Споделяне Сигнал за злоупотреба Следващ блог» Създаване на блог Вхс

FRANK SCHOPHUIZEN'S BLOG

RANDOM THOUGHTS ON THE PUBLIC PART OF MY PRIVATE LIFE, MY PROFESSIONAL INTERESTS AND OTHER THINGS THAT COME TO MY MIND.

FRIDAY, JANUARY 30, 2009

Subversion is not an SCM tool

Subversion (SVN) is not a Software Configuration Management (SCM) tool. It is a Version Control tool. Basically, Subversion is nothing more than a versioned directory tree with a few version control features.

Let me try to explain why I think that Subversion is not an SCM tool.

No real promotion model
First think that is missing from SVN, and many other version control tools, is the ability to support only the most basic promotion model: promotion of changes from your private work area to the branch in the repository, and vice versa.
A real promotion model would allow you to promote between various areas as defined for your organization. For example:

- Private work area
- Private system build area
- Team integration area (also called the "stream")
- Build area (often running a continuous build)
- Test area
- Release area
- Staging area

In a real SCM system, you can define these kind of area and the rules (criteria) to promote between them. In SVN you cannot. What you can do is use branches - which are actually just a copy of a directory - and create a script or application to automate it (or do it manually).

No real baselines
A baseline is a reproducible configuration that is immutable (you cannot change it anymore). SVN does offer "tagging" but in fact it is creating a branch - which is a directory copy. The tag/branch is reproducible, but it is not immutable! You can still check out and check in on it, which is the normal way of making changes. Again, there may be workarounds, e.g. to remove write access for all users on the tag or by writing a pre-action hook that prevents check out.

No projects, teams or streams
Many software development organizations run more than one project at the same time, and/or have projects consisting of teams that work simultaneously. Projects will make baselines and teams will

ABOUT ME

FRANK SCHOPHUIZEN

VIEW MY COMPLETE PROFILE

FOLLOWERS

Follow
with Google Friend Connect

Followers (4)

Already a member? Sign in

BLOG ARCHIVE

Blog Archive ▾

BLOGROLL

Personal blogs
Brad Appleton's ACME Blog
Community and social computing
Frank Schophuizen's blog
Project Management for Slackers
Frank's space
Tamir Gefen on ALM
Kellypuffs
Accidental Simplicity

Leadership & Excellence
Leading Blog: A Leadership Blog @ LeadershipNow
Entrepreneurial Thought Leaders
epistemologic

work simultaneously. Projects will make baselines and teams will deliver to other teams. In SVN you cannot define integration areas (aka streams) for different projects or teams that are isolated from each other.
Again, we could use branches as a workaround; project-branches result in project-tags and team-branches are integrated (merged) with other team-branches and ultimately with the project-branch.

No change sets
Here is something you may disagree with. In SVN you can submit your changes in the work area to the repository. SVN does not create a change set for it, but it creates a new revision of the (whole!) respository. Of course, the new revision will be equal to the previous revision except for the changes you submit, but still SVN does not make a change set out of it. Nevertheless, you might argue that the submitted changes form a unity, which could be called a "change set". The problem is however, that you submit *all* changes from your work area as a single change set, while in fact you may have been solving a problem report and implemented a new feature. If you want to identify the solution for the problem report (i.e. the change set corresponding to it) you are unable to do that.
A real SCM system is able to identify change sets related to units that are identified as managed items (e.g. work items, problem reports, change requests, requirements). Using a diff between revisions might be solution, but if branches are used to separate different areas, projects, teams and streams, the sequence of revisions in SVN or not so obvious. And again, adding a script to automate the selection of revisions to compare it adding an SCM feature that SVN is missing.

No roles
An SCM system is supposed to support an SCM process. One of the characteristics of an SCM process is that people can have different roles. Depending on the role, the process is different and with it the SCM tool behaviour is different. For example, a project leader is authorized to assign roles and the scope of control of a particular team or individual, but only for the project he is responsible for. These roles assigned privileges to the members of the teams.
SVN is able to assign read/write access to users at path level, but offers no means for roles like project manager, integrator, librarian, build master, developer, etcetera. Of course, you can implement this by hooks and scripts, but again you are extending SVN with SCM features that SVN lacks.

SVN does offer branching
Branching means that from a certain configuration (or revision in SVN) you create two - or more - parallel configurations. Typically, branching is intended to update a configuration from the past that cannot be implement in the current configuration (aka trunk, main, latest), for example a patch of a baseline. This is something that may be considered an SCM feature, which is supported by SVN.

Finally, SVN offers branching as one of the few SCM features. But reading back, I notice that many lacking SCM features can be resolved by applying branches for different purposes. If you are going

Personal improvement
insights you can use
The Chief Happiness Officer
You Already Know This Stuff
Creating Passionate Users
hersenspinsels en buikverzinsels

Other blogs
The Google Weblog

Software development
developer.* - The Independent Magazine for Software Professional
Managing Product Development
XProgramming - An Agile Software Development Resource

IBM Rational
Rational - Rational ClearCase
IBM developerWorks podcasts
developerWorks : Popular Threads - Rational ClearCase
Top Rational "Brand" Content
Telelogic Synergy
IBM Redbooks | Just Published
Rational - Rational ClearCase Change Management Solution
Notes from Rational Support
Dan Toczala's Blog
Uploads from @MauricioSWG
Open Services wiki

Project management
Agile Management Blog
Project Management Source

Photography
[daily dose of imagery]
Canon EOS 40D Users
Canon EOS 7D / 50D - 10D Forum: Digital Photography Review (dpre
Canon Rumors
Canon XSi | 450D Guide & FAQ | 50D Guide | Di
Chase Jarvis Blog
Digital Photography School
Digital Photography School - Photography Forums - How I took It
DSLR Wereld
ePHOTOzine
Fotografie.nl - Kennisbank
George Weir Photography
The blog of internationally acclaimed

to use the same technical solution to solve conceptually different SCM features, it will be quite difficult to distinguish between them: is the directory a promotion area, a project, a team, a stream, a baseline, a product, a component or module, just a parent directory to cluster projects, teams, streams, baselines, products, components, modules or just a leaf directory to cluster a number of files?
An SCM tool is supposed to help you implement an SCM process and a structure in the database. SVN only offers a versioned directory storage system. It's not an SCM tool.

POSTED BY FRANK AT 20:07

LABELS: CONFIGURATION MANAGEMENT, TOOLS

3 COMMENTS:

AbbotOfUnreason said...

I think it's unfair to say that Svn has no baseline ability, unless you are also willing to say that ClearCase is unsafe because it lets any user remove any element from the VOB. It provides a mechanism for immutable baselines in the same way that ClearCase provides a mechanism for protecting your elements.

I suppose that by your definition base ClearCase would not qualify as an CM tool?

As for changesets, as you said, I disagree with your assessment. In addition, Subversion's use of changelists is an interesting convenience feature for users to organize the status of their workspace and for use when committing against a defect/work item/whatever.

One thing I'm impressed by is that you didn't point to the update-modify-merge-commit model as a hindrance to SCM.

FEBRUARY 3, 2009 11:26 PM

tg said...

It's exactly the same words that I use when people ask me about SVN. I tell them it's a good version control system, but no more.

Anyway, what I miss in SVN is a process (~UCM) which includes all the assets and objects you have mentioned.

Recently, on V1.5, a change-sets have been added to SVN, but I still think it's a version control system.

Tamir
GoMidiets

wedding photographer and w
Photography Tips, Digital
Photography 101 at Cheapshooter
pildiblog by Remo Savisaar
Neil vN - tangents
pro photo life
Serious Amateur Photography
Strobist
Talkington Photo Video
The World Through My Eyes: Featured Photographer
PhotoCamel - Your Friendly Photography Forum
Alive for 365
Dutchphotozone
EOSDigitaal.nl
Fotoforum nieuws
...field of depth...
aperryProductions - Alex Perry - Photographer
SLRgear.com What's New
Reviews (latest): Digital Photography Review (dpreview.com)
Ed Verosky: Photography Blog
f stoppers
Powered by Bloglines

OTHER LINKS

Feedburner
Topic Embedded Systems

DEL.ICIO.US BOOKMARKS

MY DELICIOUS BOOKMARKS

The World Through My Eyes: Featured Photographer
Flickr: Discussing Hyperfocal Chart in Canon EOS 40D Users
CMMI for Development (CMMI-DEV) v1.2
Rosco US : Filters : Cinegel
Lensplay - Find A Lens
White balance and gelling flashes - technique
An Introduction to Photographing the Northern/Southern Lights
2008_11_15_0374 on Flickr - Photo Sharing!

Comitajets

MARCH 22, 2009 11:55 AM

Frank said...

AbbotOfUnreason,

Thank you for the feedback!

I agree that base ClearCase is not an CM tool (for other reasons than you quote). UCM comes closer but still is not a full CM tool (e.g. limited control of composite configurations).

The matter with changesets, the SVN mechanism may be convenient but not sufficient for the concept of change sets.

The change model seems irrelevant as a (dis)qualification for CM tool, to me at least.

Many VC tools can be customized to come closer to a CM tool.

APRIL 4, 2009 11:05 PM

Post a Comment

LINKS TO THIS POST

Create a Link

Newer Post Home Older Post

Subscribe to: Post Comments (Atom)

SEND A FILE TO ME

The 21 Most Popular Photography Posts on DPS in 2009

Karla+Luis=MegaPost...111 Images Warning!!

50mm f/1.2L Depth of Field on a 1.6x Crop Camera on Flickr - Photo Sharing!

Troubleshooting Windows XP, Tweaks and Fixes for Windows XP

How to consume the Rational Team Concert change management services - Library: Articles - Jazz Community Site

Distributed Builds with Ant and Cruisecontrol

Hottest 'cruisecontrol' Questions - Stack Overflow

Hottest 'cruisecontrol.net' Questions - Stack Overflow

flash and tungsten lighting ... " planet neil – tangents

Flash Photography with Canon EOS Cameras - Part I.

How to shoot a wedding - Lesson 1

CameraTools - Uw zicht op beeld

The Book .. it's here! « planet neil – tangents

Project Management Software for Agile Success | Rally Software Development

The Telelogic products have undergone a name change

Digitale foto forum • Forumindex

SendThisFile℠

Enter your email address

Enter a message here

File 1 Choose File No file chosen

File 2 Choose File No file chosen

File 3 Choose File No file chosen

File 4 Choose File No file chosen

File 5 Choose File No file chosen

SendThisFile Cancel

File:
Bytes: 0 (0%)
Elapsed: 00:00:00

Powered by: SendThisFile

BorderMaker - home

LCD monitor test images

[Tutorial] Hoe maak ik een huid mooi strak

First Look: IBM Rational Quality Manager « The Mac Daddy

Brad Appleton's ACME Blog: Embracing Change - quotable quotes on change and uncertainty

Composite-baselines and their components ...

change background of photo

15 Creative Uses Of Overexposure

Software Process Architecture Views and Quality Attributes

What is Agility? Part 2 - Software Agility

The Agility Cycle - Part 1

The Agility Cycle - Part 2

The Agility Cycle - Part 3

Sensing and Sense-making in the Agility Cycle

5S Qualities of Well Designed, Well-Factored Code

Brad Appleton's ACME Blog: Value Proposition for Agility

I am fschop on Delicious

Add me to your network

Frank Schophuizen works at TOPIC Software Services. Everything here, though, is his personal opinion and is not read or approved before it is posted. No warranties or other guarantees will be offered as to the quality of the opinions or anything else offered here.