
SUBVERSION 1.6 Preview Summit

**The long and winding road
That leads to 2.0
Will never disappear
I've seen that road before**

COLLABNET
Where Subversion™ Meets the Enterprise.

elego
Software Solutions GmbH

Who are we?

COLLABNET
Where Subversion® Meets the Enterprise.



Julian Foad

Stefan Sperling

What can be expected beyond 1.6?

COLLABNET
Where Subversion® Meets the Enterprise.



Next-generation working copy library “wc-ng”

New, custom http-based protocol “httpv2”

Tighter release cycles

<http://www.hyrumwright.org/papers/floss2009.pdf>

Incremental improvements of tree conflict handling.

Initial clean-room design by Erik Huelsmann

Requirements:

- developer sanity
- fast
- resilient to case-insensitive filesystems
- client-side true rename support
- recording of mtimes

Coding already started by Greg Stein in 2008

Centralised meta-data store, no .svn directories everywhere

Ongoing work in trunk for 1.7, Hyrum Wright is hacking, too

Subversion 1.6 detects

- tree conflicts that you need to know about
- some tree conflicts that you wish it would resolve automatically

If you merge two identical changes...

- Branch A: delete "file1"
- Branch B: delete "file1"

... it raises a tree conflict. Good or Bad?

Tree conflicts: identical changes

Let's imagine...

- While working on branch B, I apply a patch "P1"
- I merge the latest changes from branch A
- I get a tree conflict on "file1": delete/delete

because I didn't know

- you already applied the same patch "P1" on branch A

The conflict is unnecessary

Tree conflicts: identical changes

Now let's imagine...

- I did a lot of work on branch B
- I merge the latest changes from branch A
- I get a tree conflict on "file1": delete/delete

Why?

- I deleted "file1" because I incorporated its text into file "alpha"
- You deleted "file1" because you thought it was obsolete

So I do want to know about this conflict.

Tree conflicts: improvements

You can resolve it easily

- svn resolve

But, if there are many files, we want a way to say

“please be relaxed about this kind of conflict”

or “please be strict about this kind of conflict”

Subversion 1.6 is “strict”

- tools such as Subclipse can give you the choice

What does 2.0 really mean?

No backwards-compatibility *required* for:

- API
- client/server protocols
- working copy / repository formats

There will be hype.

Other than that, probably business as usual...

And after 1.7? (WARNING: speculation!)

COLLABNET
Where Subversion® Meets the Enterprise.



To be frank: There is no real roadmap!

It's best to take a look at a collection of remaining problems:

true renames

copy vs. branch/tag

UTF-8 issues

"True" Renames

Rename == copy + delete

- In branch A:

```
svn rename OLD NEW
```

```
D /A/OLD
```

```
A /A/NEW (from /A/OLD:5)†
```

This is OK...

- svn update

- finding the old name

What's Wrong with Renames?

Merging...

- In branch B:

```
svn merge ^/A
A   /B/NEW (from /A/OLD:5)↑
D   /B/OLD
```

What is a "True" Rename?

Copy and delete **the same item!**

```
svn rename OLD NEW
```

```
D /OLD
```

```
A /NEW (from ./OLD: (REV-1) )↑
```

To have “svn rename” recorded properly...

in the Working Copy

- maybe in 1.7 ?

in the Repository

- possible in 1.7/1.8 ?

and used in a Merge

- maybe in 1.9/2.0 ?

SUBVERSION 1.6 Preview Summit

**Thank you very much
for your attention!**

COLLABNET
Where Subversion™ Meets the Enterprise.

elego
Software Solutions GmbH