



Recent merging of ASAP3.0 to alma branch

...long and winding road...

Kana Sugimoto (ALMA Project, NAOJ)



Overview of the last merge

The last merge

- When?: 21 May - 9 Jun, 2010 ~ 3 weeks!!
- Merged Revisions:
 - ATNF trunk ... release 3.0.0
 - alma branch ... 1752

Philosophy ... **Implement as much as I can.**

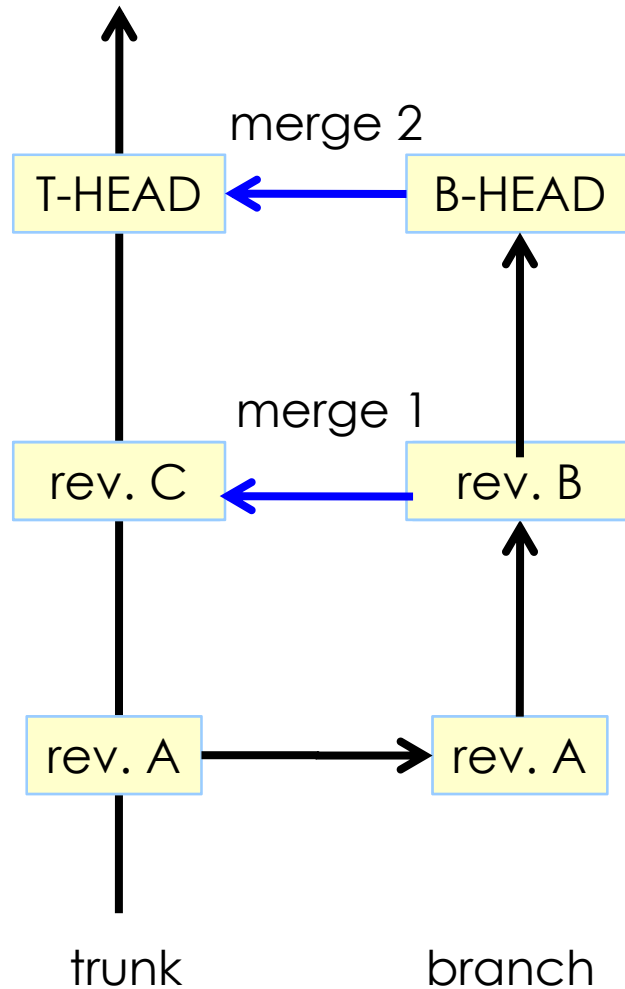
...but, its not easy.

I can NOT just simply “svn merge”... physical/effective collisions + modifications close in lines

We do many developments independently in both codes
→ **Needed to merge many codes manually... tough**

svn merge

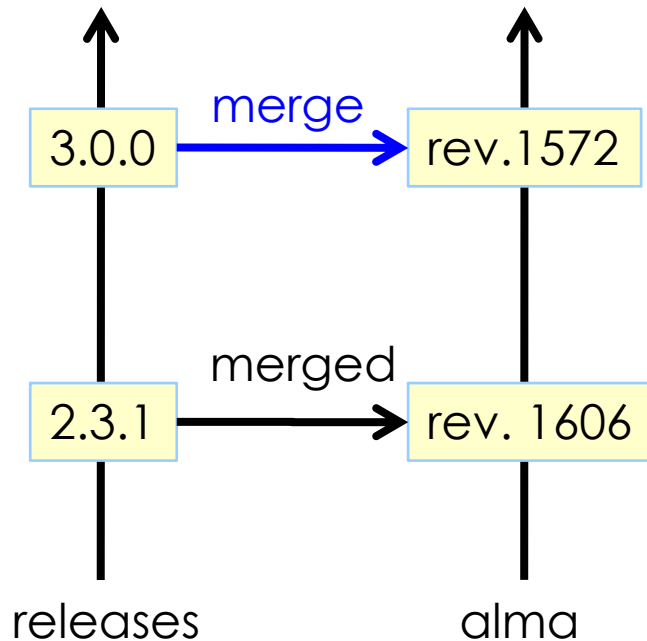
Apply the differences between two sources to a working copy path



merge 2:
merge developments between
rev. B and HEAD of the branch.
`svn merge -r revB:HEAD`
`http://path/to/the/branch`
at a working copy of T-HEAD

merge 1:
merge developments between
rev. A and HEAD of the branch.
`svn merge -r revA:HEAD`
`http://path/to/branch`
at a working copy of T-HEAD

Merging strategy



Can I merge developments between ASAP 2.3.1 and 3.0.0 to the HEAD of alma branch by simply "svn merge"?

Unfortunately NOT!!

There are many collisions both physically and effectively.

I only have to put my eyes on python/, src/, and external/!

```
[kana@casadev1 alma-merged300]$ ls
admin/  doc/  getsvnrev.sh*  plugins/  share/  web/
apps/  Doxyfile  INSTALL  python/  src/
bin/  examples/  Makefile  scons/  test/
COPYING  external/  monitor/  SConstruct  tutorials/
```

→ A snapshot. Effective codes are in CASA



The other things to keep in mind

There are some other differences between CASA-ASAP and ATNF-ASAP:

1. Build system
alma: (g)make with Makefiles
ATNF: scons
2. Versions of third party packages
python: 2.5.2 (alma) will be updated to 2.6 for the next release
matplotlib: 0.91.4 (alma) while ATNF assumes ≥ 0.98
casacore: implements developments by CASA group
3. CASA-ASAP is based on CASA codes
CASA logger , filler codes (libatnf)



Actual steps

1. “merge” developments between ASAP 2.3.1→ 3.0.0 to codes in alma branch except for python/, src/, and external/.
2. merge or apply patches for codes in python/, src/, and external/ if it's possible, i.e., no collision.
Added: 11 (python 5, src 6)
Modified: 63 (python 12, src 25, external 26)... most of them are modified in both branch
3. implement developments by hand for codes which don't accept patches (27/63)
4. update Makefiles (if necessary)
5. update CASA codes calling ASAP functions(if necessary)
6. build tests, bug fixes, regression tests, and finally... commit!!



Things not implemented to alma

1. printing logs by the decorator function, `@print_log_dec`.
in `asapfitter.py`, `asapmath.py`, `asapplotter.py`, `asapreader.py`, and
`scantable.py` (commented out)
→ unify log system or wrap the difference?
2. plot functions using new features of Matplotlib > 0.91.
`get_region` & `get_point` in `asaplotbase.py`,
keyword `interactive` not accepted for axes funcs in `asapplotter.py`
→ Matplotlib in CASA is planed to be updated in CASA 3.2.
3. Translation of `SRCTYPE` in `STWriter` ... CASA-ASAP understands
more source types defined in `PSKIO/SrcType.h`.
4. several files in `external/atnf` is not implemted to CASA-ASAP
(`PKSmsg` & `pksmb_support`)
5. bit more differences in plotter
fontsize definition (`@asaplotbase`), `scantable` parameter remains in
`plotazel` & `plotpointing` (`@asapplotter`)



Summaries and for Future... (1/2)

The ATNF-ASAP 3.0.0 developments are implemented to ALMA branch (rev.1757)... **good!**
But it required manual merging by hand for codes in python/, src/, and external/ ... **too bad!!**

This happens because...

1. we develop independently w/o communications
→ requires resolution of collisions by hand
2. long time passed since we branched and many have been changed in both codes
→ disables merging by applying patches



Summaries and for Future... (2/2)

So what shall we do? ...2 extremes + an idea in between

1. **never merge again** ... easy
2. **unify repository** ... need arrangement + developments (logger, build system, ...what else?)
3. **more communications + more frequent merge to each other**

Any other ideas?



終
Thanks