Proposal for Managing Future Releases

David Jaffe, Brett Viren

Physics Department

BROOKHAVEN NATIONAL LABORATORY

09/05/11 DB US Offline

Lessons learned from last release

Identified problems:

1. Original trunk state was not stable.
2. Expanded scope of release at last minute.
3. Listing required trunk commits to back-port in trac ticket is inefficient.
4. Trunk & release branches diverged due to long delays. Positive–feedback loop.

Some good things:

1. Fast bug fix release (1.2.1)

Obvious improvements for future releases

- Every production cluster must run bitten in order to better identify problems in trunk and get them fixed ASAP.
- Limit scope of releases and disallow any broadening after it is defined
  - Possibly make more frequent, smaller releases.

Significant changes in release process

Focus effort on making the release:

- Take releases seriously and get them finished ASAP
- No release candidates, release directly from trunk
- Inspiration from Linux kernel and MINOS

Steps:

1. Implement required scope features in trunk.
2. Declare feature-freeze on trunk
   - everyone should focus on release but may also continue development on individual branches
3. Satisfy release requirements (see next)
4. Make release branch and declare feature–freeze over
5. If important post-release bugs found:
   - apply fix to both trunk and the release branch,
   - promptly make new minor release.

Release requirements

Must compile on all production clusters

- bitten assures trunk "always" compiles so mostly just happens

Nose tests must pass on all production clusters

- Maintain cluster contact list in wiki
- Contact must report test results within two working days

Production clusters:

- must include the lab clusters and may include any university clusters that can satisfy the above.
- lack of response will not delay a release