



Bruno packaging

SuperB XI general meeting - Frascati
R. Stroili - Università & INFN Padova

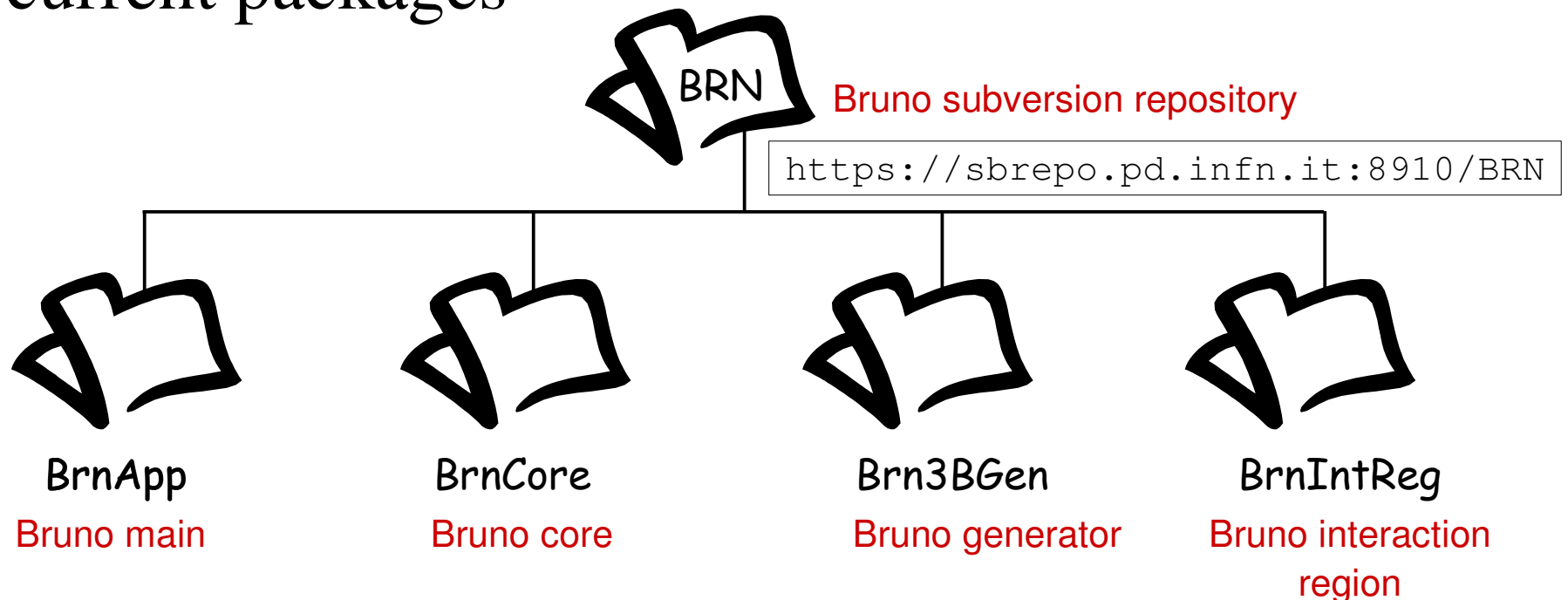


Outline

- repository(ies) structure
- tags
- software release tools
- package dependencies
- releases
- HOWTO
- tools
- trac

Repository structure

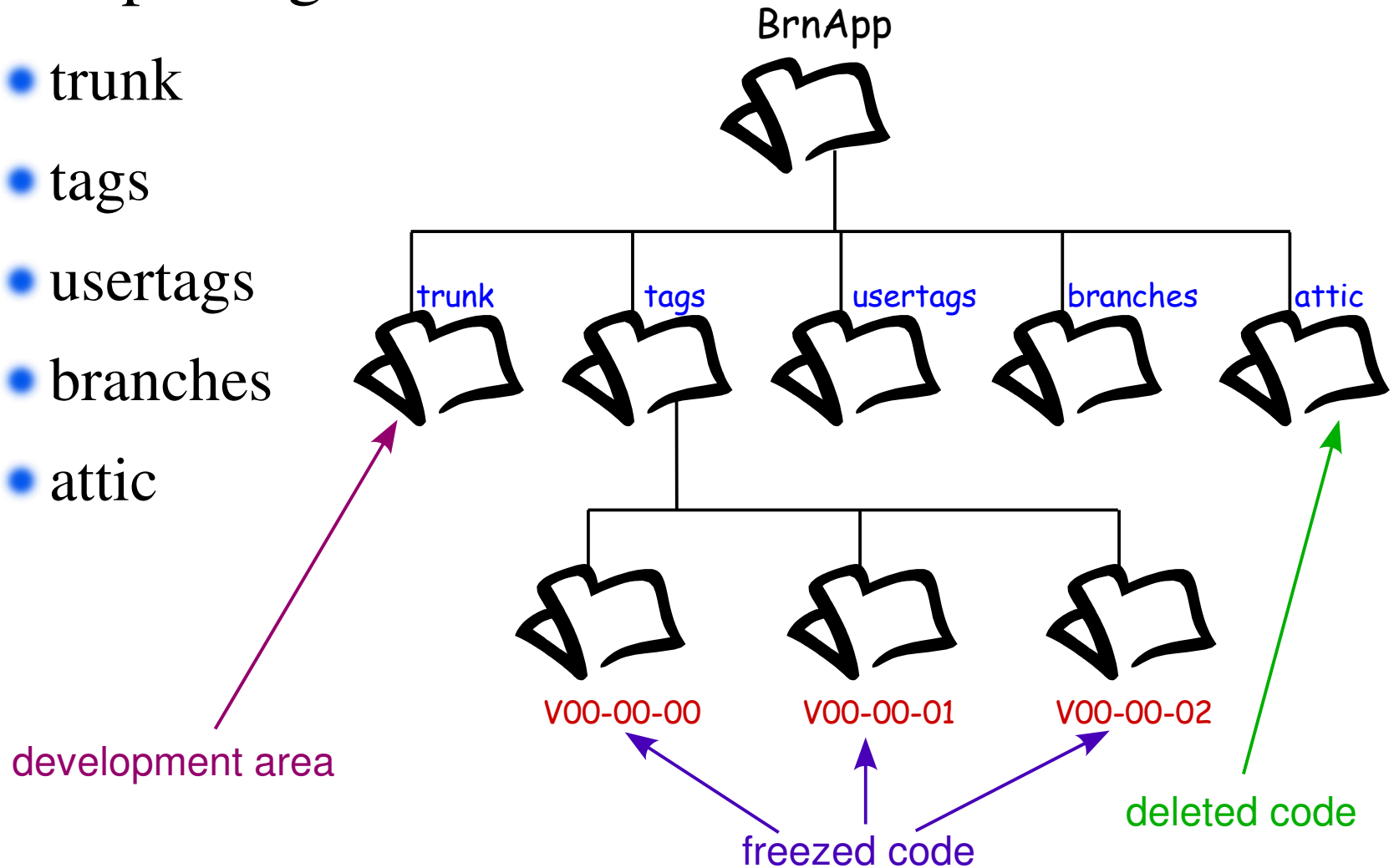
- the code is split in several packages (directories) within one or more subversion repositories
- it shares the same software tools used for FastSim
- current packages



Repository structure

- each package has its own structure

- trunk
- tags
- usertags
- branches
- attic



Tags

- tagged packages are meant to be non modifiable code
 - code that is working at some level
 - the subversion repository forces this policy through some control when a commit is made
 - it shouldn't be possible to commit code changes to a tagged version

Software Release Tools

- the software structure is similar to the FastSim one
 - hence to the BaBar one
- it uses SoftRelTools (SRT) to manage make dependencies
- external software dependencies are managed by SRT
- it requires PackageList to handle package dependencies

Package dependencies

- package dependencies are kept in file `link_all_reco.mk` in package `PackageList`

```
# Full simulation package
#
PACKAGELIST += BrnCore
ifneq ($(LINK_BrnCore),)
    LINKLISTDEPENDS += [LINK_BrnCore, $(LINK_BrnCore)]
    override LOADLIBES += -lBrnCore
    -include BrnCore/link_BrnCore.mk
endif

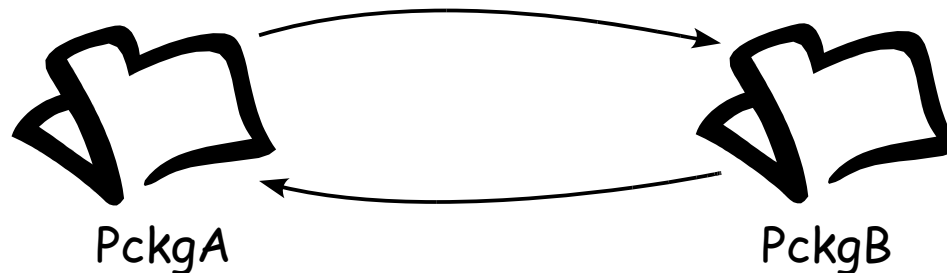
PACKAGELIST += Brn3BGen
ifneq ($(LINK_Brn3BGen),)
    LINKLISTDEPENDS += [LINK_Brn3BGen, $(LINK_Brn3BGen)]
    override LOADLIBES += -lBrn3BGen
    -include Brn3BGen/link_Brn3BGen.mk
endif
```

Package dependencies

- each package defines what packages it depends on in a file called `link_PACKAGE.mk`

```
# Please see SoftRelTools/HOWTO-dependency for documentation
# $Id$
override LINK_Brn3BGen      += BrnCore
override LINK_CLHEP         += BrnCore
override LINK_geant4        += BrnCore
override LINK_ROOT          += BrnCore
```

- avoid circular dependencies



Release

- a release is defined by the set of all tagged packages that make a well defined version of the application
- for the full simulation there's a bootstrap release (V0.0.0)

Releases

- releases are defined in the Releases svn repository

`https://sbrepo.pd.infn.it:8910/Releases`

SuperB

Preferences | Help/Guide | About Trac | logged in as stroili | Logout
Last Change | Revision Log

root

View revision:

Name ^	Size	Rev	Age	Last Change
FastSim		179	6 days	brownd: Update tags for production
V0.0.0		63	15 months	brownd: update SvnTools?
V0.0.1		72	11 months	brownd: Update PacIfr?
V0.0.2		81	9 months	brownd: Update PacEnv? tag
V0.0.3		89	7 months	brownd: Update PacDetector? tag
V0.0.4		86	8 months	brownd: Clone new release for trunk development before freezing V0.0.3
V0.0.9		120	4 months	brownd: Patch PackageList?
V0.1.0		140	4 months	brownd: Add KalTests?
V0.1.1		160	3 weeks	brownd: Update tags
V0.1.2		179	6 days	brownd: Update tags for production
README	183 bytes	42	16 months	brownd: Add README
FullSim		175	2 weeks	stroili: add README file
V0.0.0		175	2 weeks	stroili: add README file
Management		91	7 months	stroili: add new code repositories
TestRelease		174	2 weeks	stroili: add README file
README	1.2 kB	46	16 months	brownd: Move tools to ReleaseTools? repository

Note: See [TracBrowser](#) for help on using the browser.

[view changes...](#)



Releases

- a release is defined by a file called **Packages**
 - for bootstrap release V0.0.0

```
BRN/BrnApp/trunk  
BRN/BrnCore/trunk  
BRN/Brn3BGen/trunk  
Core/BaBar/tags/V00-00-00  
PACRAT/PackageList/trunk  
ReleaseTools/SoftRelTools/trunk  
ReleaseTools/SvnTools/tags/V00-00-12  
ReleaseTools/workdir/tags/V00-00-02
```

subversion repository

tag

Core/BaBar/tags/V00-00-00

package name

Releases

- once there'll be a true FullSim release it will be available compiled and packaged in RPM's (one per package)
- it will live in

```
/opt/exp_software/superb/releases/FullSim/VX.Y.Z
```

HOWTO start

- to work with the bootstrap release you have to download the full release
- to download the release there're a few steps that you have to follow:

1) install superb-scripts RPM

- it provides some useful tools to setup the environment variables and to interact with subversion

2) create your development area

```
/opt/exp_software/superb/bin/sbnewrel -r FullSim/V0.0.0 \  
-d MyFullSim
```

HOWTO start

3)cd to MyFullSim

- in your development area you will find a `GNUmakefile`, some work directories (`include`, `man`, `workdir`) and 2 packages:
 - **ReleaseFiles**: where the current release is defined
 - **SvnTools**: it contains some tools to interact with subversion

4)download the bootstrap release

```
SvnTools/sbaddpkg
```

- you will find all the packages that define your release with some other directories required by SRT

5)now you can build the application

```
make
```

HOWTO start

- don't worry if you see some messages when you add the packages

```
nbstroili: [2:34pm] [~/tmp/Bruno] SvnTools/sbaddpkg
URL: https://sbrepo.pd.infn.it:8910/Releases/FullSim/V0.0.0
Will checkout BRN/BrnCore/trunk
GNUmakefile:311: OS_check.mk: No such file or directory
GNUmakefile:586: babar_specials.mk: No such file or directory
GNUmakefile:1182: link_all_reco.mk: No such file or directory
GNU Make 3.81
Build OPTIONS = Linux26FC11_i586_gcc440-Debug-native-Optimize-Fastbuild-
SkipSlaclog-Static-Lstatic
Linux nbstroili 2.6.30.9-99.fc11.i586 #1 SMP Tue Nov 17 21:10:57 EST 2009
i686 i686 i386 GNU/Linux [uname -a]
[Warning]: ./bin/Linux26FC11_i586_gcc440 (and/or)
/opt/exp_software/superb/releases/V0.0.0/bin/Linux26FC11_i586_gcc440 is not
in PATH, type 'srtpath' to fix PATH.
Note: SBSITE undefined, no site configuration will be used
-> installdirs:
Package ReleaseTools/workdir/tags/V00-00-02 already exists
Will checkout ReleaseTools/SoftRelTools/trunk
GNUmakefile:1182: link_all_reco.mk: No such file or directory
GNU Make 3.81
Build OPTIONS = Linux26FC11_i586_gcc440-Debug-native-Optimize-Fastbuild-
SkipSlaclog-Static-Lstatic
Linux nbstroili 2.6.30.9-99.fc11.i586 #1 SMP Tue Nov 17 21:10:57 EST 2009
i686 i686 i386 GNU/Linux [uname -a]
```

Tools

- most of the tools have some help

```
nbstroili: [2:38pm] [~/work/FullSim] SvnTools/sbaddpkg -h
Usage: addpkg [-tlnpv] [-R release] [-r repository] [-f file] <package>

Options:
  -t, -H, --trunk           Take the svn trunk version
  -f file, --file           Read list of /repo/package/path from file
  -n, --dryrun              do not execute, just show messages.
  -p, --noprompt           do not prompt user if package exists, just update it.
  -r, --repo                Specify repository. Otherwise repository name must
                           be specified as part of package name.
                           This option only applies to package specified on the
                           command line and cannot be used in conjunction with
                           the -f option.
  -u, --update             update release description before running
  -v, --verbose            verbose mode

Arguments:
  <package>  package name(s)

If no <package> is specified, get all packages listed for the current release.
```


Tools

- some useful tools
 - **addtag**: to create a tag for a given package

```
nbstroili: [2:44pm] [~/work/FullSim] SvnTools/addtag -h
Script to create a tag for a package.

Usage: addtag [-p PACKAGE] [-c] [--noprompt] [-v] [-h]

Options:
  -p, --package          package
  --vtag                 official tag to create
  -u, --utag            user tag to create
  -m, --message         svn commit message
  --noprompt            do not prompt user
  -v, --verbose         verbose mode
  -n, --nextvtag        create next sequential vtag
  -d, --dryrun          do not execute, just show messages
  -h, --help            print this help
```

Tools

- **switchtotrunk**: to refer the package to its trunk
 - if you want to commit your changes you have to do to the trunk not to a tagged version

```
nbstroili: [2:44pm] [~/work/FullSim] SvnTools/switchtotrunk -h
Script to switch PACKAGE in the current directory to the trunk of its repository.
If no revision is specified the package will be switched to the HEAD of the trunk
otherwise it will be switched to the specified revision.

Usage: switchtotrunk -p PACKAGE [-r REVISION]

Options:
  -v, --verbose          verbose mode
  -h, --help            print this help
  -p, --package PACKAGE specify the package
  -r, --revision REVISION specify the revision
```

Trac

- all the subversion repositories have a trac site
 - it's a useful tool for
 - browsing the code in the repository
 - looking at the commits
 - comparing the code revisions
 - it has the following capabilities:
 - wiki
 - ticketing/bug track
 - discussion
 - blog






`https://sbrepo.pd.infn.it:8911/projects/`



root

View revision:

- Wiki
- Timeline
- Roadmap
- Browse Source**
- View Tickets
- New Ticket
- Search
- Blog
- Admin
- Discussion
- Tags

Name ▲	Size	Rev	Age	Last Change
▶  FastSim		179	6 days	brownd: Update tags for production
▶  FullSim		175	2 weeks	stroili: add README file
▶  Management		91	7 months	stroili: add new code repositories
▶  TestRelease		174	2 weeks	stroili: add README file
 README	1.2 kB	46	16 months	brownd: Move tools to ReleaseTools? repository

Note: See [TracBrowser](#) for help on using the browser.



Mailing lists

- a set of mailing lists is available for communications under sympa (`@lists.infn.it`):
 - **superb-releases**: announcement for RPM releases
 - **superb-repo**: commit messages are automatically sent here
 - **superb-softadmin**: to communicate general problems with subversion, tools, RPM's, ...