



# Introduction to CVS

---

October 26th, 2004  
by Tobias Lasser



# ○ ● ● Topics

---

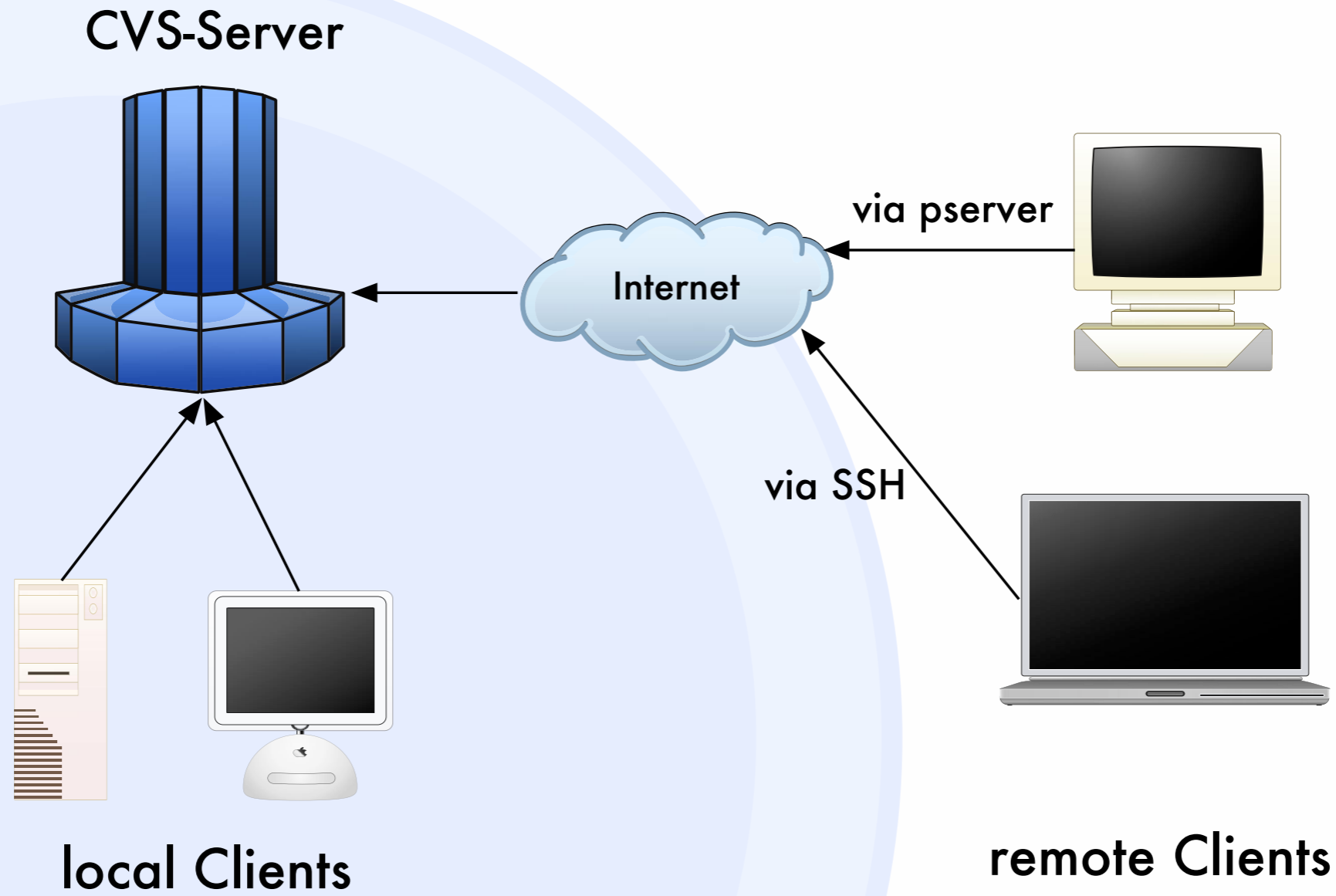
1. What is CVS?
2. Client/Server Model
3. Networked access methods
4. Example session
5. Outlook: Tags and Branches
6. Conclusion

# ○ ● ● What is CVS?

---

- **CVS = Concurrent Versions System**
- **an open-source network-transparent version control system**
- **records history of all changes to files**
- **manages simultaneous development by several people**


# ○ ● ● Client/Server Model



# ○ ● ● Network access

---

- on localhost
- via NFS/SMB-Mount: **dangerous!** (file-locking, permission issues)
- via pserver protocol (TCP/IP, Port 2401): telnet-like, good for anonymous access
- via SSH: encrypted TCP/IP via Port 22, needs account on server



# Example Session

---

```
$ export CVSROOT=:ext:lasser@atnavab5:/home/cvs/  
$ export CVS_RSH=ssh
```

```
$ cvs checkout ramp
```

```
$ cd ramp
```

```
$ cvs status Makefile
```

```
=====
```

```
File: Makefile      Status: Up-to-date
```

```
Version:           1.4      Tue Oct    26 11:43:02 2004
```

```
RCS Version:       1.4      /home/cvs/ramp/Makefile,v
```

```
$ vim Makefile
```

```
$ cvs update
```

```
M Makefile
```

```
...
```

```
$ cvs commit
```

```
...
```

# ○ ● ● Tags and Branches

- Tag: mark files with a symbolic revision:

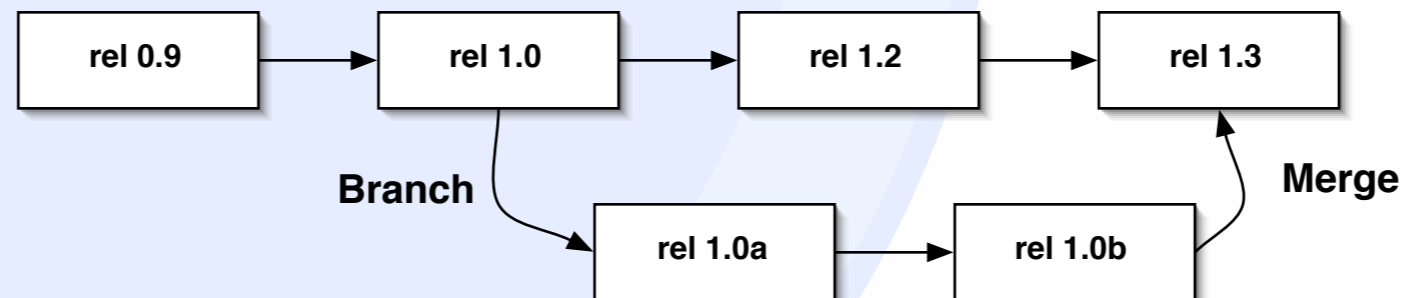
```
$ cvs tag rel-1-0
```

- checkout this revision explicitly:

```
$ cvs checkout -r rel-1-0 test
```

- create a branch:

```
$ cvs tag -b rel-1-0-patches
```



# ○ ● ● Conclusion

---

- CVS is the standard today
- plenty add-on software packages available, tutorials, examples and documentation
- More advanced system: Subversion
  - has most CVS features
  - directories and renames versioned
  - HTTP-Access via WebDAV



# ○ ● ● References

---

- **CVS:** *<http://www.cvshome.org>*
- **Subversion:** *<http://subversion.tigris.org>*