

Sistemas de Controlo de Versões

Laboratórios de Informática I

Marco Couto Cláudio Lourenço Nuno Macedo
Tiago Oliveira Hugo Pacheco José Proença

2017-2018 — Universidade do Minho

Trabalho em Equipa

O trabalho em equipa pode ser **difícil de gerir**, especialmente quando as equipas são grandes e os sistemas de **partilha de ficheiros** são rudimentares.

Sistemas tradicionalmente usados:

- suportes físicos (pen USB, discos externos);
- email;
- Dropbox, Google Drive, ...

Trabalho em Equipa

Estes meios de partilha são **pouco eficazes** em projetos de programação. São lentos, dificultam trabalho simultâneo e tendem a produzir imensas cópias.

Troca típica de emails:



1. projeto.zip
2. projeto2.zip
3. projeto-final.zip
4. projeto-FINAL.zip
5. projeto-final-entrega.zip
6. ...

E a Dropbox/Google Drive?

Estes sistemas reduzem o número de cópias, e permitem uma melhor organização do espaço. No entanto, não lidam bem com [retrocesso de versões](#):

“Dropbox keeps snapshots of all changes made to files in your Dropbox within the past 30 days [...]” — Dropbox Help Center

Nem com [resolução de conflitos](#):

Name	Date modified
 Sample File (Scott's conflicted copy 2009-10-15)	10/15/2009 4:30 PM
 Sample File	10/15/2009 4:30 PM

Sistemas de Controlo de Versões

Estes sistemas são especializados em promover o trabalho colaborativo. Alguns dos seus pontos fortes:

- histórico completo de revisões – o que mudou e quem fez as alterações;
- permitem reverter para versões anteriores;
- permitem resolução de conflitos – automática sempre que possível;
- mensagens descritivas do que mudou em cada versão;
- ramificação, estatísticas, ...

Sistemas de Controlo de Versões

Sistemas centralizados:

- CVS
- SVN

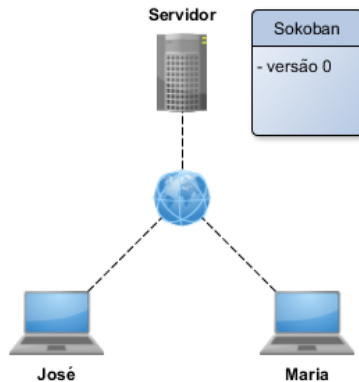
Sistemas distribuídos:

- Git
- Mercurial
- Bazaar

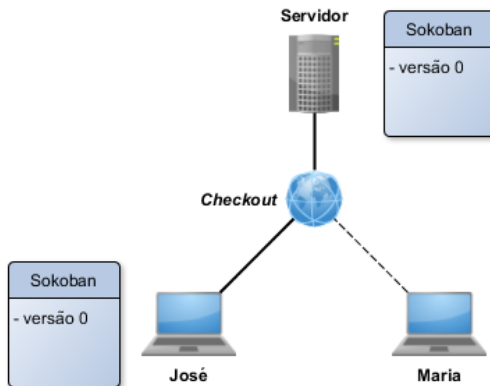
Utilizados para manter documentação, ficheiros de configuração e código fonte.

Não é recomendado submeter no repositório ficheiros executáveis.

Exemplo



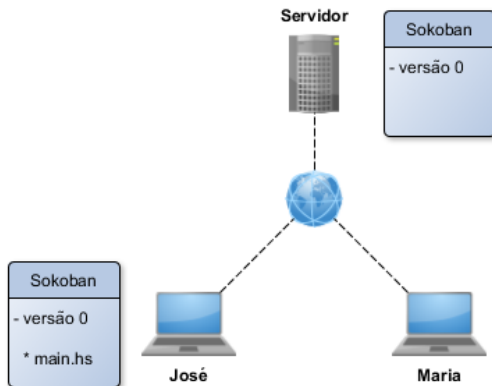
Exemplo



O José:

```
$ svn checkout svn://svn.alunos.di.uminho.pt/2017li1g999  
--username jose
```

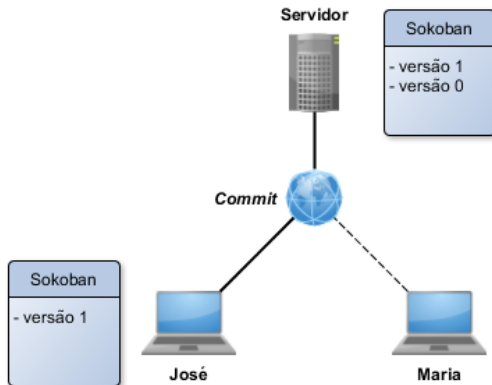

Exemplo



O José:

```
$ subl main.hs  
$ svn add main.hs
```

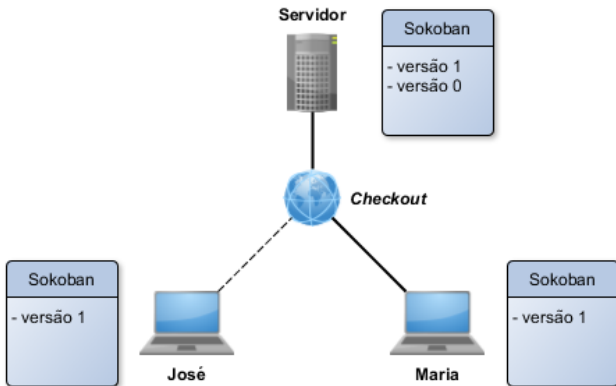
Exemplo



O José:

```
$ svn commit -m "josé: adicionada main.hs"
```

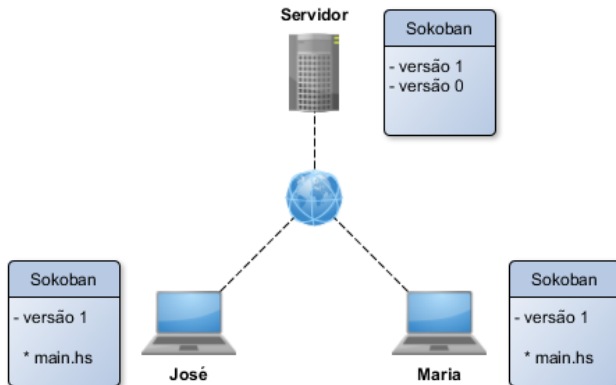
Exemplo



A Maria:

```
$ svn checkout svn://svn.alunos.di.uminho.pt/projecto  
--username maria
```

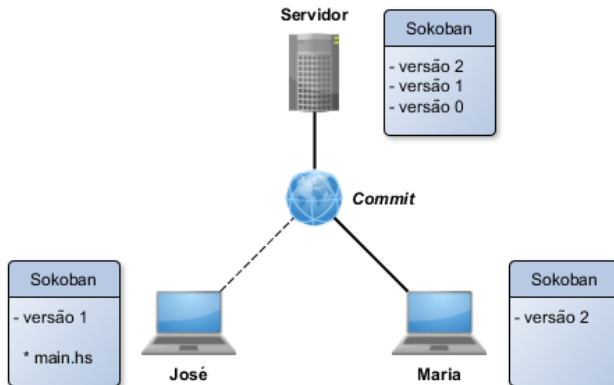
Exemplo



Em paralelo, o José e a Maria:

```
$ subl main.hs
```

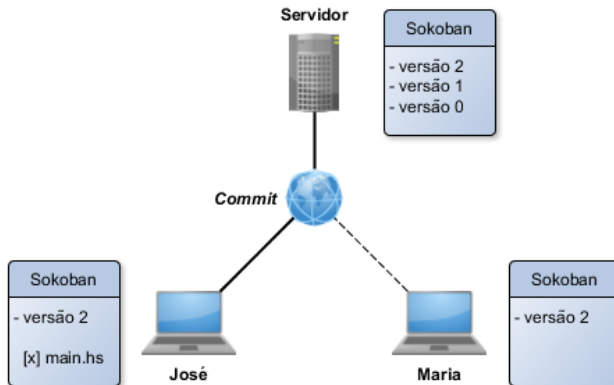
Exemplo



A Maria:

```
$ svn commit -m "maria: alterada main.hs"
```

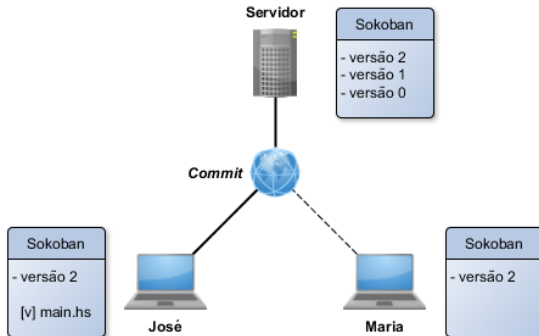
Exemplo



O José:

```
$ svn commit -m "josé: alterada main.hs"
```

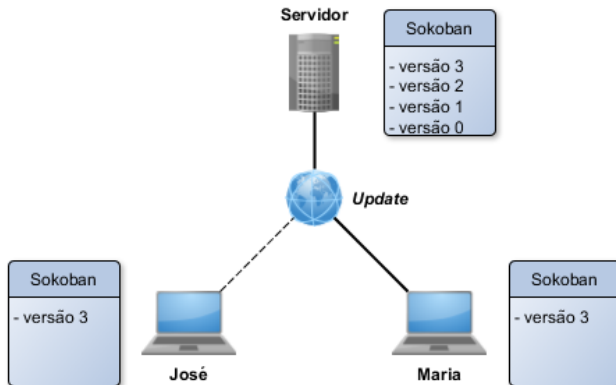
Exemplo



O José:

```
$ svn update  
$ subl main.hs  
$ svn commit -m "josé: resolvido conflito main.hs"
```

Exemplo



A Maria:

```
$ svn update
```


Dicas SVN

- Mostrar o log dos últimos 3 commits

```
svn log --limit 3
```

- Alterar o editor de texto padrão do SVN:

Tipicamente na pasta `/home/username/.subversion:`

- Existe um ficheiro `config`:
- Adicionar a seguinte instrução:

```
editor-cmd = vim
```

ou

```
editor-cmd = gedit
```

ou

```
editor-cmd = "subl"
```

- Verificar diferenças entre revisões:

```
svn diff -r{revisão inicial}:{revisão final}  
ficheiro.txt  
$ svn diff -r1:4 README.txt
```

- Ficheiro removido acidentalmente:

```
$ svn update
```

Dicas SVN

- Restaurar ficheiro:

```
$ svn revert ficheiro
```

- Retroceder ficheiro para revisão específica:

```
$ svn update -r revisao ficheiro
```

- Pedir ajuda:

```
$ svn help
```