BaTLaB

A Continuous-Integration Facility

Building Communities for SISI Workshop
Arlington, VA - Oct 2011

Todd Tannenbaum

Center for High Throughput Computing

University of Wisconsin-Madison

Today’s Overview of BaTLab

- What
- Why
- How
- Experience
- Coming next
- Get started

WE'RE GOING TO TRY SOMETHING CALLED AGILE PROGRAMMING.
scottadams@aol.com
www.dilbert.com
THAT MEANS NO MORE PLANNING AND NO MORE DOCUMENTATION. JUST START WRITING CODE AND COMPLAINING.
11-26-07 ©2007 Scott Adams, Inc./Dist. by UFS, Inc.
I'M GLAD IT HAS A NAME.
THAT WAS YOUR TRAINING.

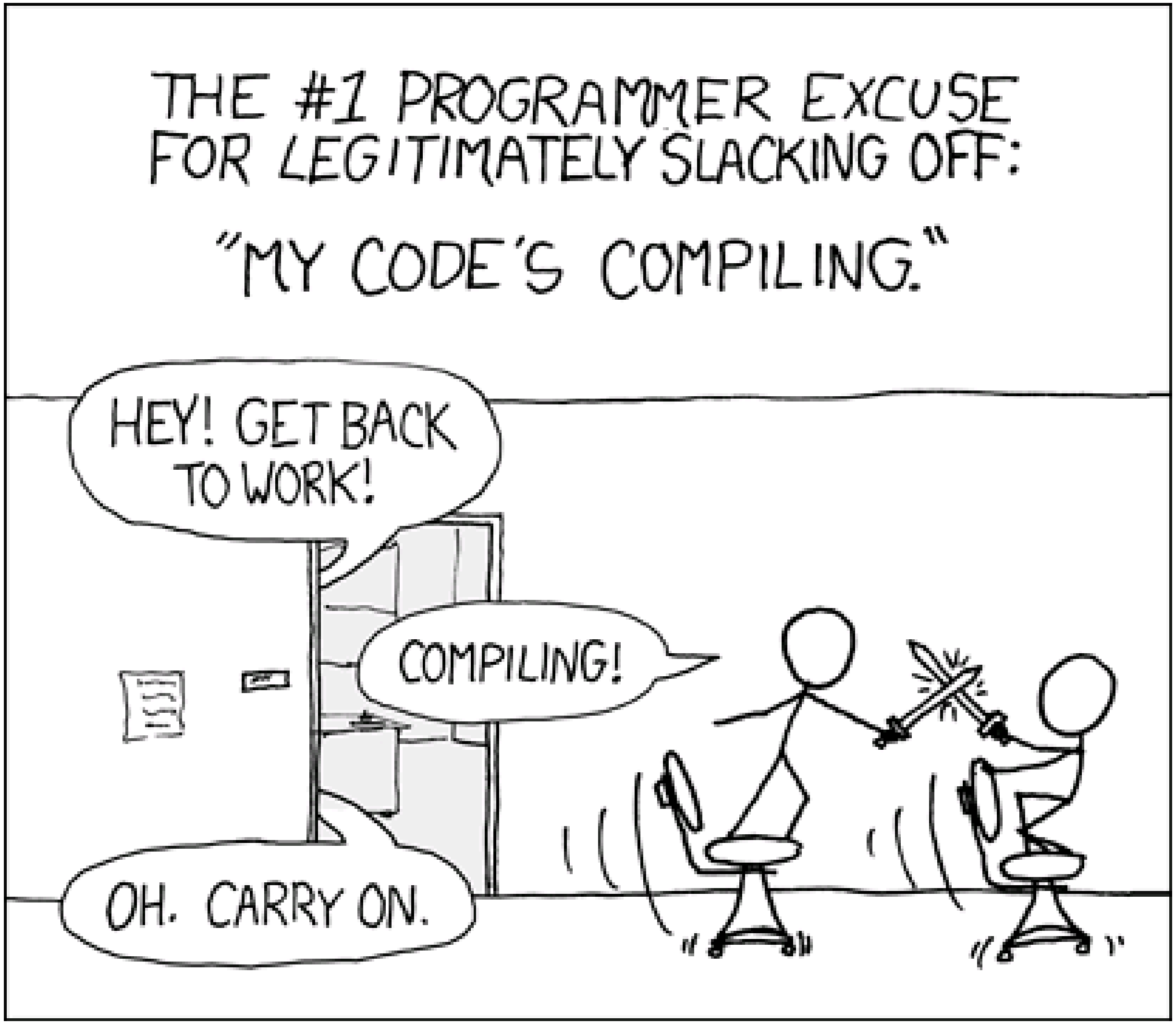
© Scott Adams, Inc./Dist. by UFS, Inc.

What: 10,000 foot view

Build and Test Lab = BaTLab

- Lab Infrastructure
  - many different platforms, professionally managed
- Lab Software = Metronome
  - Performs regular builds and/or tests
  - User specifies source location (ex: web server, CVS, SVN, git, …), platforms to use, declares what to build or test
  - Results stored in RDBMS, reports visible via a web portal

Why?


THE #1 PROGRAMMER EXCUSE FOR LEGITIMATELY SLACKING OFF:
"MY CODE'S COMPILING."
HEY! GET BACK TO WORK!
COMPILING!
OH. CARRY ON.

Escrow

- Can others outside your environment even build it at all?

Continuous Integration

- Detect problems early, before expensive to fix
- Ship releases on schedule
- Find problems before users
- Even if code is stable, changes are happening both above and below the application
  - Changes in OS, dependencies, user expectations

Managed Languages

- Write once, run everywhere? (“WORA”)
- Hint: Below shell script produces different results between Red Hat vs Debian vs BSD, even Debian vs Debian…

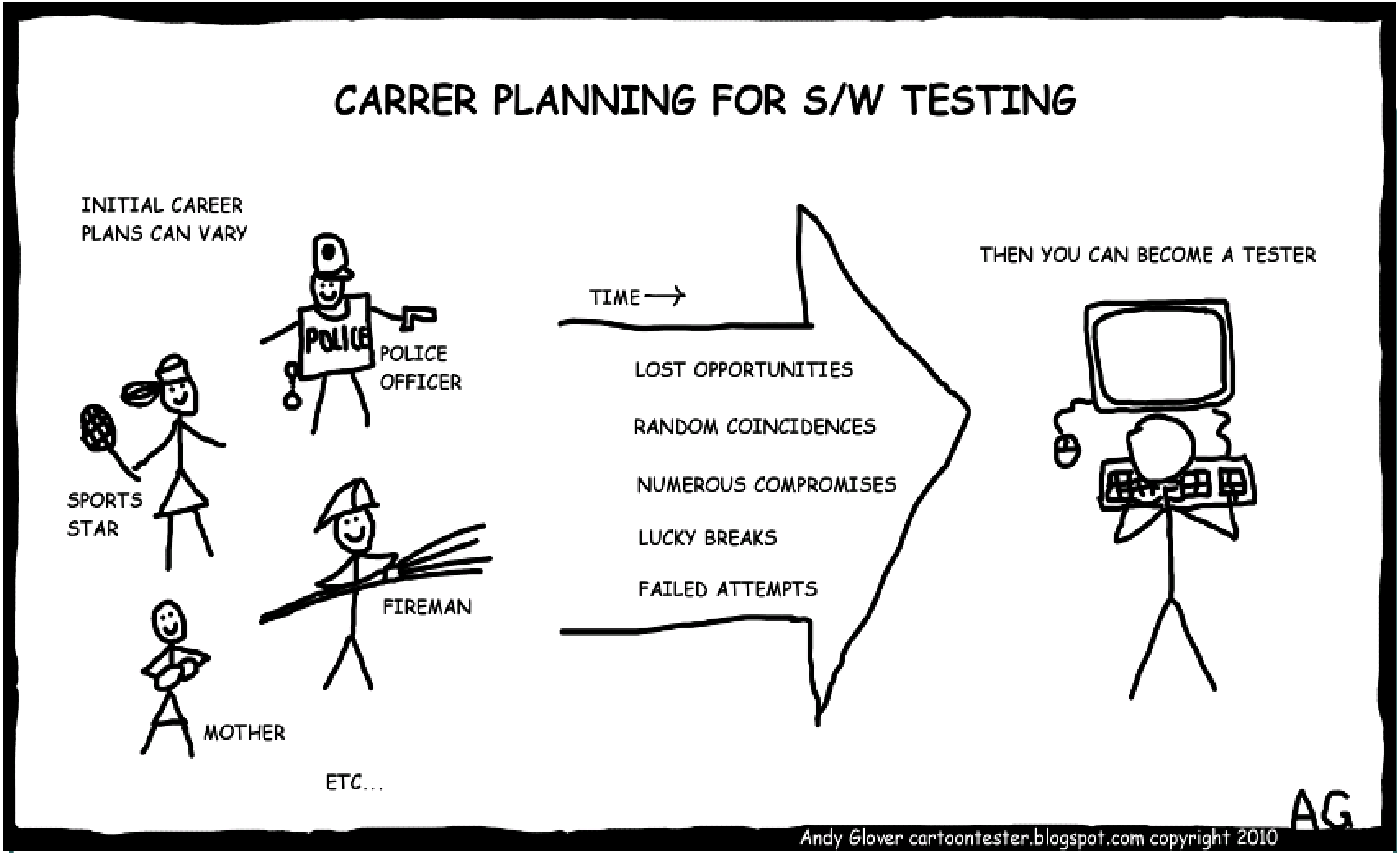
```
#!/bin/sh
echo “Is WORA reality?”
exit -1
```

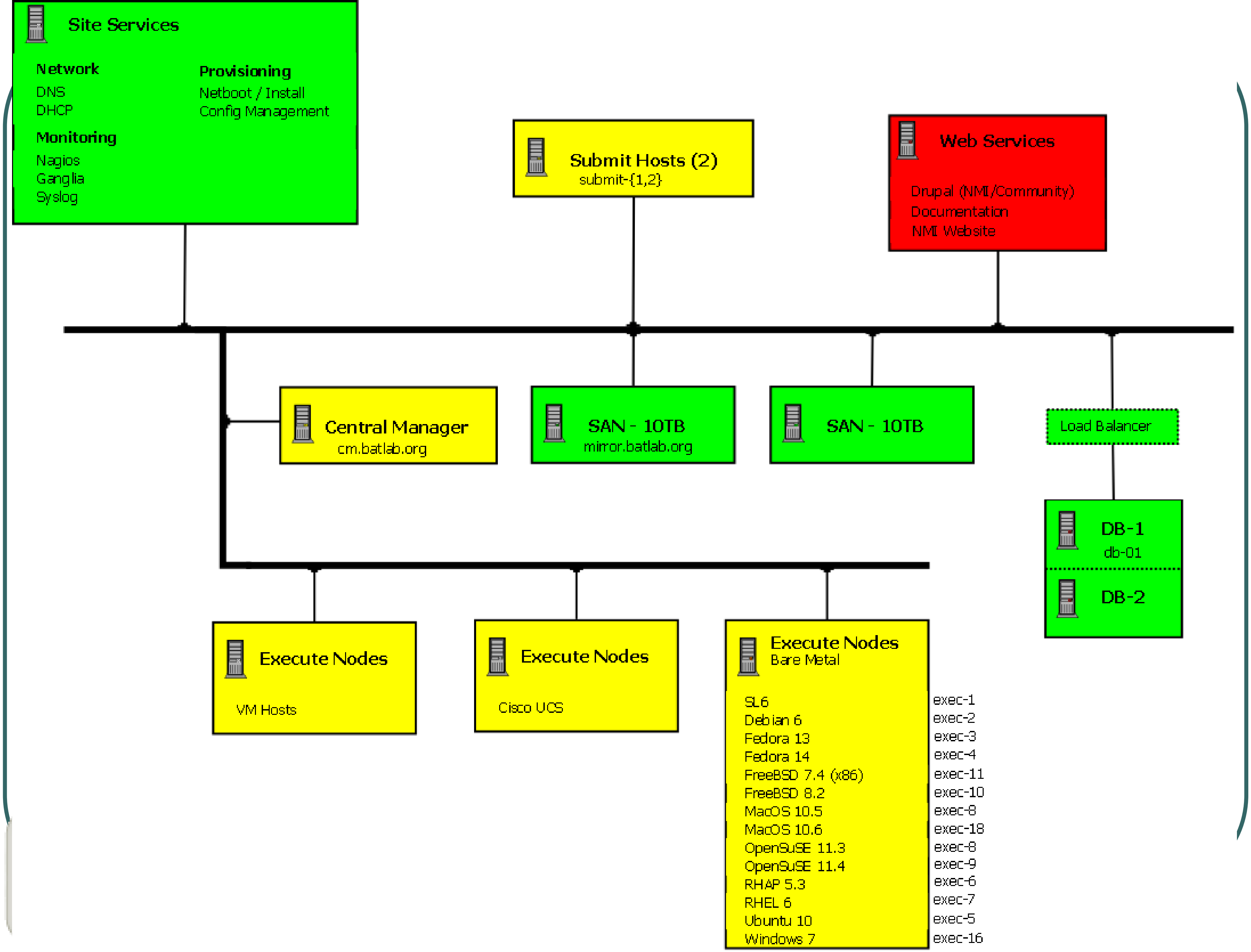
WORA reality

- Still sitting on top of heterogeneous OS environments
- Even if all Linux, different distros have:
  - Different interpreter versions (Ex: bash vs dash)
  - Different kernel, libc versions
  - Different compiler versions
  - Different LSB standards (packaging!)
  - Different library / modules versions
  - Different packages installed by default

Build and TEST!

- Regression tests
- Function vs Unit
- Scalability tests
- “Sweep” tests
- Forward and Backwards compatibility
- Cross versioning


CARRER PLANNING FOR S/W TESTING
INITIAL CAREER PLANS CAN VARY
POLICE
POLICE OFFICER
SPORTS STAR
FIREMAN
MOTHER
ETC...
TIME →
LOST OPPORTUNITIES
RANDOM COINCIDENCES
NUMEROUS COMPROMISES
LUCKY BREAKS
FAILED ATTEMPTS
THEN YOU CAN BECOME A TESTER
AG
Andy Glover cartoontester.blogspot.com copyright 2010



Site Services

Network
DNS
DHCP

Provisioning
Netboot / Install
Config Management

Monitoring
Nagios
Ganglia
Syslog

Submit Hosts (2)
submit-{1,2}

Web Services
Drupal (NMI/Community)
Documentation
NMI Website

Central Manager
cm.batlab.org

SAN - 10TB
mirror.batlab.org

SAN - 10TB

Load Balancer

DB-1
db-01

DB-2

Execute Nodes
VM Hosts

Execute Nodes
Cisco UCS

Execute Nodes
Bare Metal

SL6	exec-1
Debian 6	exec-2
Fedora 13	exec-3
Fedora 14	exec-4
FreeBSD 7.4 (x86)	exec-11
FreeBSD 8.2	exec-10
MacOS 10.5	exec-8
MacOS 10.6	exec-18
OpenSuSE 11.3	exec-8
OpenSuSE 11.4	exec-9
RHAP 5.3	exec-6
RHEL 6	exec-7
Ubuntu 10	exec-5
Windows 7	exec-16

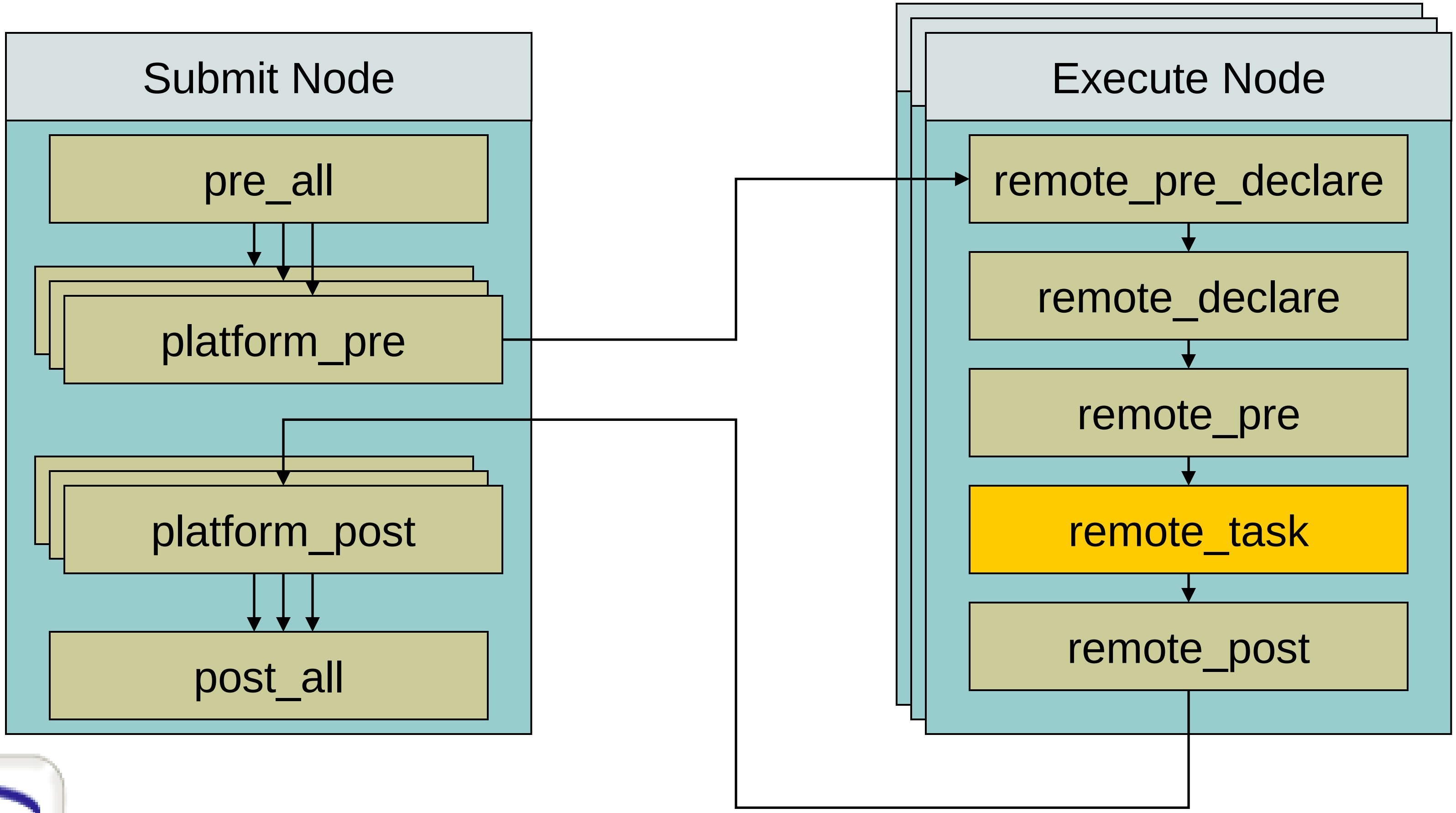
BaTLab Infrastructure

- ~50 unique platforms for builds/tests
- Web portal (http://nmi.cs.wisc.edu)
- 4 submit hosts
- Database cluster
- Backup server
- Network management (DNS, DHCP, SSL)
- Monitoring (Nagios, Ganglia)
- Internal Infrastructure (Condor, …)

How?

- Get a BaTLab account
- Create Metronome “glue” scripts
- Submit Metronome build or test jobs

Metronome Framework for callouts to glue scripts



Metronome Submit File

- run_type = build
- remote_task = hello
- inputs = hello.input
- platforms = x86_64_winnt_6.1, \
  x86_64_deb_6.0, x86_64_rhap_6.1

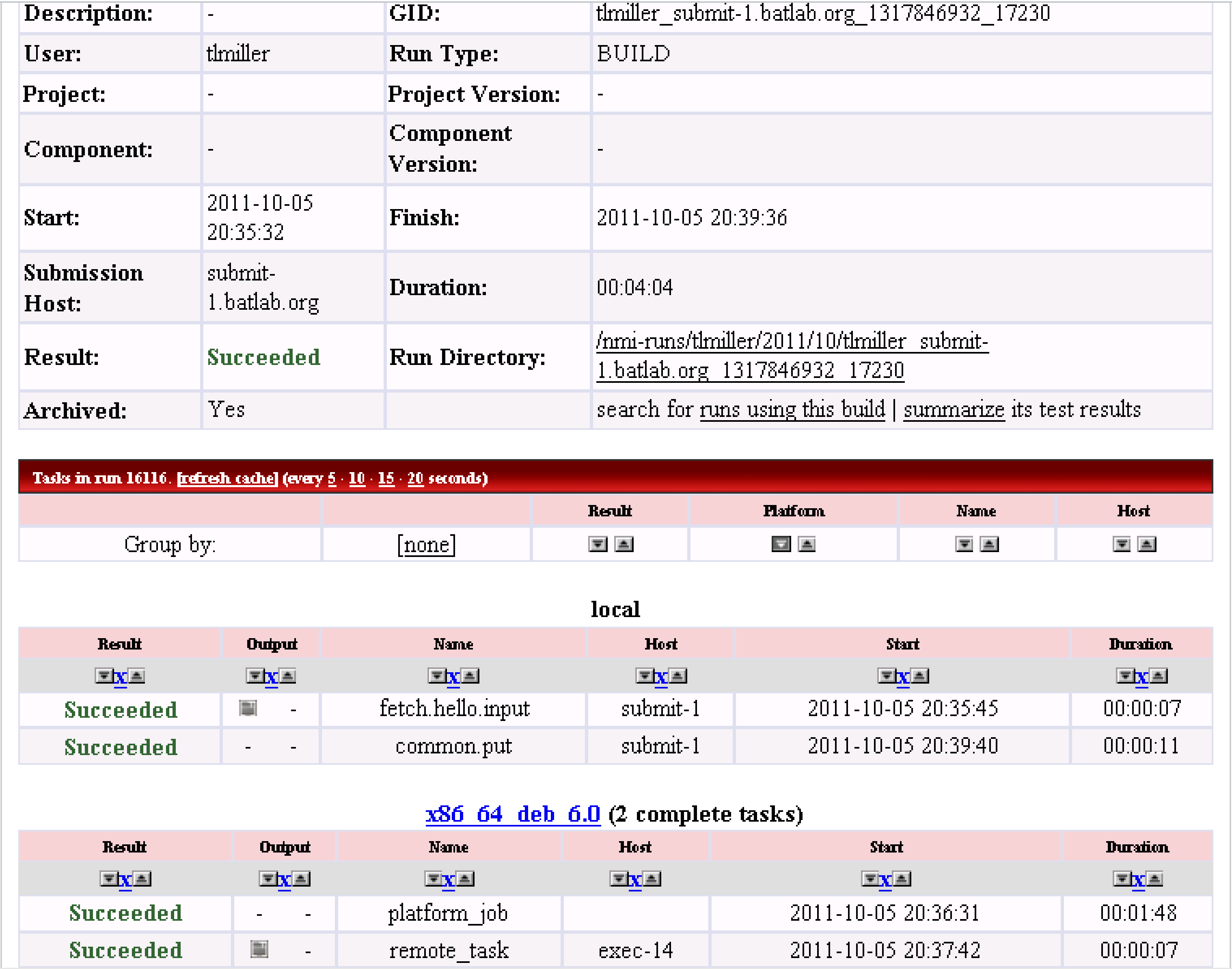
Metronome Input File

- method = scp
- scp_file = me@mymachine:/path/to/hello

Metronome Command Line

- $ nmi_submit hello.submit

```
Global ID:      long_global_id
Run Directory:  /path/to/run/directory
All jobs submitted.  Waiting for monitor to get Run ID from
database......
Run ID:         16116
http://submit-1.batlab.org/nmi/results/details?runID=16116
```



Description:	-	**GID:**	tlmiller_submit-1.batlab.org_1317846932_17230
User:	tlmiller	**Run Type:**	BUILD
Project:	-	**Project Version:**	-
Component:	-	**Component Version:**	-
Start:	2011-10-05 20:35:32	**Finish:**	2011-10-05 20:39:36
Submission Host:	submit-1.batlab.org	**Duration:**	00:04:04
Result:	**Succeeded**	**Run Directory:**	/nmi-runs/tlmiller/2011/10/tlmiller_submit-1.batlab.org_1317846932_17230
Archived:	Yes		search for runs using this build | summarize its test results

Tasks in run 16116. [refresh cache] (every 5 · 10 · 15 · 20 seconds)

		Result	Platform	Name	Host
Group by:	[none]	▼ ▲	▼ ▲	▼ ▲	▼ ▲

local

Result	Output	Name	Host	Start	Duration
▼x▲	▼x▲	▼x▲	▼x▲	▼x▲	▼x▲
Succeeded	▣ -	fetch.hello.input	submit-1	2011-10-05 20:35:45	00:00:07
Succeeded	- -	common.put	submit-1	2011-10-05 20:39:40	00:00:11

x86_64_deb_6.0 (2 complete tasks)

Result	Output	Name	Host	Start	Duration
▼x▲	▼x▲	▼x▲	▼x▲	▼x▲	▼x▲
Succeeded	- -	platform_job		2011-10-05 20:36:31	00:01:48
Succeeded	▣ -	remote_task	exec-14	2011-10-05 20:37:42	00:00:07

Building CMake: Run Spec

- run_type = build
- remote_task = build.sh
- inputs = build.scp, cmake.git
- platforms = x86_64_deb_6.0, x86_64_rhap_6.1

Building CMake: Input Specs

- method = scp
- scp_file = /path/to/build.sh

- method = git
- git_repo = git://cmake.org/cmake.git
- git_path = CMake

Building CMake: Build Script

- cd Cmake
- ./bootstrap --prefix=`pwd`/../install \
  - && make \
  - && make install \
  - && cd .. \
  - && tar -c -z -f ./results.tar.gz ./install
- exit $?

Building CMake: command line

- nmi_submit /path/to/cmake.run-spec
  - Global ID
  - Run Directory
  - Run ID
  - Status URL

Experience

Condor Before Batlab

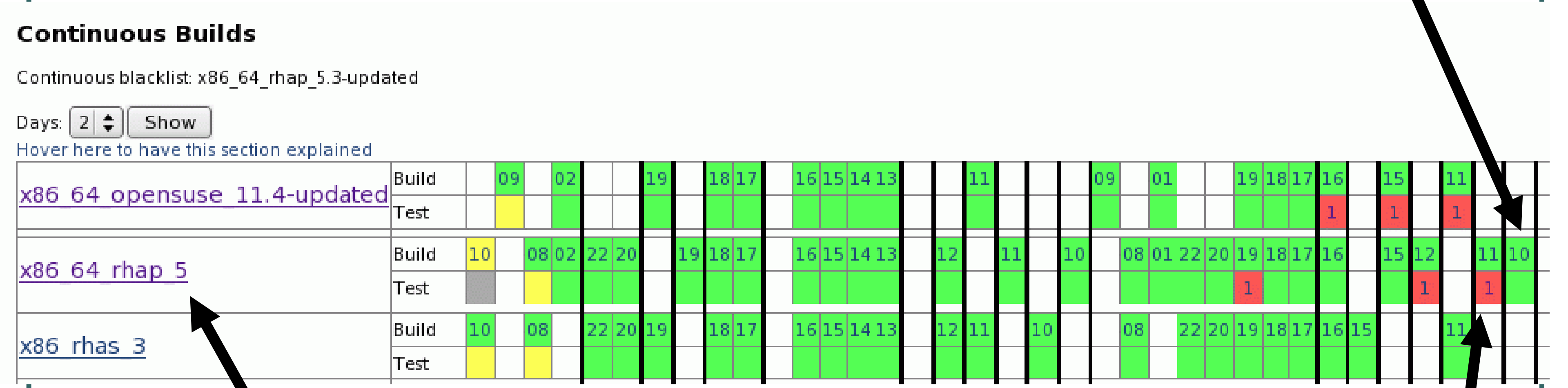
- Either Windows or Linux on desktop
- No Irix, AIX, oddball Linux, Solaris, etc. etc.
- Weeks before release
  - Try to build by hand on each platform
  - Try to fix porting issues introduced weeks earlier
  - Run some tests by hand
- Took weeks just to get a set of binaries
    - Time to fix a bug goes up 10x further away found

First steps

- With Batlab, nightly build on all ports
- Bugs found within 24 hours
  - Usually fixed within 24 – 72 hours
    - Still 24 hour latency on all platforms
    - Test failures much harder to debug than build
- Test failures found within 24 hours
  - Unless masked by build failures (problem)
- Developer one-off ‘workspace builds’
  - Much better than before, but still lots of steps

Web portal snapshot

Green build/test here at 10 am



Continuous Builds

Continuous blacklist: x86_64_rhap_5.3-updated

Days: 2 Show

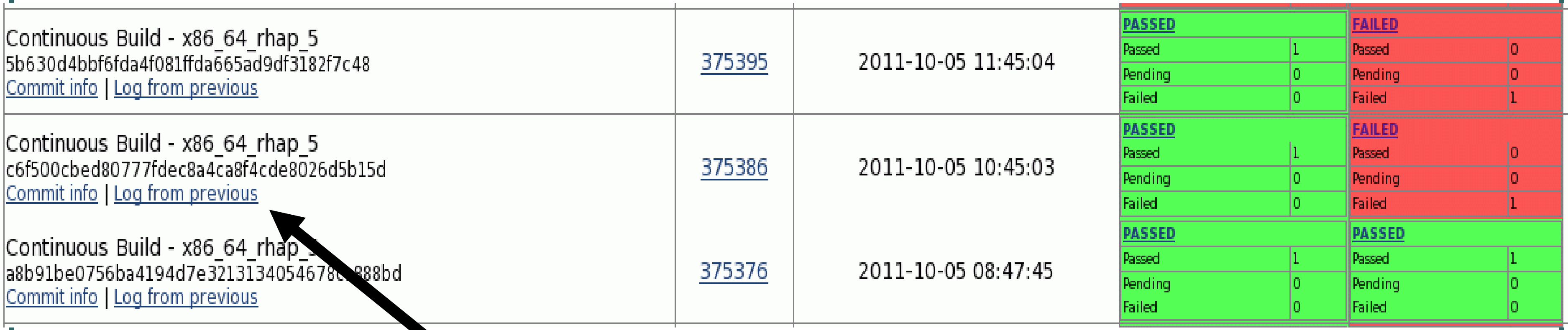
Hover here to have this section explained

x86_64_opensuse_11.4-updated	Build		09		02			19		18	17		16	15	14	13			11				09		01			19	18	17	16		15		11		
	Test		🟨		🟩			🟩		🟩	🟩		🟩	🟩	🟩	🟩			🟩				🟩		🟩			🟩	🟩	🟩	1		1		1		
x86_64_rhap_5	Build	10		08	02	22	20		19	18	17		16	15	14	13		12		11		10		08	01	22	20	19	18	17	16		15	12		11	10
	Test	▨		🟨	🟩	🟩	🟩		🟩	🟩	🟩		🟩	🟩	🟩	🟩		🟩		🟩		🟩		🟩	🟩	🟩	🟩	1	🟩	🟩	🟩		🟩	1		1	🟩
x86_rhas_3	Build	10		08		22	20	19		18	17		16	15	14	13		12	11		10			08		22	20	19	18	17	16	15			11		
	Test	🟨		🟨		🟩	🟩	🟩		🟩	🟩		🟩	🟩	🟩	🟩		🟩	🟩		🟩			🟩		🟩	🟩	🟩	🟩	🟩	🟩	🟩			🟩		

Click here to find out

What happened here?

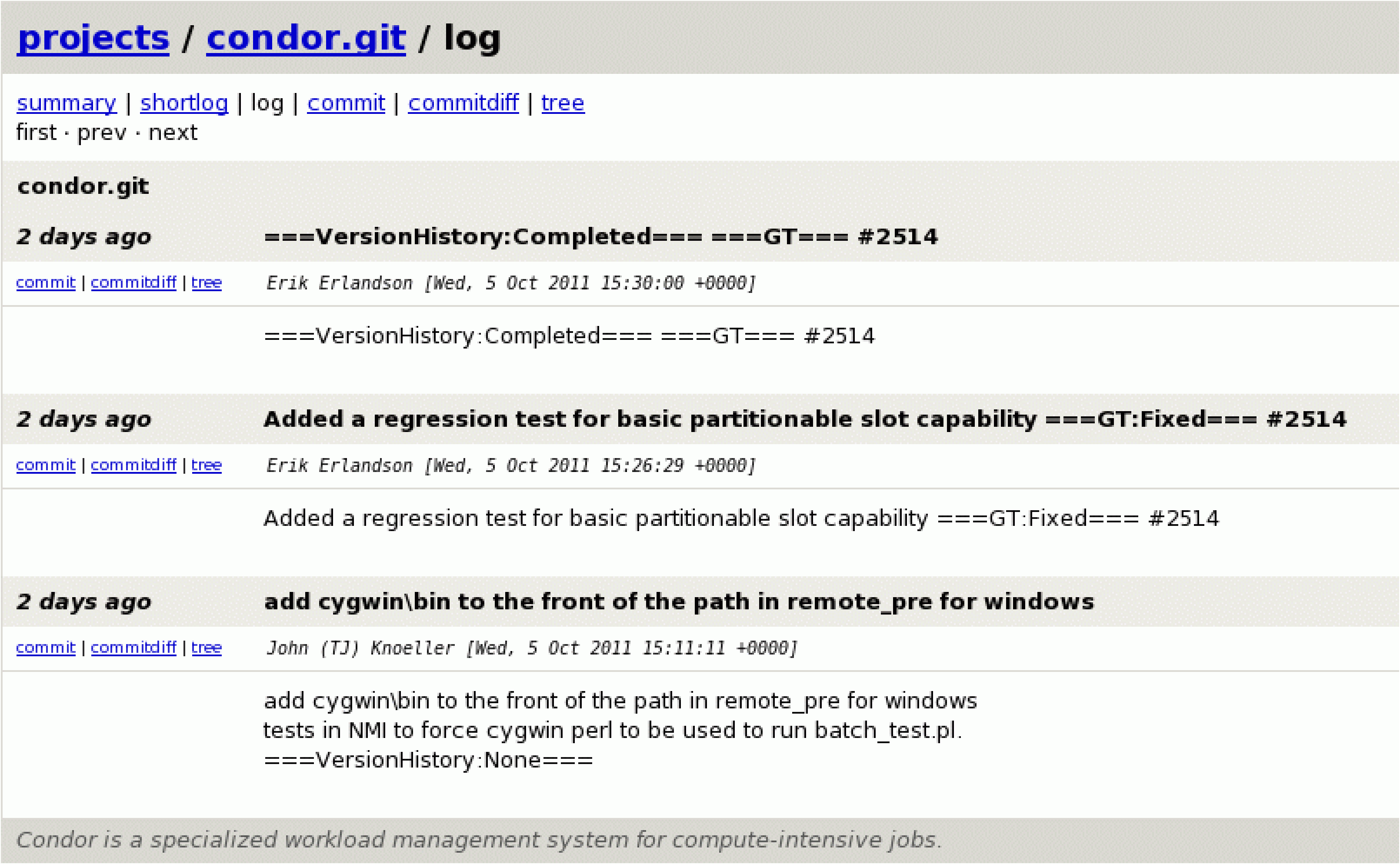
What happened?



Continuous Build - x86_64_rhap_5 5b630d4bbf6fda4f081ffda665ad9df3182f7c48 Commit info | Log from previous	375395	2011-10-05 11:45:04	PASSED Passed 1 Pending 0 Failed 0	FAILED Passed 0 Pending 0 Failed 1
Continuous Build - x86_64_rhap_5 c6f500cbed80777fdec8a4ca8f4cde8026d5b15d Commit info | Log from previous	375386	2011-10-05 10:45:03	PASSED Passed 1 Pending 0 Failed 0	FAILED Passed 0 Pending 0 Failed 1
Continuous Build - x86_64_rhap_5 a8b91be0756ba4194d7e3213134054678c[illegible]888bd Commit info | Log from previous	375376	2011-10-05 08:47:45	PASSED Passed 1 Pending 0 Failed 0	PASSED Passed 1 Pending 0 Failed 0

Click here

Whom to blame?



projects / condor.git / log

summary | shortlog | log | commit | commitdiff | tree
first · prev · next

condor.git

2 days ago **===VersionHistory:Completed=== ===GT=== #2514**

commit | commitdiff | tree *Erik Erlandson [Wed, 5 Oct 2011 15:30:00 +0000]*

===VersionHistory:Completed=== ===GT=== #2514

2 days ago **Added a regression test for basic partitionable slot capability ===GT:Fixed=== #2514**

commit | commitdiff | tree *Erik Erlandson [Wed, 5 Oct 2011 15:26:29 +0000]*

Added a regression test for basic partitionable slot capability ===GT:Fixed=== #2514

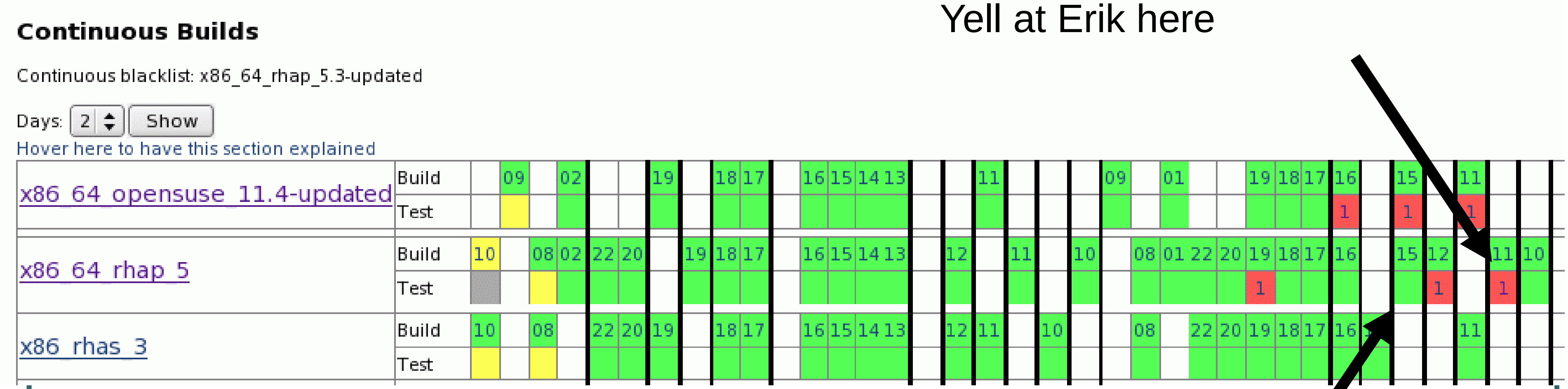
2 days ago **add cygwin\bin to the front of the path in remote_pre for windows**

commit | commitdiff | tree *John (TJ) Knoeller [Wed, 5 Oct 2011 15:11:11 +0000]*

add cygwin\bin to the front of the path in remote_pre for windows
tests in NMI to force cygwin perl to be used to run batch_test.pl.
===VersionHistory:None===

Condor is a specialized workload management system for compute-intensive jobs.

Back in business



Continuous Builds

Continuous blacklist: x86_64_rhap_5.3-updated

Days: 2 Show

Hover here to have this section explained

x86_64_opensuse_11.4-updated	Build		09		02			19		18	17		16	15	14	13			11				09		01			19	18	17	16		15		11		
	Test		🟨		🟩			🟩		🟩	🟩		🟩	🟩	🟩	🟩			🟩				🟩		🟩			🟩	🟩	🟩	1		1		1		
x86_64_rhap_5	Build	10		08	02	22	20		19	18	17		16	15	14	13		12		11		10		08	01	22	20	19	18	17	16		15	12		11	10
	Test	▨		🟨	🟩	🟩	🟩		🟩	🟩	🟩		🟩	🟩	🟩	🟩		🟩		🟩		🟩		🟩	🟩	🟩	🟩	1	🟩	🟩	🟩		🟩	1		1	🟩
x86_rhas_3	Build	10		08		22	20	19		18	17		16	15	14	13		12	11		10			08		22	20	19	18	17	16	1			11		
	Test	🟨		🟨		🟩	🟩	🟩		🟩	🟩		🟩	🟩	🟩	🟩		🟩	🟩		🟩			🟩		🟩	🟩	🟩	🟩	🟩	🟩	🟩			🟩		

Yell at Erik here

Test fixed here

“Hourly builds” on three platforms

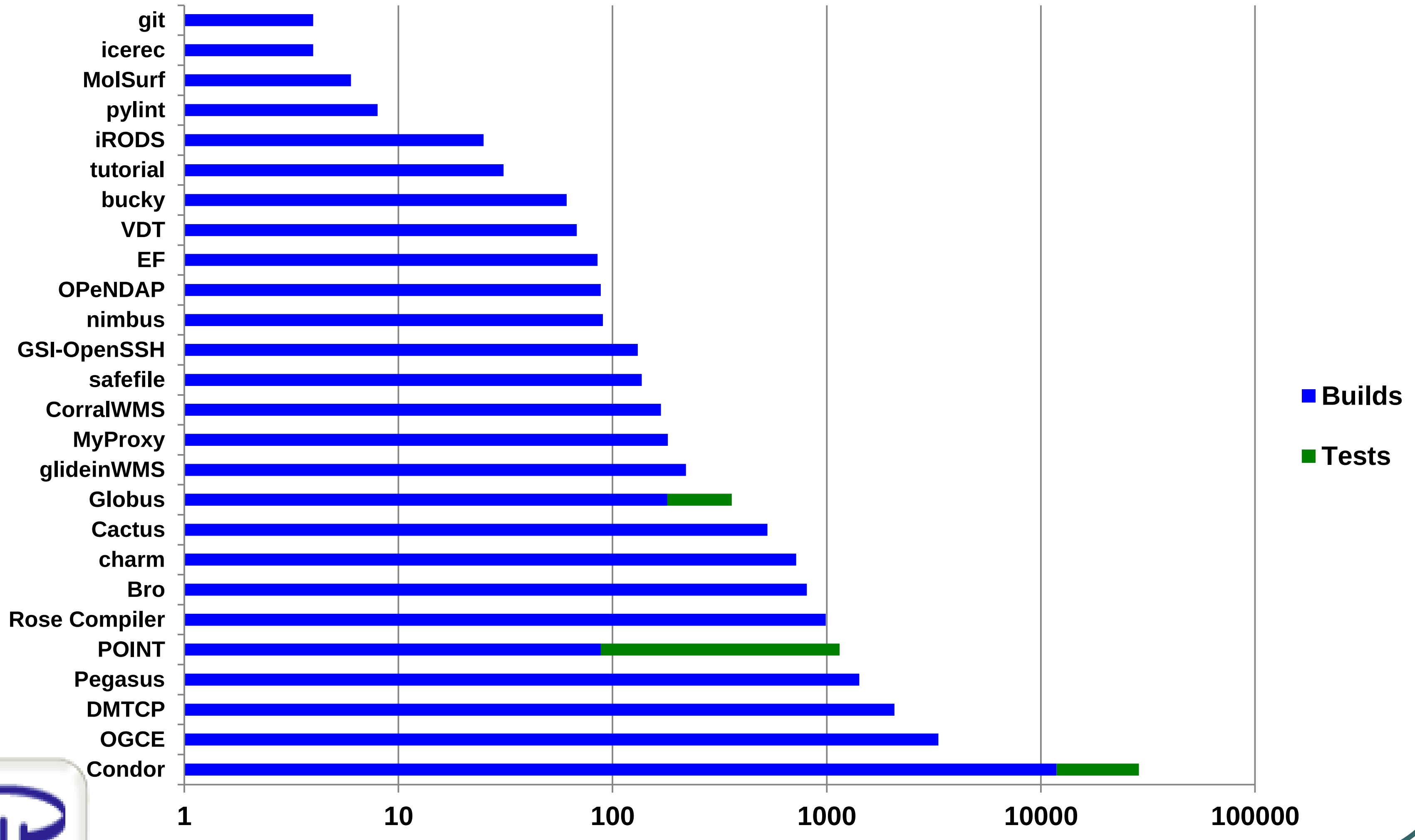
- Builds and esp tests fall behind
  - Soln: JobPrio == Qdate
- Dramatically improved # of green nightly builds – almost always, except for late pushes
  - Lesson learned – more build per day, better

Average Condor month…

- Performed over 170 commits to the codebase
- Modified over 350 source code files
- Changed over 8,500 lines of code (Condor source code written at UW-Madison as of June 2011 now sits at about 922,422 lines of code)
- Compiled about 2,500 builds of the code for testing purposes
- Ran about 930,000 regression tests (both functional and unit)

→ Release a new version

Usage by Project, last 90 days



Usage by Platform, last 90 days

58 Unique Platforms (all x86 64 bit unless noted)

12 Platforms with more than 1000 builds and tests
rhap_5, x86_rhas_3, x86_winnt_5.1, opensuse_11.4-updated, rhap_5.3-updated, deb_5.0, x86_deb_5.0, x86_rhap_5, rhas_3, rhap_6.1-updated, deb_6.0-updated, ubuntu_10.04, rhas_4

29 Platforms with more than 100 builds and tests
rhap_5.2, macos_10.5-updated, opensuse_11.3-updated, rhap_5.3, fedora_14-updated, x86_rhas_4, x86_ubuntu_10.04, freebsd_8.2-updated, sol_5.10, unmanaged-x86_rhap_5, sles_9, sol_5.11, fedora_12-updated, x86_freebsd_7.4-updated, macos_10.6-updated, x86_macos_10.4, macos_10.6, fedora_13-updated, fedora_13, fedora_11, sl_5.5, fedora_12, x86_suse_10.2, ia64_rhas_3, ia64_sles_9, x86_suse_10.0, ubuntu_8.04.3, rhap_6.0-updated, x86_deb_4.0

6 Platforms with more than 10 builds and tests
ppc64_sles_9, ppc_macos_10.4, x86_sles_9, ppc_aix_5.3, sun4u_sol_5.10, x86_macos_10.5-updated

10 Platforms with less than 10 builds and tests
sun4u_sol_5.9, x86_winnt_6.0, ppc_aix_5.2-pl5, ps3_fedora_9, winnt_5.1, x86_deb_6.0-updated, sl_6.0-updated, opensuse_tumbleweed-u, x86_rhap_6.1-updated, fedora_14

Coming “Real Soon Now”™

- New BaTLab.
- Leveraging UCS.
- Evolving to offer other tools.
- Customized result presentation.

“RSN”™: New BaTLab

- Following shift users’ shift in focus to native packaging.
- New hardware, new platforms.
  - Service level designations.
  - Regular, controlled update schedule.
  - Empirical but rigorous package selection, derived by building common software.

“RSN”TM: UCS

- A Cisco product that permits dynamic bare-metal provisioning, on-the-fly creation of virtual lans.
- Will allow us to provide physical hardware for virtual machine testing without compromising our network or statically partitioning it.

“RSN”™: Tool Evolution

- Metronome: continuous integration.
- Koji & Mock: native (RPM) packaging.
  - Use Lab resources without using Metronome.
  - Testing this capability with the VDT.
  - Interface specific to packaging.
  - Can already run as root.

“RSN”™: Custom Dashboards

- Custom dashboard loved by developers
  - Organized by source code branch
  - Visual representation of the continuous tests
  - Don’t even see anything unrelated
- but imposes its own maintenance costs, and can’t be shared with other projects
- Solution: collaborate with Metronome to rewrite and generate a toolkit.

What do YOU want?

Getting Started

- Fill out a form, get an account and get rollin!
  - http://nmi.cs.wisc.edu/
  - Clink “How do I get started -> “Complete this form to request an account”
- Mailing Lists
  - nmi-users@cs.wisc.edu
  - uw-nmi-announce@cs.wisc.edu
- Additional Questions
  - nmi-support@cs.wisc.edu