



Build System Overview and Discussion



What is a “Build System”?

- An automated system that
 - > detects new CVS commits;
 - > accepts and analyzes build requests;
 - > fetches source code from the CVS repo;
 - > prepares build environment;
 - > build the products;
 - > stores the built artifacts somewhere;
 - > reports build status;

Why do we need a “Build System”?

- Factors:
 - > We have several products
 - lustre, ldiskfs, e2fsprogs
 - > Our products have branches
 - lustre: bl_4, bl_6, HEAD, etc
 - > We support many platforms
 - RHEL/SLES/..., i686/x86_64/ia64/ppc/...
 - > We need quick deployment during testing
- But it takes time and effort to...
 - > do this manually
 - > build all the needed packages for all platforms

A look into the past

- LTEST — An “almighty God”
 - > accepts test request
 - > 24x7 baseline testing
 - > a built-in build component
 - > installs the built packages
 - > sets up testing nodes with proper env
 - > sets up lustre filesystem
 - > does testing
 - > reports testing results
 - > ...
- Doing builds is a part of LTEST

A look into the past (Cont'd)

- It works
 - > All the glorious old days...
- But...
 - > QA are bundled to maintaining LTEST
 - > Extremely slow builds
 - NO dedicated build nodes
 - Do builds sequentially
 - Ccache is located in an shared NFS directory
 - Spent hours to build for a platform even with the help of distcc.
 - Sometimes 20+ hours!!!
- The demise of LTEST

The rise and fall of Frog team

- The origins of Frog
- Goals of Frog
 - > LRE — Lustre Runtime Environment
 - > LTS — Lustre Testing System
- Deliverables of Frog
 - > LTS
 - Don't be misled by its name!
 - A build system backend
 - > LBATS — Lustre Build and Testing System
 - Again, don't be misled by its name!
 - It does neither build nor testing
 - A web based frontend for SLURM/LTS

Current build system - Overview

- Components
 - > LBATS
 - > SLURM
 - > LTS
 - > Build Nodes
 - > OS Images
 - > File cache

Current build system - LBATS

- > A Michael McDonald product
- > Written in Ruby on Rails
- > A web interface for users
- > Accepts build requests
- > Dispatches request to SLURM
- > Shows request status
- > Exports built artifacts: logs/packages/...
- > Maintains user's patchlists
- > Can build lustre and e2fsprogs
- > <https://wikis.clusterfs.com/intra/index.php/LBATS>

Current build system - SLURM

- Simple Linux Utility for Resource Management
- Maintained by Lawrence Livermore National Lab
- We use it as a scheduler for our build jobs
- <https://computing.llnl.gov/linux/slurm/slurm.html>

Current build system - LTS

- The one that does the real job
- Mainly written in Perl
- Has a CLI: `lts <cmd> <option list>`
- It
 - > Prepares build environment
 - > CO the source code
 - > Applies custom patches if needed
 - > does the builds
 - > Stores the built packages to file cache

Current build system - Build Nodes

- Head node: `lts-head`
 - > `LBATS/SLURM/LTS/FileCache` live on it
- One dedicated node for each arch
 - > `i686: lts-build-i686-0`
 - > `x86_64: lts-build-x86-64-0`
 - > `ia64: lts-build-ia64-0`
 - > `ppc: lts-build-ppc64-0`
- On the dedicated build nodes
 - > `SLURM/LTS/OSImages` live on it
 - > File Cache is mounted through NFS

Current build system – OS Images & File Cache

- OS Image
 - > A chroot environment for the build job
 - > one for each distro/arch pairs
- File Cache — A place to store
 - > upstream packages
 - > OS Images
 - > Ccache files
 - > built artifacts
 - > ...

the Benefits

- Much faster build speed
- A file cache to look for packages
- Developer has the chance to build their own packages on real boxes
- Relieve QA from doing it themselves
- Ability to detect broken commits quickly

the Issues

- LTS
 - > An example of failed aspirations
 - > A mess in code and structures
 - > Partly re-written by Mike
 - > No one knows much about it :-)
- LBATS
 - > Written in Ruby on Rails
 - > No one is a RoR expert
 - > No one knows much about it :-)
- SLURM
 - > Again, no one knows much about it!

Needed Features

- Customized build options
- Patch list for users
- Solaris package support
- ...

Conclusion & Discussion

- Conclusion
 - > It also works
 - better at build than LTEST
 - > But it just works
- Discussion
 - > Questions?
 - > Thoughts?



Thank You!
yibin.wang@sun.com