Design, Status, and Experience with BaBar LTDA

Concetta Cartaro
SLAC
On behalf of
BaBar Computing Group

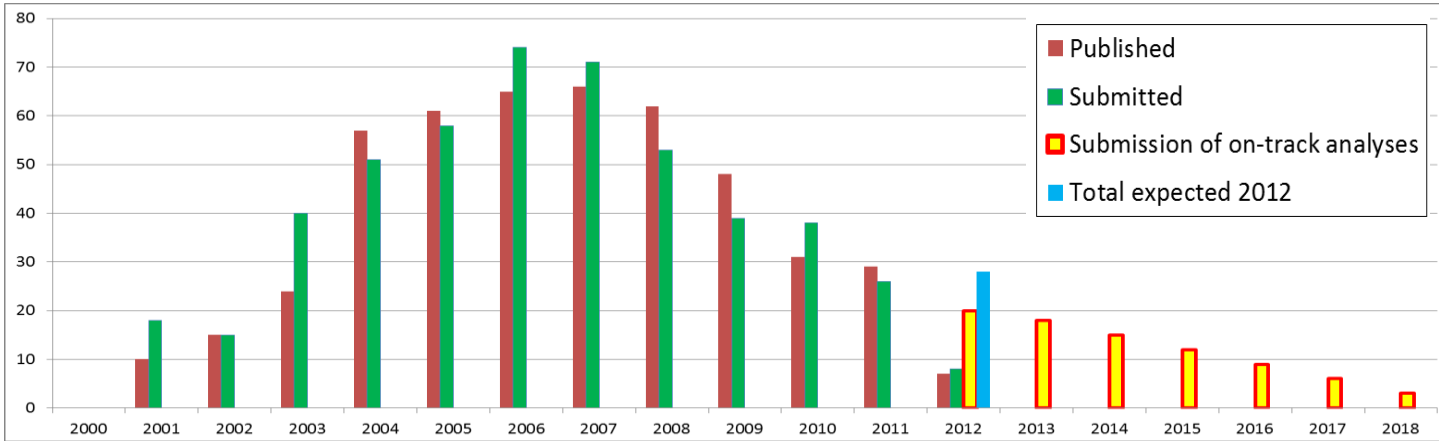
DPHEP @ CHEP
New York, May 24th, 2012

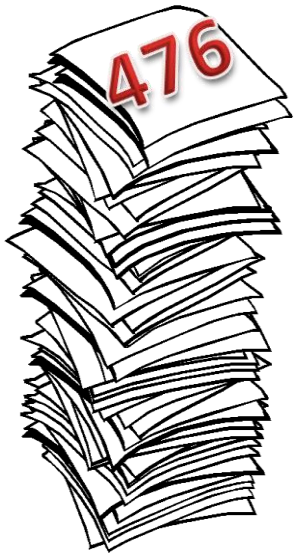
OUTLINE

- BaBar data and choices for the future
- The Long Term Data Access project
- Requirements and design of the LTDA
- The cluster
- Managing the cluster and experience
- Budget, expertise, and long term planning
- Conclusions

BABAR DATA

- BaBar has collected data from Oct 22nd 1999 to Apr 7th 2008
  - 800TB of raw data, 1.2 PB from the last data reprocessing
  - 476published papers to date
  - 84 on track analyses
    - Plus ~30 going forward at a slower pace and about the same number to which we assign a publication probability lower than 50% (generally lacking manpower)
    - Possibilities for new previously unforeseen analyses including discovery analyses
- BaBar (and Belle) data will not be superseded by LHC data
  - Belle II and SuperB will do it in 5-10years
  - Some datasets expected to remain unique for longer (Y(3S) dataset)





BEYOND 2012

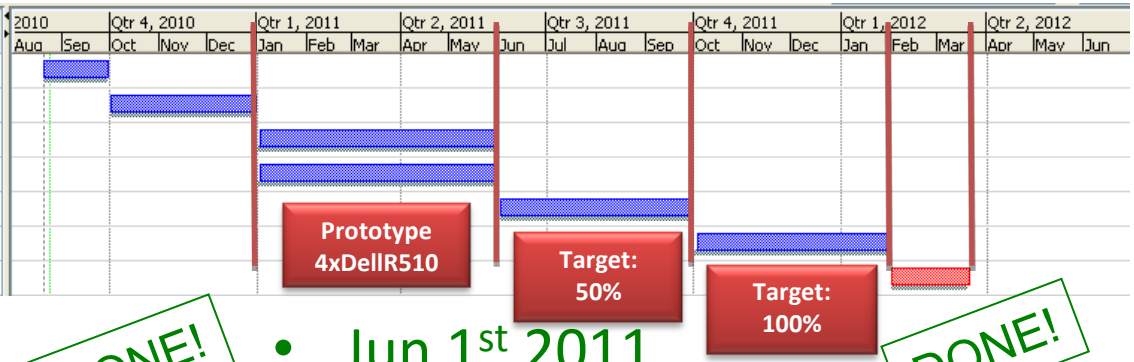
- The BaBar Long Term Data Access project aims to preserve both the data and the ability to support analysis until at least 2018 and will provide support for >50 publications foreseen beyond 2012
- We need to minimize the effort needed to maintain the system
  - Validations, upgrades (OS, tools, …) , documentation, hardware maintenance, etc.
- Close BaBar Framework into a frozen environment
  - Freeze the environment, not the Framework!
  - Simpler to maintain documentation and provide support
- Still not very easy on the long run
  - Hardware support and lifecycle
  - Maintain out of date OS
    - Potential security risks
    - Need to keep know-how about the old OS and the Framework

LONG TERM DATA ACCESS

- Insure the ability to do analysis on the BaBar data beyond 2012 preserving:
  - Data, conditions and calibrations, releases and tools, databases, capability of running production and user jobs
    - This means that in 5 years from now it will be possible, for example, to add a new decay mode, produce the MC events and the relevant skims, and perform a completely new analysis developing new selection code, fitting procedures, etc.
  - Documentation
- Providing a stable environment
  - Last validated OS enclosed in a virtualization layer running the BaBar Framework
- Open formats
  - Data format is based on ROOT which is open and will be part of the system
  - Databases will move away from Oracle and will be stabilized on MySql
  - Code is written in open formats: C/C++, Tcl, Perl, Python.
- Data Storage
  - 2PB will be stored on tape in two Tier A sites (SLAC, CC-IN2P3)
  - Raw data hosted at INFN Padova and CCIN2P3
  - Most used data will sit on disk

MILESTONES

	Name	Start	Finish
1	CEP+PO LTDA Prototype	8/20/10 8:00 AM	9/30/10 5:00 PM
2	Installation/configuration complete	10/1/10 8:00 AM	12/31/10 5:00 PM
3	Run7 available for test users	1/1/11 9:00 AM	6/1/11 5:00 PM
4	Fix, test, tune. PO for final design	1/1/11 9:00 AM	6/1/11 5:00 PM
5	Extended system ready (25% to50%)	6/2/11 8:00 AM	9/30/11 5:00 PM
6	100% system acquired	10/1/11 8:00 AM	1/31/12 5:00 PM
7	Deploy LTDA	2/1/12 9:00 AM	3/22/12 9:00 AM



- Sep 2010
  - PO for the prototype
  - Prototype on site by October
- Dec 2010
  - Installation/configuration complete
- Jan 1st 2011
  - System available for test users
- Jun 1st 2011
  - Test phase ends, LTDA final design ready
  - PO for first 50%
- Jul 1st 2011
  - First 50% of LTDA available
- Oct 1st 2011
  - PO for 100% of LTDA
- March 21st, 2012
  - Deployment of the LTDA
  - All new analyses will use LTDA

DONE!

DONE!

DONE!!

DONE!

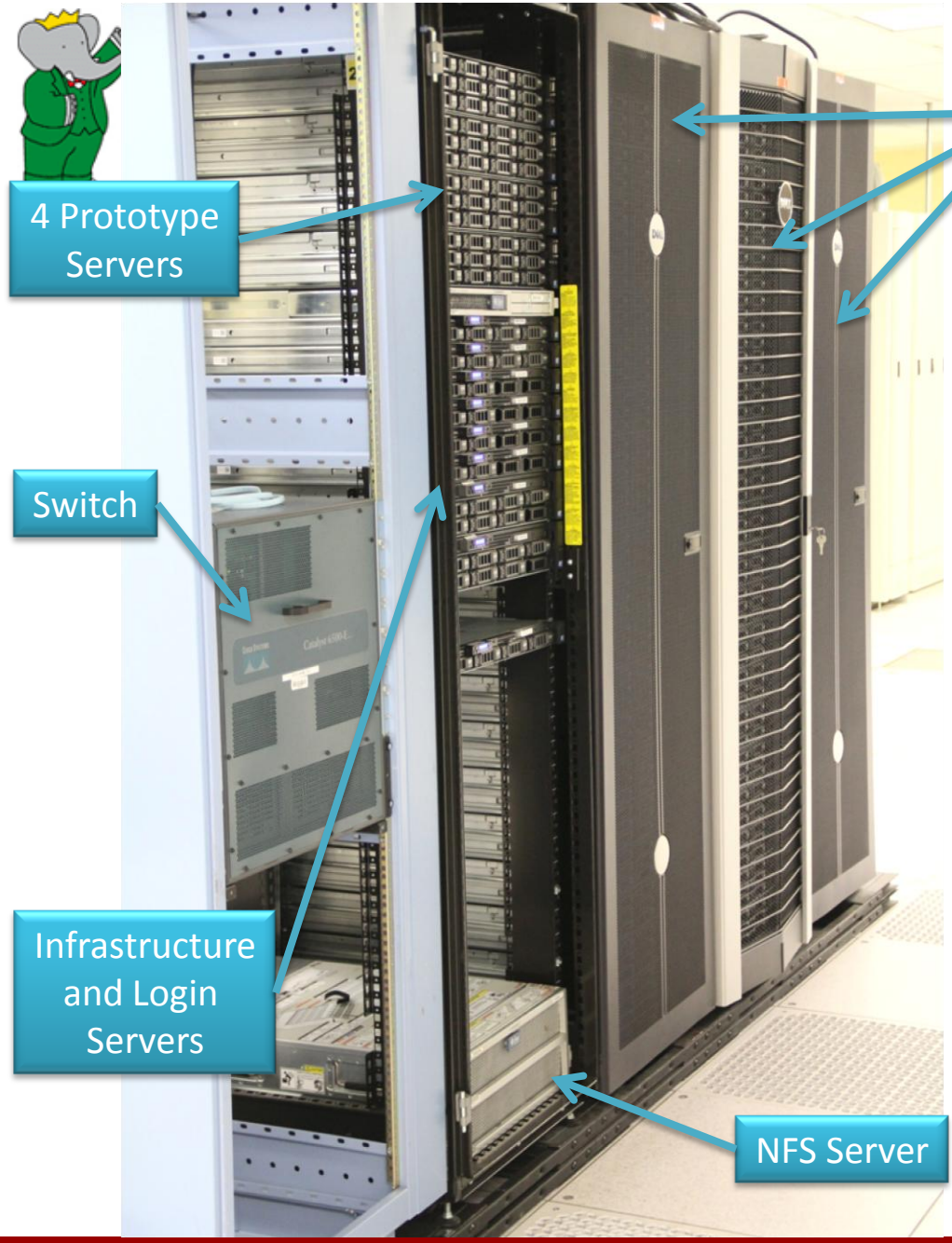
DONE!

DONE!
100% on site

DONE!!!

HARDWARE

- 9 infrastructure servers (Dell R410/R510)
  - 3 front end machines (**`bbrltda`** load balanced pool), 1 cron server, 1 test server, 2 infrastructure servers (network and identification services), 2 database servers (mirrored)
- 54 batch servers
  - Dell R510: Intel dual 6-core, 3GHz, 48GB RAM, 24TB disk
  - 4 of them were the prototype
  - The data is staged on the server disks through XROOTD
- Cisco 6506 network switch with 2x10Gb link card and 192Gb ports
- NFS server
  - Sun X4540 Thor server with 12 cores, 32 GB memory and 32TB storage
  - A second Thor server will be added soon



The whole LTDA Cluster in its final setup in the SLAC computing building

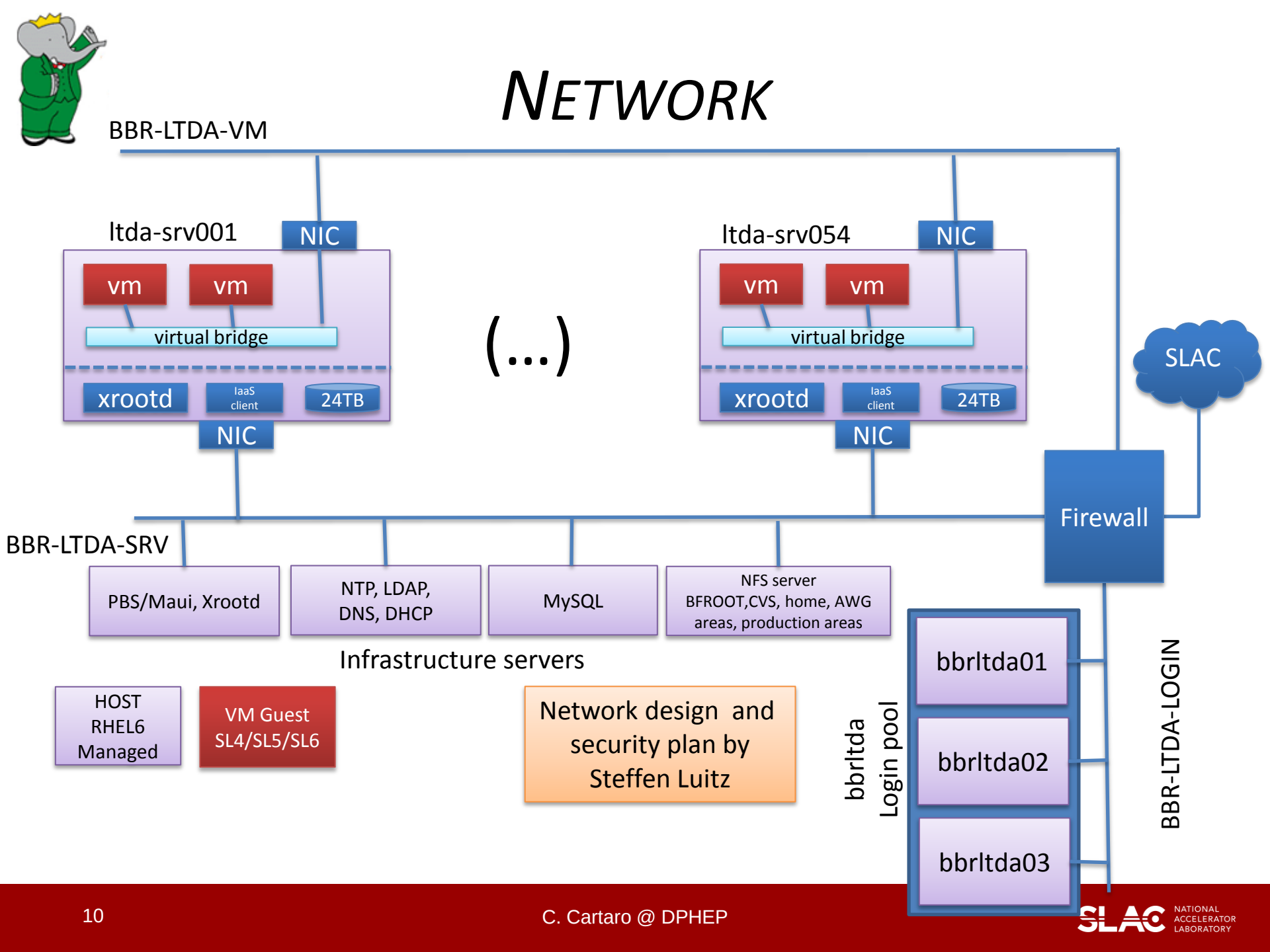
VIRTUALIZATION

- Virtualization can provide virtual hardware support on which we can run virtual images with the wanted OS
  - Hardware support problem solved for the foreseeable future
  - Managing a small number of images is easier than managing a large physical cluster
- However a VM connected to a network is not different from a physical machine and running old OS poses a security threat
  - The dynamic creation and destruction of the VMs adds a small amount of security
  - Images are read-only, qcow2 produces a temporary file with changes to OS and scratch area and it is deleted when the VM's shut down

→ Risk based approach assuming that the VMs are compromised

- Directly affects the network architecture
  - Isolation of back versioned components
  - Physical hosts centrally managed by SLAC CD
  - Firewall rules

NETWORK


BBR-LTDA-VM
ltda-srv001
NIC
vm
vm
virtual bridge
xrootd
IaaS client
24TB
NIC
(...)
ltda-srv054
NIC
vm
vm
virtual bridge
xrootd
IaaS client
24TB
NIC
SLAC
Firewall
BBR-LTDA-SRV
PBS/Maui, Xrootd
NTP, LDAP, DNS, DHCP
MySQL
NFS server BFROOT,CVS, home, AWG areas, production areas
Infrastructure servers
HOST RHEL6 Managed
VM Guest SL4/SL5/SL6
Network design and security plan by Steffen Luitz
bbrltda Login pool
bbrltda01
bbrltda02
bbrltda03
BBR-LTDA-LOGIN

SUBNETS

- Three subnets and one switch to implement the filtering rules between the subnets
  - Login network
    - Open to SLAC network
    - This is as far as the users go
  - Server network
    - File server, DB server, infrastructure (DHCP, DNS, NTP, LDAP)
      - LDAP is a subset of the SLAC Kerberos list mapped on /nfs internal home directories
  - VM network
    - Well defined rules between the VM subnet and the other two

FIREWALL

- VMs are not allowed to connect to SLAC network or the world
- The Login network is protected from the VM network
  - Allow one way ssh from Login to VM network
  - VMs are not allowed to write over the Login network
- Well defined services between VM network and SRV network
  - Infrastructure (DNS, LDAP, NTP), file service (Xrootd, nfs), batch scheduling
- Allow SRV and Login networks use SLAC infrastructure

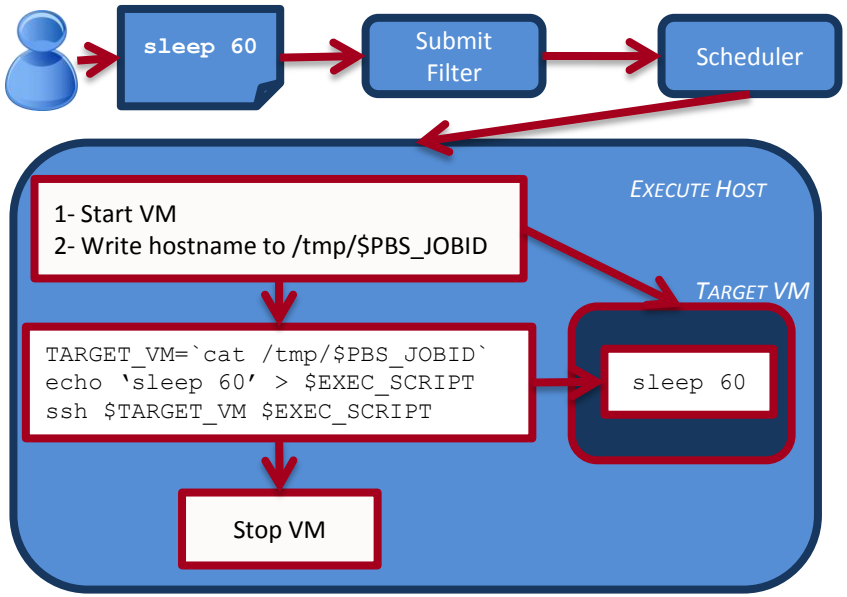
From / To	SLAC	BBR-LTDA-LOGIN	BBR-LTDA-SRV	BBR-LTDA-VM
SLAC	1	ssh, infrastructure	Infrastructure, tape access, backup	X
BBR-LTDA-LOGIN	infrastructure	1	ssh, NFS, infrastructure	ssh (for interactive VM)
BBR-LTDA-SRV	infrastructure, tape access, backup	NFS	1	LRM, DHCP, xroot, NFS, LDAP
BBR-LTDA-VM	X	X	LRM, DHCP, xroot, NFS	1

LTDA Cloud

- PBS/Torque is used to manage the batch resources and Maui is the batch scheduler
  - Moved away from Condor and Nimbus due to their instabilities
- The virtualization layer uses qemu with kvm support directly
  - Moved away from libvirt due to instability
- Resources
  - Each batch server has 12 physical cores of which one is dedicated to the host itself and the XROOTD service. The other 11 cores are used to run virtual machines.
  - Each batch server has 12x2TB disks of which 11 are dedicated to XROOTD (no raid – all data is recoverable from tape) and one disk is dedicated to use for the copy-on-write VM images (OS and scratch)
- Hyper-threading
  - Performance tests (→ few slides ahead)

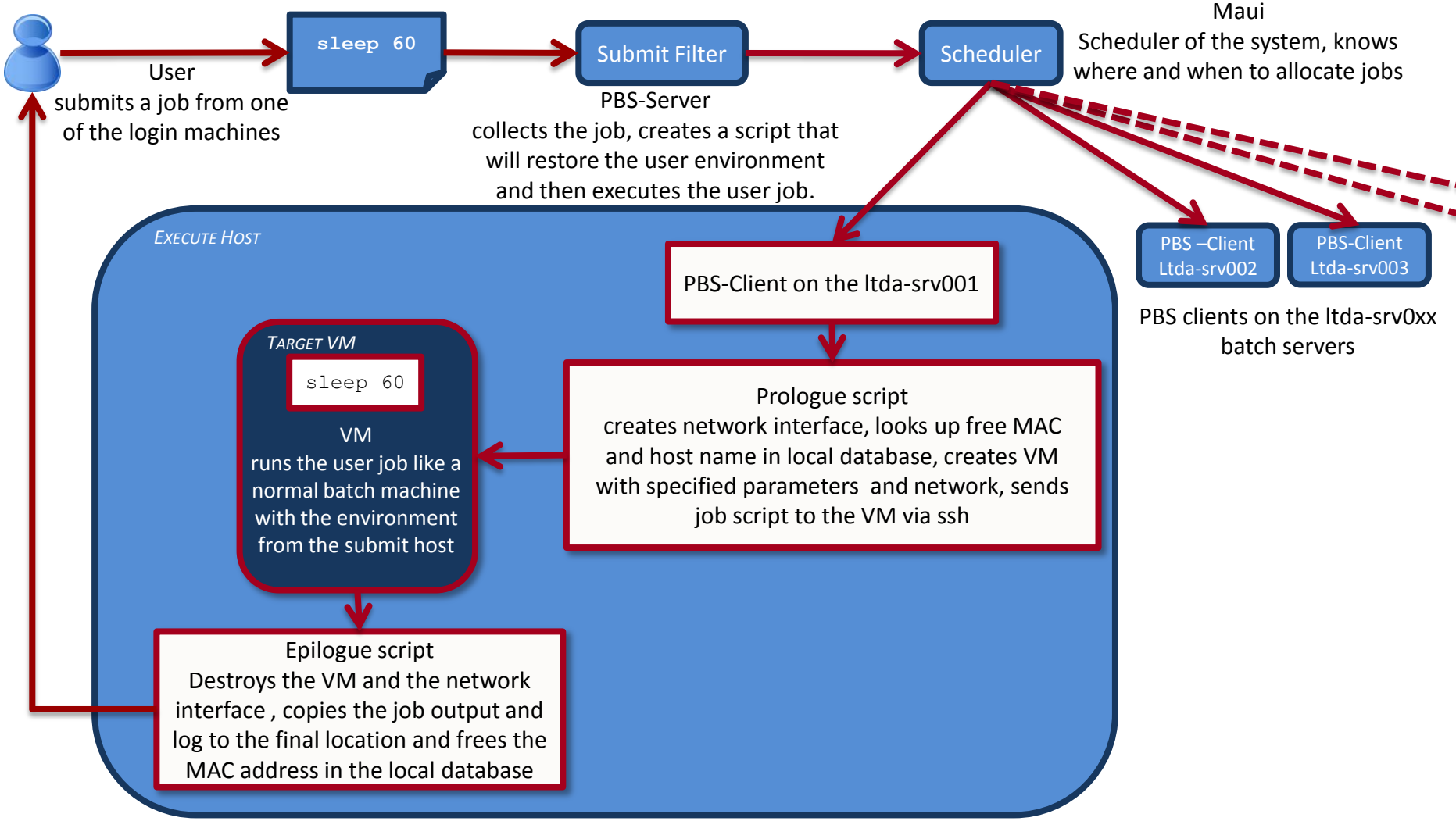
JOB SUBMISSION

- Qemu is called directly
  - Libvirt removed
- Need to create the network interface for the VMs
  - 24 MAC addresses and usage status stored in local db
- PBS Prologues and Epilogues scripts are used to create and destroy the VM's and the needed network environment



Marcus Ebert
Kyle Fransham

A Little More Detail ...



User Perspective

- Submit a job
- Wait for the result or enter interactive VM
- Limitations are due to the restrictions on the VMs
  - The code cannot be edited from a VM if it resides in the home directory
  - Any change done to a VM will disappear with the VM
    - Changes go to a temporary copy-on-write image that is removed when the VM is destroyed
  - The VM environment is limited and designed to do only one thing: run and debug BaBar code

PERFORMANCE TESTS

- VM's vs SLAC batch queue
- Hyper-threading on/off
- I/O performance tests

VM vs Bare Metal

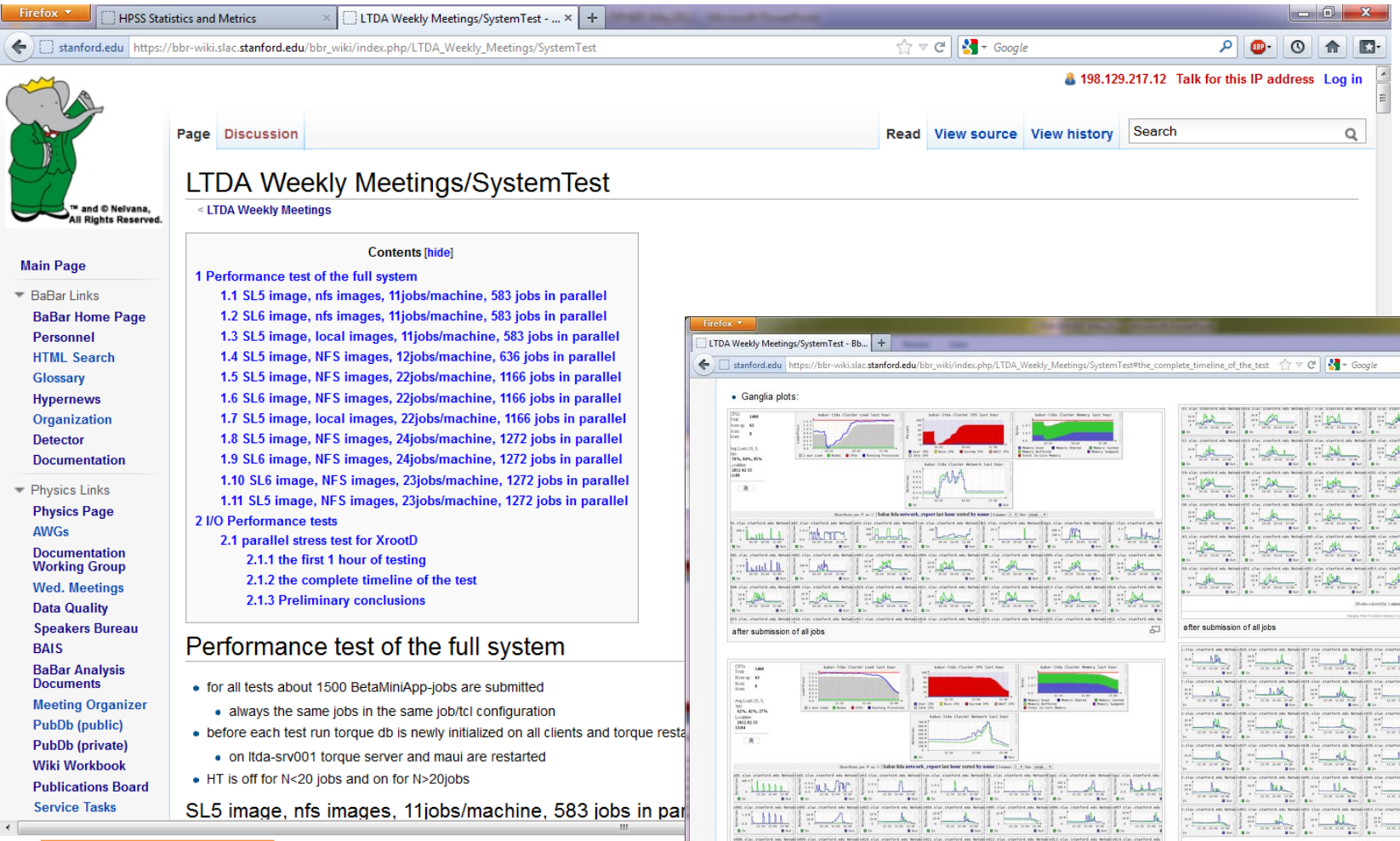
Name of the output file	USER CPU time on the LTDA	VM name on the LTDA	USER CPU time on the normal system	Host name on the normal system(CPU type)	Differences in used time	percentages of cpu time used more on LTDA
LambdaC-Run1-OnPeak-R24c-1.out	5521	bbr-ltda-vm049	4212	hequ0167 (Intel(R) Xeon(R) CPU X5570 @ 2.93GHz)	1309	31.07
LambdaC-Run1-OnPeak-R24c-10.out	5572	bbr-ltda-vm050	6765	fell0171 (Intel(R) Xeon(R) CPU X5355 @ 2.66GHz)	-1193	-17.63

~1500 jobs later …

LambdaC-Run6-OnPeak-R24c-99.out	5274	bbr-ltda-vm037	6862	fell0249 (Intel(R) Xeon(R) CPU X5355 @ 2.66GHz)	-1588	-23.14
all	9019444		9272133		-252689	-2.72

LTDA host machines:
Intel(R) Xeon(R) CPU X5670 @ 2.93GHz (prototype machines)
Rhel5 / Hyper-threading off

HYPER-THREADING TESTS



Marcus Ebert

COMPARISON TABLES

- CPU intensive jobs show no difference when single job or 11 jobs run on a machine
  - but variation for repeated tests is up to 30%
- I/O intensive jobs can be up to 300% slower when 11 jobs run in parallel on a single machine compared to a single job/machine
- BetaMiniApp is about 20% slower when running 11 jobs in parallel on a single machine compared to a single job/machine
- CPU time used for all Run1-6 jobs when running 11jobs/LTDA machine in parallel is comparable to running same jobs on the general SLAC queue
  - LTDA used about 2% less CPU time for all jobs
- CPU time used for a BetaMiniApp job is comparable to HT off when running same number of jobs in parallel
- single BetaMiniApp can use up to 50% more CPU time when running 22jobs in parallel instead of 11
- CPU intensive again independent of the number of parallel jobs
  - <10% slower than HT off
- I/O intensive jobs can be up to 900% slower when running 22jobs/machine compared to a single job

HT off vs HT on

	11 jobs SL5 NFS image	**22 jobs SL5 NFS image**	difference
CPU time	7849934	11243061	+40%
Wall time	8697739	11996994	+38%
	11 jobs SL6 NFS image	**22 jobs SL6 NFS image**	difference
CPU time	5152806	7286484	+41%
Wall time	7245687	10267120	+42%

SL5 vs. SL6

	11 jobs SL5 NFS image	**11 jobs SL6 NFS image**	difference
CPU time	7829719	5120721	-35%
Wall time	8371660	7200235	-14%
	22 jobs SL5 NFS image	**22 jobs SL6 NFS image**	difference
CPU time	11233998	7249142	-35%
Wall time	11987646	10214567	-15%

RESULTS

True in general, not LTDA specific

Firefox ▾ | LTDA Weekly Meetings/SystemTest/resu... | stanford.edu https://bbr-wiki.slac.stanford.edu/bbr_wiki/index.php/LTDA_Weekly_Meetings/SystemTest/results | Google

conclusion

- CPU intensive jobs show no dependency on the number of parallel jobs
- I/O intensive jobs depend heavily on the number of parallel jobs
  - that's expected
- for BetaMiniApp jobs:
  - running the same number of jobs in parallel shows no difference between HT off and HT on
  - running 11jobs/machine shows no difference to the general SLAC queue
  - running 22 jobs/machine slows down single jobs up to 50% compared to 11jobs/machine
  - the difference for CPU/Wall time between usage of NFS or local images is only very small
    - for using NFS images the network load on wain062 is higher
    - but no problem so far since it has a 4G etherchannel and peak value was around 3.2G
    - could become a problem when adding more servers
  - running the same binary on SL6 instead of SL5 reduces CPU time by about 35%
  - using all cores for VM's slows down single jobs compared to 11(HT off) or 22(HT on) jobs/machine
  - time to finish all Run1-6 jobs using 22jobs/machine is about 15% shorter than for 11jobs/machine
    - this number depends heavy on the number of jobs
    - for no HT ~1500 jobs mean ~2times full load while for HT on it's only 1x full load (+ running only some jobs/machine for both)
    - time difference will be much smaller if one uses only 500 jobs but with much more events processing/job
    - time difference will be much larger if one uses 3000 jobs but with much less events processing/job
    - difference can be between 0% and ~30%

proposal for the final system

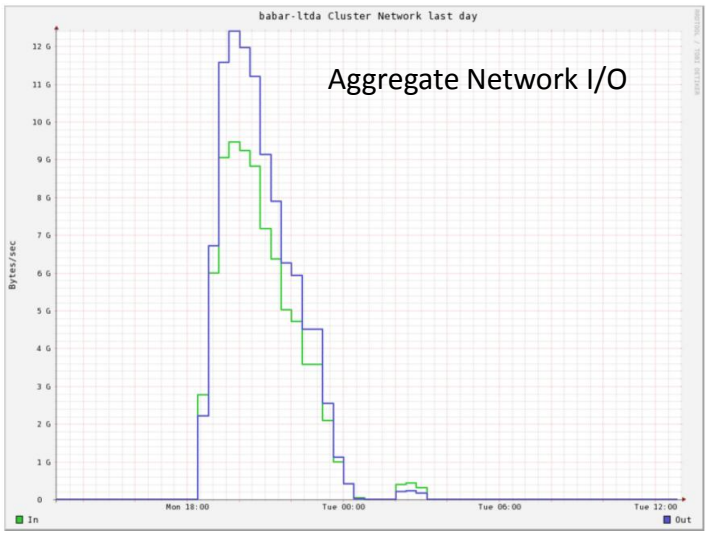
- use HT on
- allow 22jobs/machine
- use NFS images
  - switch to local images could easily be done if we see problems with more servers
- switch to a release which can be build and run on SL6
- reduce the wall time for the general queue again to let not run jobs with too many events processing
- repeat some tests once RHEL6.x is installed on all machines

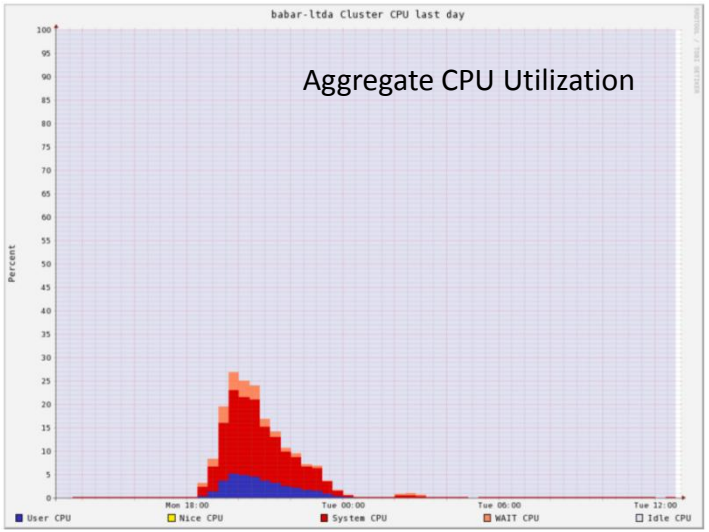
I/O Performance test For XrootD

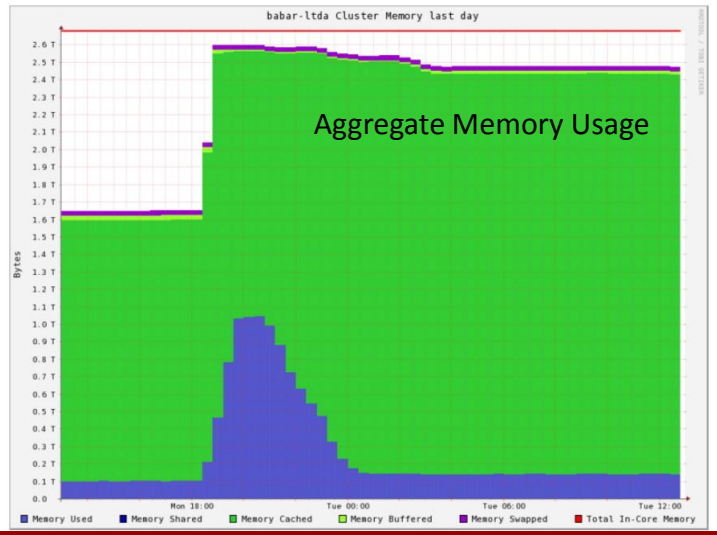
- The main goal of the test was to see how much data can be delivered under extreme load by LTDA XrootD installation to clients processes. Resource (memory and CPU) utilization was also being monitored during the tests using Ganglia. Scalability of the XrootD installation has been tested as well.

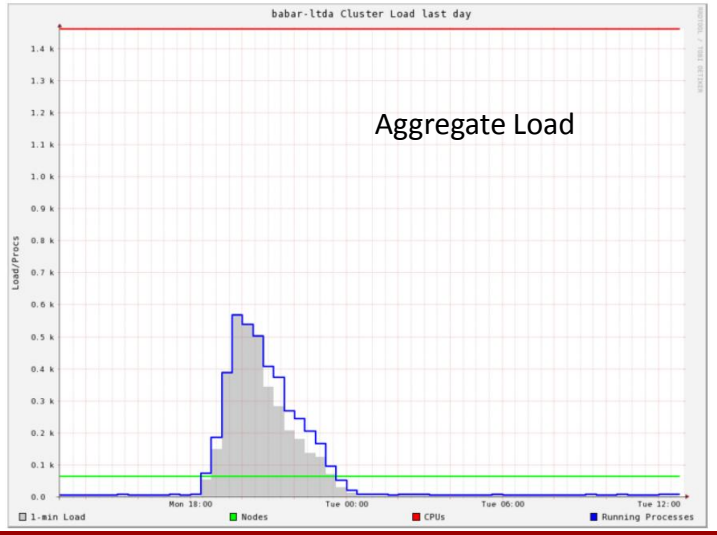
Igor Gaponenko

MONITORING THE TEST





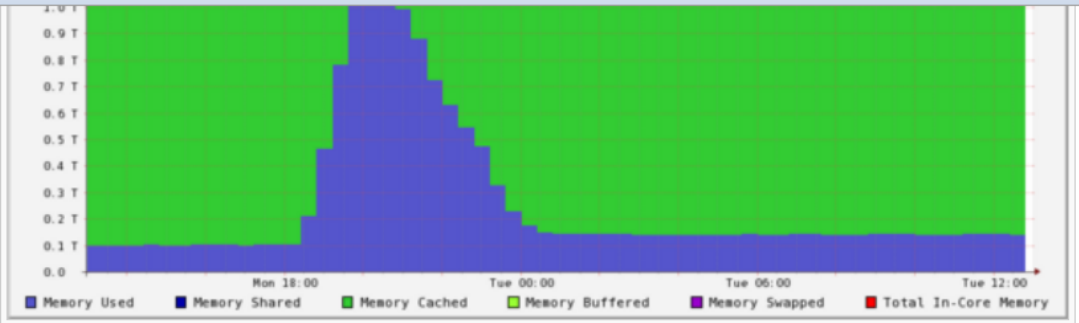




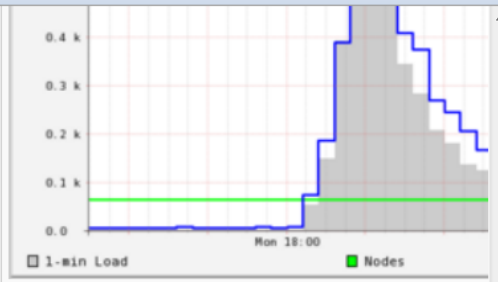
RESULTS

Firefox — LTDA Weekly Meetings/SystemTest - Bb...

https://bbr-wiki.slac.stanford.edu/bbr_wiki/index.php/LTDA_Weekly_Meetings/SystemTest#the_complete_timeline_of_the_test



Aggregate Memory Usage



Aggregate Load

Preliminary conclusions

- the architecture of the cluster is capable of delivering up to 12 GB/s of data to client applications
- moderate resource (CPU and memory) utilization under extreme I/O load leaves more than enough room for client job to perform any useful processing on the data
- the I/O capability of the cluster far exceeded any possible demand from known BABAR applications run (or to be run) on the Cluster
- hence, the cluster can be easily expanded with storage-less (pure) compute nodes should this be needed by the BABAR experiment
- actual limits of the expansion can be easily drawn by cross-correlating performance numbers of this test against I/O requirements of the applications. For instance, based on prior tests run on the cluster it's probably safe to estimate at least 5 times greater number of compute nodes in the Cluster as compared with its current configuration.

This page was last modified on 28 February 2012, at 14:08.

This page has been accessed 212 times.

Privacy policy About Bbr_wiki Disclaimers

Powered By MediaWiki

Some Problems & Solutions

- **qemu couldn't be started**
  - error message: kvm_create_vm: Interrupted system call
  - for about 0.5% of all jobs
  - jobs listed in the queue until wall time is over
  - known bug of kvm/qemu
  - should be solved in new versions
  - expect this problem to be gone on RHEL6.2
- **maui often shuts down without any hint in the log file**
  - seems to happen always with a high load on the scheduler and the network and more than 600 jobs

→ solution:

  - don't allow torque to push jobs to maui
  - only Maui looks every 10s for new jobs
  - for one schedule cycle only 10jobs/user are considered
  - if there are free nodes and waiting jobs, then let maui wait 4s between sending jobs to the nodes
  - users could also put in their scripts a delay of 1s between submission of jobs

- **very high network usage on some server**
  - due to the loading of one condition file in cond24boot09
  - not seen for cond24boot11

→ solution:

  - reduplicate conditions on more servers

- **input collections or conditions couldn't be found**
  - to many open connections in xrootd
  - Network problems on the xrootd client hosts
  - connection couldn't be established
  - wrong mounted hard disk

→ solution:

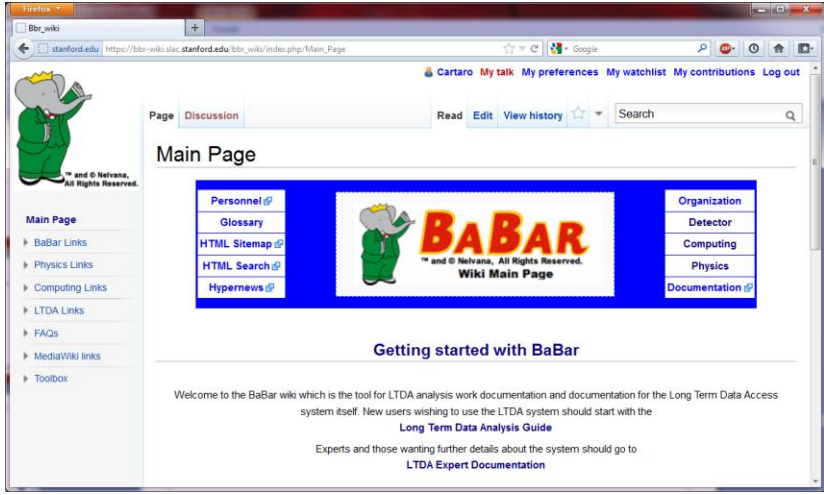
  - correctly mount the hard disk on ltda-srv005
  - reduplicate the conditions on many servers to reduce the load on a single one
  - tune the tcp parameters on all ltda-srv0xx
  - use a timeout in xrood for the connections to the clients

- **NFS server stopped to give new nfs exports out**
  - after some runs with more than 1000VM in parallel no new nfs mounts have been possible
    - this includes the home mount using automounter on the login machines for new logins
    - all existing mounts still worked
  - seems like a limit in the nfs server, maybe in open network ports, was reached

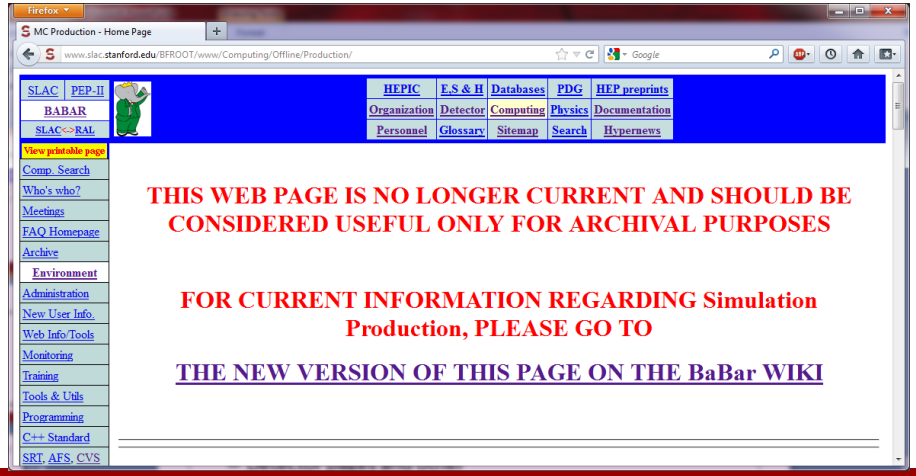
→ solution:

  - unix-admin changed some nfs related settings, we will see in the future if it's enough

DOCUMENTATION



- Strong push toward documentation clean up, ease of access, and clarity
- All most used and fundamental info are being checked, updated and moved to a Media Wiki server, the *BABAR WIKI*
  - Old pages clearly marked but kept online for archival purposes
  - Detector pages and other pages that will supposedly never change again will be left in their original location



Documentation Working Group

- The effort needed is not trivial
- Created a Documentation Working Group to coordinate the migration effort aided by an advisory committee
  - Many new students joined the effort but the input from senior members of the Collaboration is fundamental
  - There are 10 official members in the DWG but we promote the migration to the wiki as a Collaboration effort
  - Roadmap and completion deadlines being established
  - Experts will sign-off on the content of migrated pages

Matt Bellis, DWG Coordinator

NEW BABAR PUBLIC PAGE

http://www-public.slac.stanford.edu/babar/

Firefox | Documentation Working Group - Bbr... | BaBar Experiment Public Web Site

BABAR

SLAC NATIONAL ACCELERATOR LABORATORY

- Home
- Purpose of BABAR
- BABAR Physics
- How BABAR Works
- BABAR Publications
- All BABAR News
- Images, Video, & More
- Organization
- SLAC Home
- BABAR Internal Page
- Site Index

The BaBar Experiment

Welcome to the BABAR public web site. BABAR is a particle physics experiment designed to study some of the most fundamental questions about the universe by exploring its basic constituents - elementary particles. The BABAR Collaboration's research topics include the nature of antimatter, the properties and interactions of the particles known as quarks and leptons, and searches for new physics. We invite you to explore the site and learn about the BABAR detector, our research, and the physicists who perform it.

Recent News

- **Particle-Physics History on Display**
  May 9, 2012
  BABAR's innermost system, the Silicon Vertex Tracker, gets a new life as a unique exhibit of particle-detector design and construction.
- **Hunting Dark Photons and Higgs with BABAR: Science Highlight and Symmetry Magazine Article**
  May 2012
  Light dark photons? Dark Higgs bosons? Scientists look for signs of these weird-sounding particles in data from BaBar—an experiment designed to explain a completely different mystery.
- **BABAR Members Greeted Warmly at Wintery Conference**

collaboration braved frigid temperatures to present the

BABAR Highlights

- **BABAR's role in the 2008 Nobel Prize in Physics, December 8, 2008.** Matter-Antimatter Asymmetry Measurements Put Final Seal of Approval on the Theory of Quarks.
- **BABAR Discovers the Bottom-Most Bottomonium, July 9, 2008.** The "ground state" of $b\bar{b}$ quark pairs is finally detected.
- **New form of matter-antimatter transformation observed for the first time, March 13, 2007.** BABAR finds evidence for mixing in the charm system.
- **BABAR discovers a new massive particle, July 1, 2005.** The Y(4260) is not expected theoretically and its nature is a mystery.
- **BABAR sees direct charge-parity violation, August 2, 2004.** B mesons and their antiparticle partners undergo a radioactive decay at different rates.
- **BABAR announces new result on charge-parity violation, July 23, 2002.** Precise measurement of the parameter sin2β.
- **BABAR establishes charge-parity**

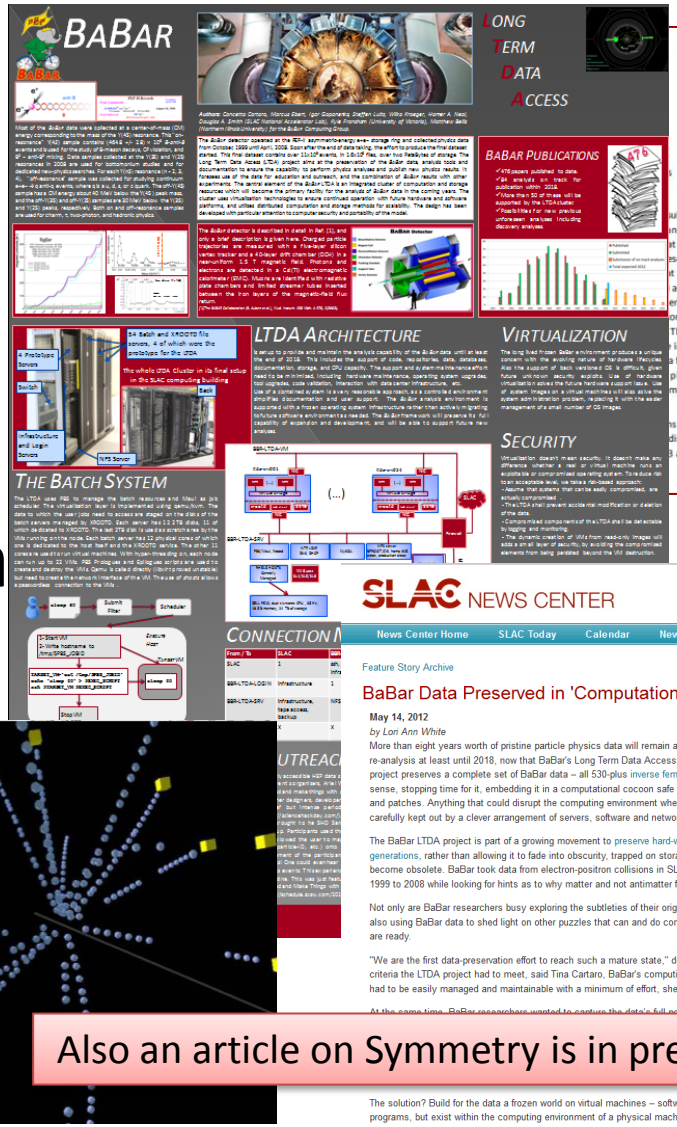
Abi Soffer, BaBar PAC & DWG Member

MANPOWER, EXPERTISE & BUDGET

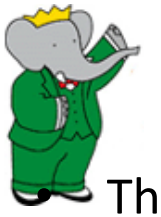
- Designing and maintaining something like the LTDA through the years requires many talents and careful planning
  - *BABAR* experts
    - Releases, databases, data management and documentation
    - The Collaboration will have to provide such expertise
  - Computing experts
    - Network architects, security, system and networks administration, virtualization, …
    - 0.5 FTE/year foreseen after 2012
- Costs (not FTEs)
  - Hardware and refreshment program
  - Recharge (somewhat unknown) costs
  - Red Hat entitlements for virtualization
    - Use SL virtual machines to avoid extra cost (SL is not supported at SLAC)

LTDA Beyond the Cluster

- CHEP
- DPHEP
- Documentation
- Duplication of data
- Portability
  - BaBar ToGo …
- Outreach
- Visibility



Also an article on Symmetry is in preparation….



ACKNOWLEDGMENT PAGE

Thanks to the LTDA developers

- Coordinator: Tina Cartaro
- BaBar software expert: Homer Neal
- Development and system administration of LTDA: Marcus Ebert
- Network design: Steffen Luitz
- Virtualization expert: Kyle Fransham
- System performance and CDB: Igor Gaponenko
- Databases, tools and production: Douglas Smith and Tim Adye
- Computing Division experts
  - System setup and adminstration: Booker Bense, Lance Nakata, Randall Radmer and all the Unix-Admin team
  - Xrootd expert: Wilko Kroeger
  - Network setup: Antonio Ceseracciu
  - BaBar-SLAC liaison: Len Moss
- Thanks to the PPA Management and the DOE for the strong support
- Thanks to the Advisory committee for the precious discussions and suggestions
  - Justin Albert (BaBar), Fabrizio Bianchi (CSC/BaBar, Committee Chair), Cristi Diaconu (Marseille University and CPPM), Richard Mount (SLAC/ATLAS), Jean-Yves Nief (CCIN2P3)

CONCLUSION

- System went into production on March 21st
  - On time and within budget
- There are more than 40 users of the system
  - 10 very active
  - 354486 jobs have run already this year by the users!
- In use for production
  - Behaves like a Tier site
- Currently the system batch capacity comes from 54 12-core systems with doubling thanks to hyper-threading (up to 1296 possible slots) with 24 TB of storage on each server
- Extension of system already planned and approved for accelerated migration of analysis load from general queues to the LTDA
  - 20 systems with equivalent processing capacity but no xrootd storage will be added for a total of 1776 possible slots (using all cores).

Q & A

- Why LTDA and not the GRID or the commercial clouds?
  - Code and data is directly accessible from the common NFS area and don't need to be included in the VM image thus allowing memory and startup efficient VM's and easy use of new releases and access to a global CVS repository.
  - Files are directly accessible from the NFS areas instead of needing to copy them to the VM space. A job has the same local read access privileges as those of the user.
  - Many research computing clusters using virtualization allow only certain directories to be mounted for importing files thus resulting in extra effort from the user to create a setup appropriate for batch submission which is often different from the development setup.
  - One can log into the VM that the job is running on and diagnosis resource usage problems.
  - GRID systems are frequently very difficult to debug because one can not directly observe the job as it is processing.
  - One can start interactive VM sessions for development/debugging work.
  - Users will not have to invest time and effort to acquire funds for using a commercial cloud system.
  - A platform equivalent to that for the batch jobs will always be available for development and debugging purposes.
  - The setup allows users to migrate seemlessly to the LTDA system with only a few minor restrictions on what one can do.
- Why doesn't one just use the standard batch system but with VM's?
  - The whole SLAC batch system would have to be adjusted to be behind a firewall to protect against use of insecure platforms and other projects running behind the firewall could be affected.
  - New tools and technologies may allow it in the future
- The lifetimes of the latest RHEL releases have been extended, doesn't it remove the need for the LTDA?
  - No because the expertise and person power will cease to exist to do full release validations and possible update of the hundreds of packages of code every time a new security patch (typically coming along with a new version of glibc) is released.
- What about dependencies on the virtualization system?
  - Look for alternatives (xen, …) and ultimately use hardware emulation