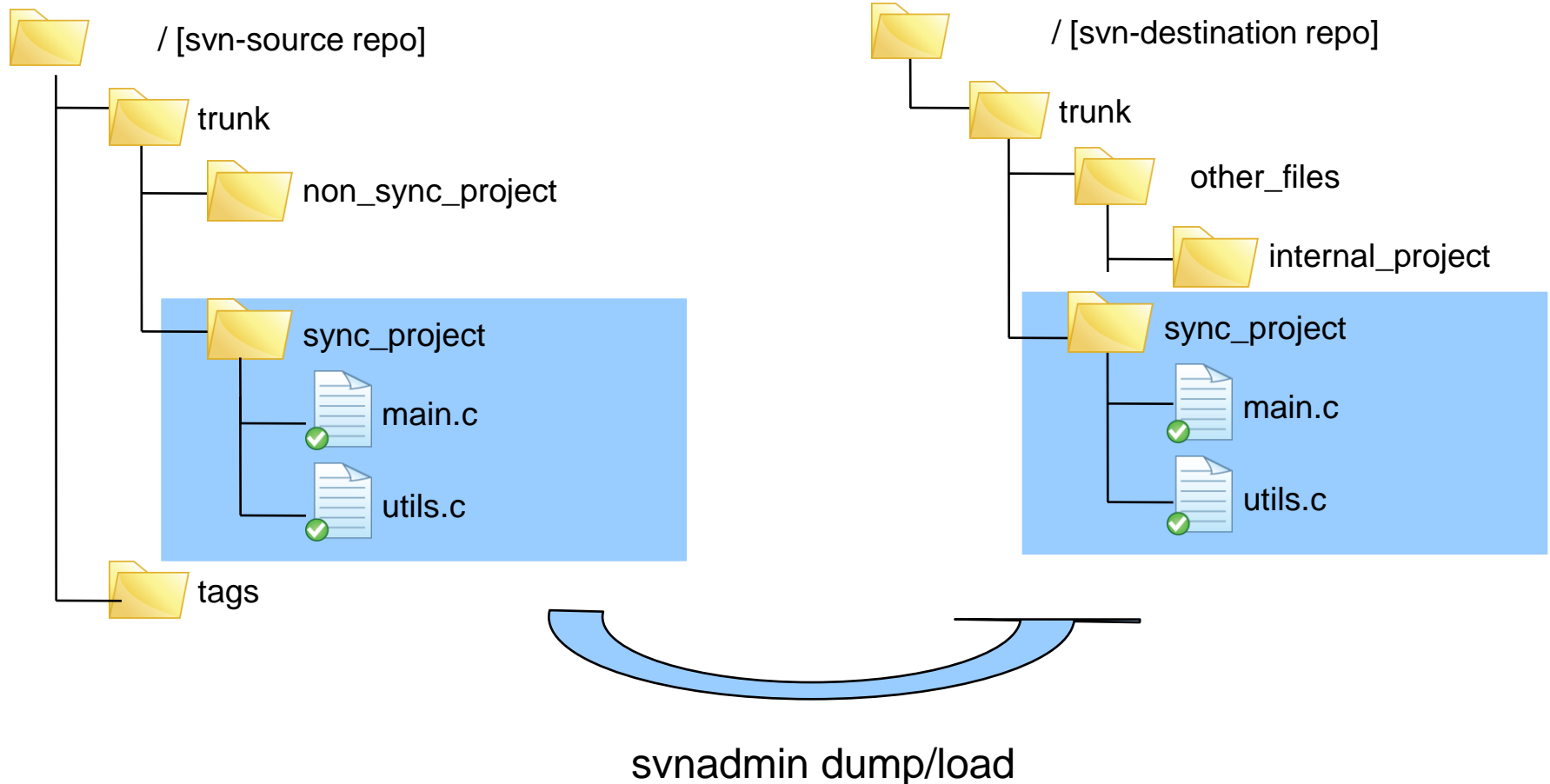
syncpath has to be writeprotected in destination, other paths can be writable

you cannot copy from outside of syncpath



# Subversion Replication via Apache proxy

## Read-only alternatives 2: dump/load





# Subversion Replication via Apache proxy

## Setting up Slave

httpd.conf

```
<Location /svn>  
  DAV svn  
  SVNParentPath <SVN_Parent_Path>  
  SVNMasterURI http://master.com/svn/  
  #AuthType Basic ... and so on..  
</Location>  
  
<Location /svn-sync>  
  DAV svn  
  SVNParentPath <SVN_Parent_Path>  
  Order deny,allow  
  Deny from all  
  Allow from master.com  
</Location>
```

http://slave.com/svn

http://slave.com/svn-sync

Toronto(Slave)





# Subversion Replication via Apache proxy

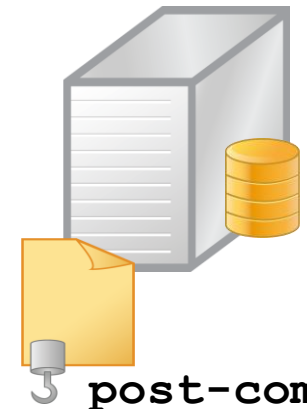
## Setting up Master

httpd.conf

```
<Location /svn>  
  DAV svn  
  SVNParentPath /opt/svn_master  
  # AuthType Basic .. and so on..  
</Location>
```

<http://master.com/svn>

Munich(master)



post-commit



# Subversion Replication via Apache proxy

## Setting up Master Hooks

**IMPORTANT: initialize slave-repo by:**

```
svnsync init http://slave.com/svn-sync http://master.com/svn
```

```
post-commit
```

```
#!/bin/bash  
svnsync sync http://slave.com/svn-sync/<reponame> &
```

Munich(master)



post-commit  
post-revprop-change

```
post-revprop-change
```

```
#!/bin/bash  
rev=${2}  
svnsync copy-revprops http://slave.com/svn-sync/<reponame> ${rev}&
```



# Subversion Replication via Apache proxy

## Problem: LOCKS

Description:

LOCKS have to be replicated to the client otherwise they either only show up on commit, or they will not be replicated



# Subversion Replication via Apache proxy

## Solution 1: LOCKS

Solution 1:

LOCKS have to be transferred to the master repository.  
use `post-lock-hook` for this



# Subversion Replication via Apache proxy

## Solution 2: LOCKS

Solution 2:

LOCKS have to be transferred to the master repository.  
and should use the same lock token

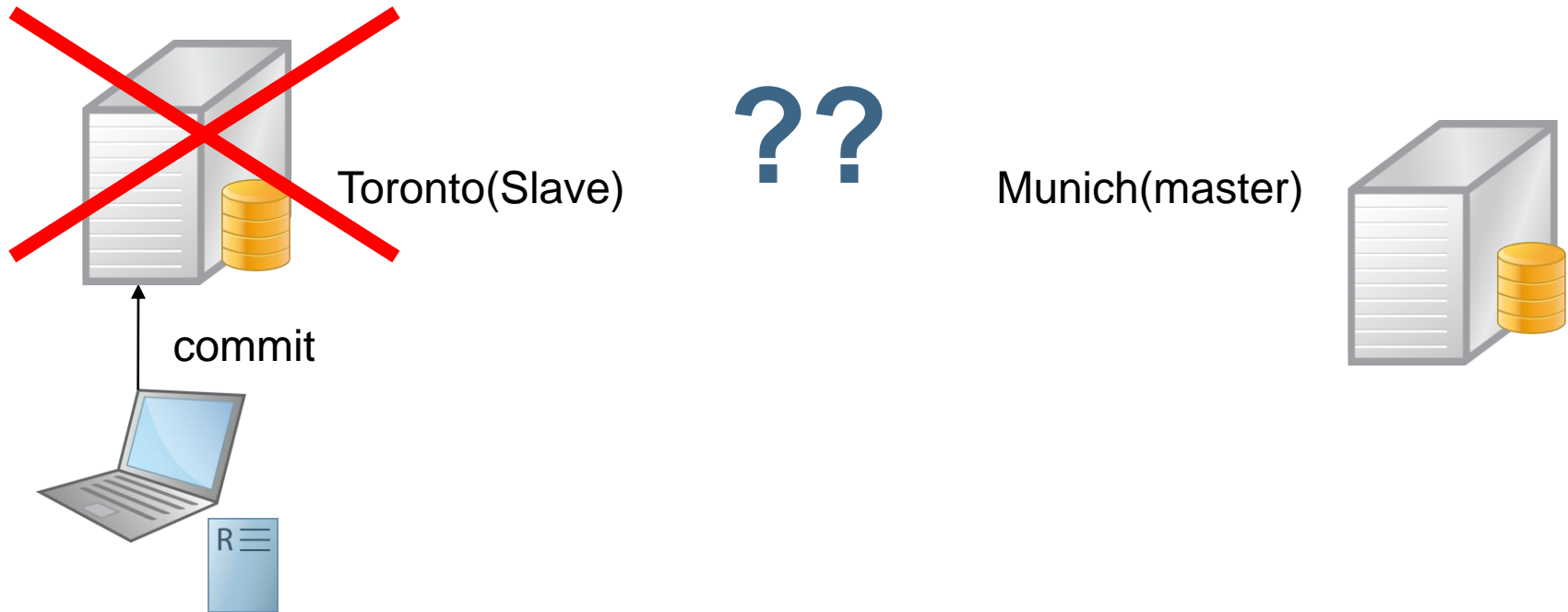
rsync the <reponame>/locks-directory



# Subversion Replication via Apache proxy

## Problem: Slave offline

If slave server will fail, work cannot continue, as working copys are pointing to slave repository



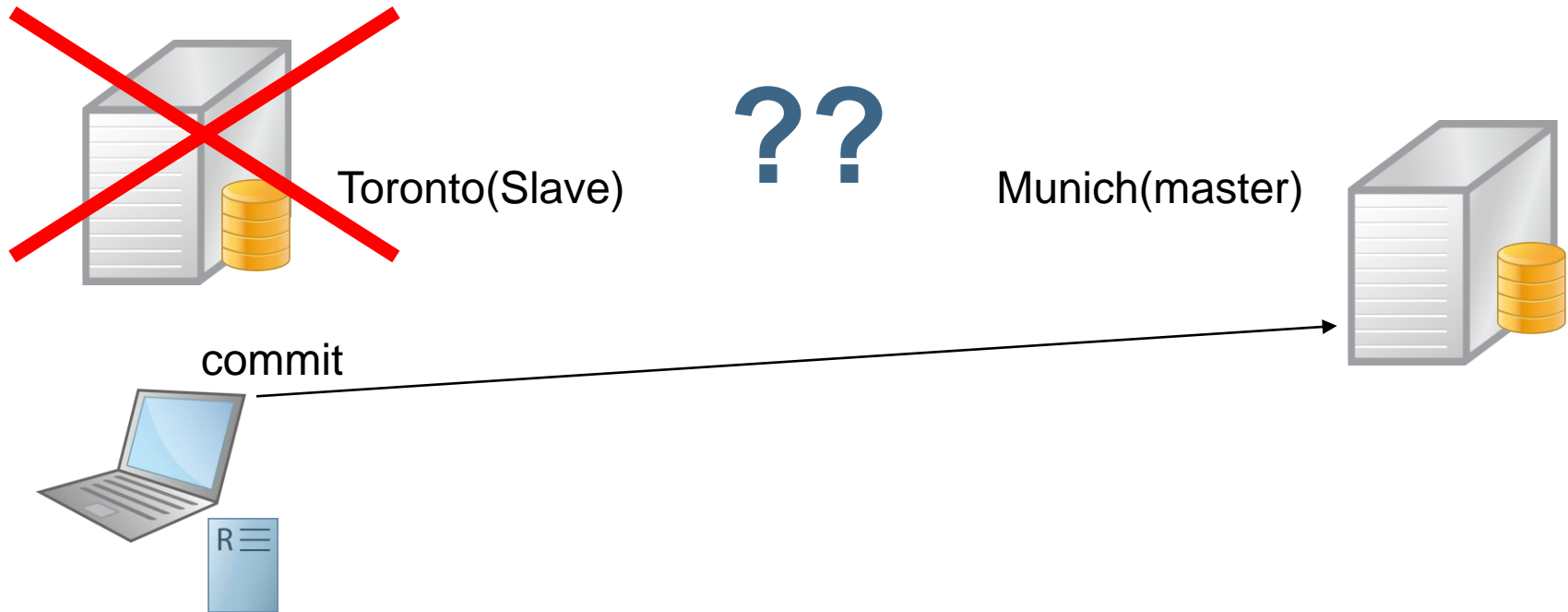


# Subversion Replication via Apache proxy

## Solution: Slave offline

```
svn switch -relocate master.com/svn/<reponame>
```

But this is only possible if master and slave share same uuid



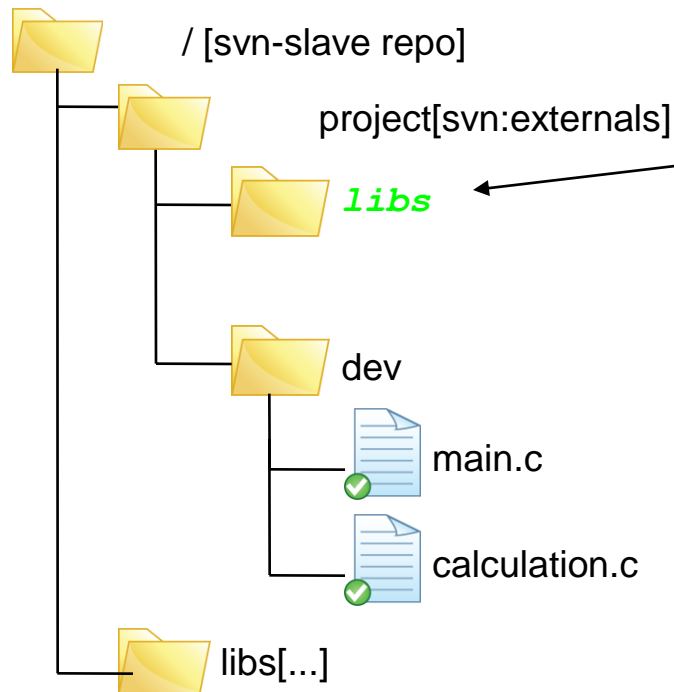


# Subversion Replication via Apache proxy

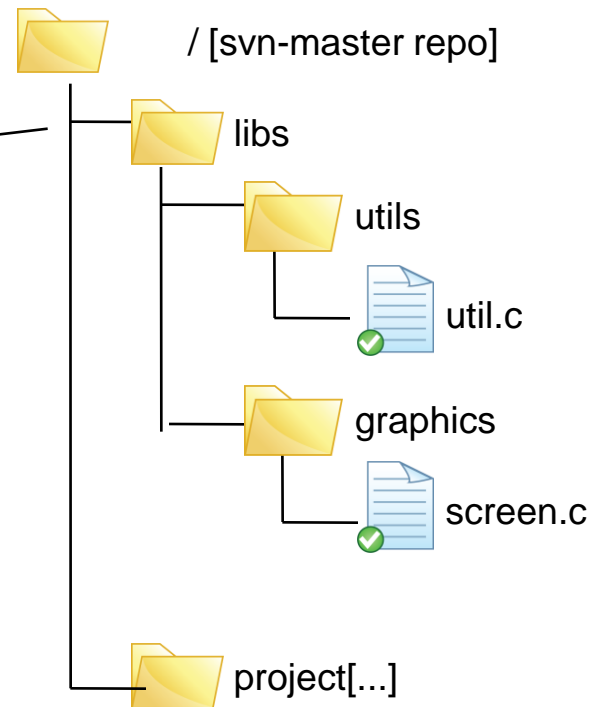
## Problem: externals

Pre-1.5 svn:externals will point to the master repository and in this case advantages of proxying are limited

Toronto(Slave)



Munich(master)



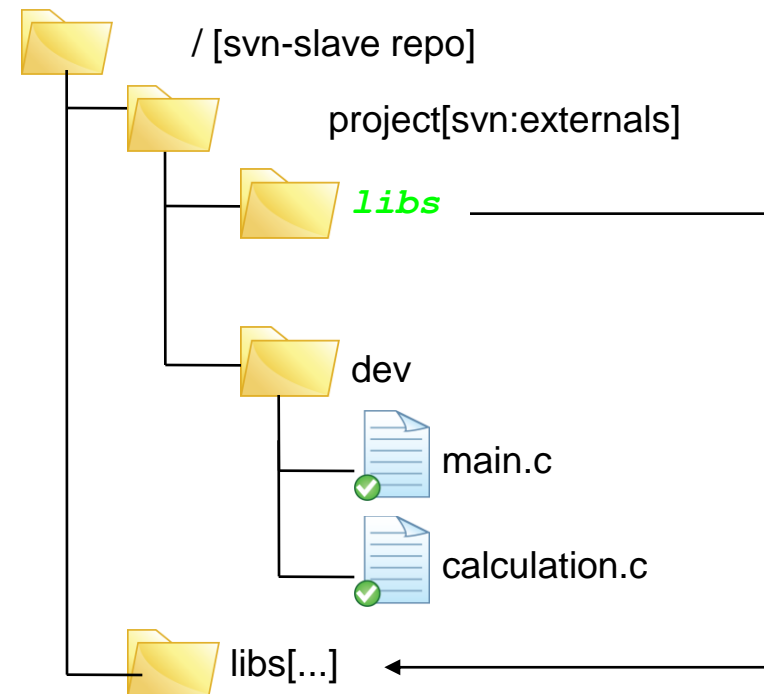


# Subversion Replication via Apache proxy

## Solution: externals

externals have to be rewritten in SVN1.5 relative format:

- ../ relative to directory of externals-property
- ^/ relative to root of repository
- // relative to URL scheme of the
- / relative to URL of the server





# Subversion Replication via Apache proxy

## Summary

- Subversion cannot replicate, but separates the write operations
- + performance gain
- complex setup
- + runs smoothly, if it is setted up
- difficult to track errors(another layer of indirection)



Any Questions?