

SCDB Status

Michel Jouvin

LAL/Orsay

jouvin@lal.in2p3.fr

Quattor Workshop, Dublin 2007



- SCDB
 - What is it ?
 - Changes since October 06
 - Coming soon..
 - Still on todo list
 - Comparison of profile versions
- HTTP RPM repositories (HTTPPrep ?)
 - What is it ?
 - Possible improvements



- SCDB is just an alternative implementation for CDB
 - Doesn't require any modification in other parts of Quattro
 - Client side, All or SWrep
 - Uses the same PAN templates to describe configuration
 - Can be used independently of QWG templates and/or new PAN compiler
- Design goals and usage paradigm different from CDB
 - Rely as much as possible on standard components/products
 - Allow greater flexibility using Subversion instead of CVS
 - Logical structure of templates reflecting physical layout
 - Support complex site structures with several subsites
 - No locking at any point
 - Offload the Quattro server from compile work by allowing better use of administrators local desktops
 - A possibility, not a requirement
 - Allow every platform to be used for SCDB (templates) administration

- # Changes since October 00
- No major changes
 - Focus on documentation and packaging
 - Documentation, including installation instructions, pretty complete
 - <https://trac.lal.in2p3.fr/Doc/SCDB>
 - Moved off Quattor installation guide : just a link to wiki site
 - LCG QWG repository contains a central point for getting SCDB tools, including externally maintained ones
 - Independent of QWG templates releases
 - 1 script allowing easy creation of a vanilla SCDB with a specific release of QWG templates (gLite 3 templates by default)



- integration of new PAN compiler as the default compiler
 - Ability to compile with old compiler will remain available through specific ant targets (suffix '.c' appended)
- Minor improvements
 - Rebuild RPM repository templates only if needed to avoid unnecessary full recompile
 - JavaSVN upgrade to SVNKit (SVN 1.4 based)
 - May require some attention as this requires SVN client upgrade to 1.4+ if used in the same workspace as SCDB tools
 - A minor problem with SVNKit password cache delayed the release
 - New 'tags' branch structure organized by year/month/day
 - Easier cleanup
 - Avoid SVN side effects with 5000+ of tags in the same branch experienced at GRIF



- CGI in replacement of SSH for triggering deployment
 - Avoid storing SSH keys without password for connection between SVN server and Quattor server
 - Direct execution of deployment script from SVN server if the same machine as Quattor server (no CGI, nor SSH)
- // compile of several clusters when several subsites
 - Distribute the load on several servers
 - On each server, compile only the profiles used at the site
 - Not useful when only 1 site with 1 server : panc already make an efficient use of all cores available
- Deployment of a selected cluster (after full compile)
 - An old request...
 - Many possible drawbacks in term of ability to guarantee configuration consistency
 - Main goal is speed up : improvement in panc, reduce need for full compile and // compile of clusters may be enough
 - What is max for deployment time ? Is 3 minutes acceptable as a m

- Starting with PANC v5, compiled profiles (XML) can be text compared
 - No arbitrary difference
 - False differences (formatting) if comparing a profile produced by panc v6 and panc v7
- Script available to easily compare all or some profiles between 2 configuration versions
 - `src/utils/profiles/compare_xml` : list all the profiles having differences
 - `-d` allows to check also dependency files produced by PAN
 - `-p` allows to check only one profile
 - `-v` allows to display the differences (unified diff format)
- Require result of previous (reference) compilation to be saved in `build.saved` (`cp -R build build.saved`)
 - Always possible to `'svn update -r n'` to recompile reference revision
- May probably be used outside SCDB

- Not really part of SCDB
 - Management tool integrated with SCDB management tool
 - `ant update.rep.templates`
- Design : match RPM repositories to document directories on the Web server
 - Updating RPMs require remote access to the file system where are the repositories
 - Filesystem ACLs can be used to control access
 - Update of RPM repository templates doesn't require access to the file system (done through http)
- 1 template associated with each repository
 - Special comment at head of the template associating it with the Web server directory
- No specific code to maintain, apart ant task



- A todo list rather than a planning...
 - Idea is to keep Quattor specific code very small
 - Nothing really done since last meeting
- Initial loading of repositories and synchronization
 - Add to standard templates a well known source
 - Add a task to do the "rsync"
- YUM-like features for automatic incremental update
 - A basic script, rpmUpdate.pl, can be used as a basis
 - A rock solid tool would probably require to parse metadata from another package manager like YUP or AP
 - Is it possible ? Who has the knowledge ?
 - Would prefer to avoid creating and maintaining our own metadata
- Use of squid to provide RPM repository caches
 - For increase perfs and load distribution in distributed sites

- Documentation and how to get it :
 - <http://trac.lal.in2p3.fr/LCGQWG>
 - Include a script to download a complete vanilla SCDB with OS, MW and other standard templates
 - Includes working examples that can be compiled with ant
- No documentation on client installation... because there is nothing to install !!!
 - All the SCDB specific components are inside SCDB itself.
 - Even easier with new Java based compiler, distributed with SCDB (no specific component to install)
 - Use of Eclipse for SCDB administration now documented on LCGQWG site
 - Removed from Quattor installation guide

