April 2009 Offline Software Upgrade

Brett Viren

Physics Department



09/05/11 DB US Offline

Outline

Overview

How To Install and Test

Improved Installation of Externals

Improved Platform Support

Summary

Upgrade motivations

- Support GCC 4.3 needed by Debian 5.0+, Fedora 9+, SL 5.3+
  - Stricter C++ standard compliance
  - C/C++ header cleanups
  - Fixed upstream, don't want to backport these fixes to our old versions.
- Have fallen behind by major versions in many important packages.
- Opportunity to clean up / simplify dybinst.

Upgrade mechanics

1. April 1, branch dybinst and projects into "`upgrade`" branches.
2. Since then: merge upstream changes into our copies in SVN, test and fix on 9 platforms:
   - Ubuntu Hardy (8.04) and Intrepid (8.10)
   - Debian Etch (4.0) and Lenny (5.0)
   - Mac OS X 10.5 (thanks, David)[1]
   - Fedora 9 (thanks, Christine)
   - SLC 4.4, 4.5 & 5.2 (thanks, Wei for 5.2)
3. Now: open up to everyone to install "upgrade" branches for 1 week.
4. +1 week: fix to any problems, merge "upgrade" branches back into `trunk`.

[1] And Laur for the use of his Mac workstation.

Basic Installation and Testing

Almost just like installing "trunk" or a release, but you must use "dybinst" from the "upgrade" branch. You can install this in a new location or under the same area of other releases.

```
DYBINSTURL=http://dayabay.ihep.ac.cn/svn/dybsvn/installation
svn export $DYBINSTURL/branches/upgrade/dybinst/dybinst
./dybinst upgrade all
```

Run the code like normal:

```
cd NuWa-upgrade/
source setup.sh
cd dybgaudi/DybRelease/cmt
source setup.sh
cd $ROOTIOTESTROOT
nosetests
```

Improvements in Externals

Use LCG/AA's LCG_Builders instead of our scripts.[2]

`external/tarFiles/` local cache of source code

`external/build/LCG/*` where the packages are built

`external/*` where they are installed (as before)

Source tar files are pulled from:

`http://dayabay.bnl.gov/software/offline/tarFiles/`

To pick a different cache, mirror the above and do:

```
./dybinst upgrade cmt checkout
emacs NuWa-upgrade/lcgcmt/LCG_Builders/LCG_BuildPolicy/cmt/requirements
# change LCG_tarurl macro
./dybinst upgrade external projects
```

To do: add explicit, site-dependent local caches chosen based on a tag.

[2] A couple packages still rely on our scripts.

Improvements in Platform support

No more "`uncharacterized_linux`"!

- Must now have explicit platform support
  - less mysterious, more explicit failures
- `CMTCONFIG` can be used to select different build platforms
  - cross compilations
- `LCG_system` (binary subdirectory) name matches `CMTCONFIG`.
- Use emerging standard for `CMTCONFIG` names:

`CPU-OS-COMPILER-[opt|dbg]`

Optimization vs. Debug

The last part of `CMTCONFIG`:

- `*-dbg` Debugging symbols are included in the object code.
- `*-opt` No debugging symbols and turn on compiler optimization

Notes:

- Default is `*-dbg`. Slow, but assists bug fixing.
- Explicitly set `CMTCONFIG` before running dybinst for `-opt`.
- Controls how externals are built too.

Adding Support for New Platforms

New platforms are introduced by adding tags to `lcgcmt/LCG_Settings/cmt/requirements`.

`host-*` describe the host platform (determined from implicitly defined CMT tags).

`target-*` describe the target platform for which to build (explicitly set based on `CMTCONFIG`).

Tautological bootstrapping of the platform definition:[3]

- `CMTCONFIG` determines top-level `target-*` macros.
- default `CMTCONFIG` is set in `NuWa-RELEASE/setup.sh`
- `dybinst` writes `NuWa-RELEASE/setup.sh` during installation.
- Initial value based on implicitly defined `host-*` values.

So, the platform is auto-detected unless user wants to force it to be something else. Dybinst will warn you in the latter case.

[3] This method has been submitted to LCG/AA for inclusion upstream.

Adding Support for New Platforms, cont.

Adding support to `LCG_Settings/cmt/requirements`:

1. Make sure each section of `HOST TAGS` includes relevant descriptions
2. Under `TARGET TAGS`, make sure your canonical `CMTCONFIG` tag is present and implies relevant `target-*` tags. Add additional ones under the Daya Bay specific areas.
3. Add to `host-compiler`, `host-cpu`, and `host-os` if needed.
4. Make sure `LCG_basesystem` includes `target-*` tags that imply your canonical `CMTCONFIG` value (minus the trailing `-dbg/opt`).

Troubles?

1. Add your platform's implicit tags to:
   `https://wiki.bnl.gov/dayabay/index.php?title=CMT_tags`
2. Add trac ticket:
   `http://dayabay.ihep.ac.cn/tracs/dybsvn`

Summary

- Upgrade branches built and tested on 9 platforms.
- Improvements made to the process.
  - Better/standard build system for "external"
  - Platform support more robust
- Ready for anyone to test.
- If no problems, will merge to trunk ∼20 May.

Some notes on the upgrade are here:

`https://wiki.bnl.gov/dayabay/index.php?title=Software_Upgrades`