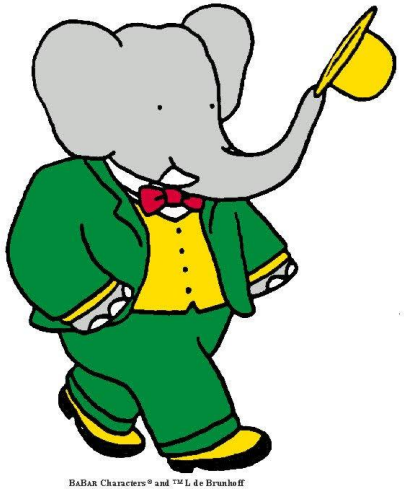
Data Preservation

BaBar's experience



Marcus Ebert
BaBar Computing Coordinator

BaBar timeline

- BaBar stopped data taking in 2008
- manpower within BaBar going down since ~2010
  - data preservation should have started by then
  - only framework freeze in 2012
- since 2019: minimal support for BaBar infrastructure at SLAC
  - preservation efforts started
- beginning of 2021: support for infrastructure at SLAC stopped
  - everything needed to be moved away from SLAC and can’t depend on SLAC infrastructure anymore

BaBar timeline

- BaBar stopped data taking in 2008
- manpower within BaBar going down since ~2010
  - data preservation should have started by then
  - only framework freeze in 2012
- since 2019: minimal support for BaBar infrastructure at SLAC
  - preservation efforts started
- beginning of 2021: support for infrastructure at SLAC stopped
  - everything needed to be moved away from SLAC and can't depend on SLAC infrastructure anymore

Should have started much earlier with data preservation.

Data preservation

Data preservation

Data:

All information created as a result of planning, running and exploiting of an experiment - especially not just the “physics data” collected by an experiment

Preservation:

Not just storing the data safely, but keeping the data accessible and usable to extract new results.

(see DPHEP Global Report 2022)

What should be preserved?

What should be preserved?

- Physics data
  - collected Data, MC events (signal and generic)
  - Metadata
- Analysis framework
  - analysis software, OS, specific version of tools
    - perl, python, xrootd, root libs,...
- **Documentation to make use of the analysis framework**
  - software description
  - data available and how to access (skims, produced MC samples,...)
- Collaboration tools
  - **discussion places**, email list, meeting software, member database,...

Physics data

- all data in root files
  - data files and conditions db
  - data access by streaming via xrootd

- metadata in mysql
  - events in which root files are in which collections/skims/run periods/...
  - bad events not to use for analyses
  - file checksums,...

Physics data preservation

data preservation: copy all files to a new long term destinations for active usage and backup, verify checksum, and mark as available in the db (backup db too)

Challenge: Find a site/place that can host the data and give access to it, longterm.

- site management/policy/politics can change O(10years)
- data hosting/access costs

Physics data preservation

data preservation: copy all files to a new long term destinations for active usage and backup, verify checksum, and mark as available in the db (backup db too)

- GridKa offered to store data and MC files from the latest processing run (AllEvents, skims, conditions db,...) for active usage

- GridKa also continues to host mysqld with the metadata db
  - xrootd and mysqld at Gridka managed by BaBar

- IN2P3 hosts since a long time a second copy of all BaBar data, incl. raw data, as backup (not for active usage) and agreed to continue to do that
- CERN offered via DPHEP/Open Data Portal to also host a copy of all data

Physics data preservation

data preservation: copy all files to a new long term destinations for active usage and backup, verify checksum, and mark as available in the db (backup db too)

Issues: Some tools (MC mode lookup, luminosity calculations,...) depend on SLAC ORACLE database, usage hardcoded in tools, also true for other db like publications db

- only found when access to Oracle was changed at SLAC
- difficult to find which databases exactly where used
  - getting dump took time
  - efforts to use in mariadb still ongoing
    - make data compatible
    - change tools

Analysis framework

- BaBar software 32bit, users usually write C++ code and compile their analysis modules
  - does not compile on 64bit
- depends on older software releases, e.g. perl, xrootd,...
  - latest verified system: SL6.3, gcc 4.4.x, kernel 2.6,...

- early preservation efforts: BaBar built an own cloud in 2012
  - software framework and OS frozen, no changes allowed
  - VM based on verified OS and software
  - software framework in well defined directory structure, mounted in VM

Using VMs and a well defined structure for the software frameworks made it relatively easy to preserve possibility to do analyses.

Analysis framework preservation

framework preservation: archive and copy whole directory tree together with VM images to long term storage and make it usable from there

Analysis framework preservation

framework preservation: archive and copy whole directory tree together with VM images to long term storage and make it usable from there

- easy task (copied all over to UVic which hosts new analysis system)
- but details are not...
  - some links, hardcoded path in source code, scripts,... not using relative paths but absolute ones
    - mount NFS under the same structure as it was, using /afs/slac....
  - some even point to user directories ($HOME, testing areas,...)
    - production tasks run by single users long term on their own accounts, not on general production accounts
    - users tasked with patching did so in their own area, and compiled/linked from there - dynamically linked so libraries still expected to be in those testing areas still
  - issues only found when things broke after moving the framework out of its initial environment

Analysis framework preservation

framework preservation: archive and copy whole directory tree together with VM images to long term storage and make it usable from there

- easy task (copied all over to UVic which hosts new analysis system)
- but details are not...
  - some links, hardcoded path in source code, scripts,... not using relative paths but absolute ones
    - mount NFS under the same structure as it was, using /afs/slac....
  - some even point to user directories ($HOME, testing areas,...)
    - production tasks run by single users long term on their own accounts, not on general production accounts
    - users tasked with patching did so in their own area, and compiled/linked from there - dynamically linked so libraries still expected to be in those testing areas still
  - issues only found when things broke after moving the framework out of its initial environment

Do not run production tasks on personal accounts!

Analysis framework preservation

In addition:
Since BaBar uses VMs already, create a VM for users to run on any system.

Two images: OS as used on the central analysis system, second image contains software framework (latest release only, same structure as in the central system).

Users can write, compile, and test their code, and run over any data/MC files when internet connection is available on any machine that supports VMs, including their own laptop.

Using batch systems

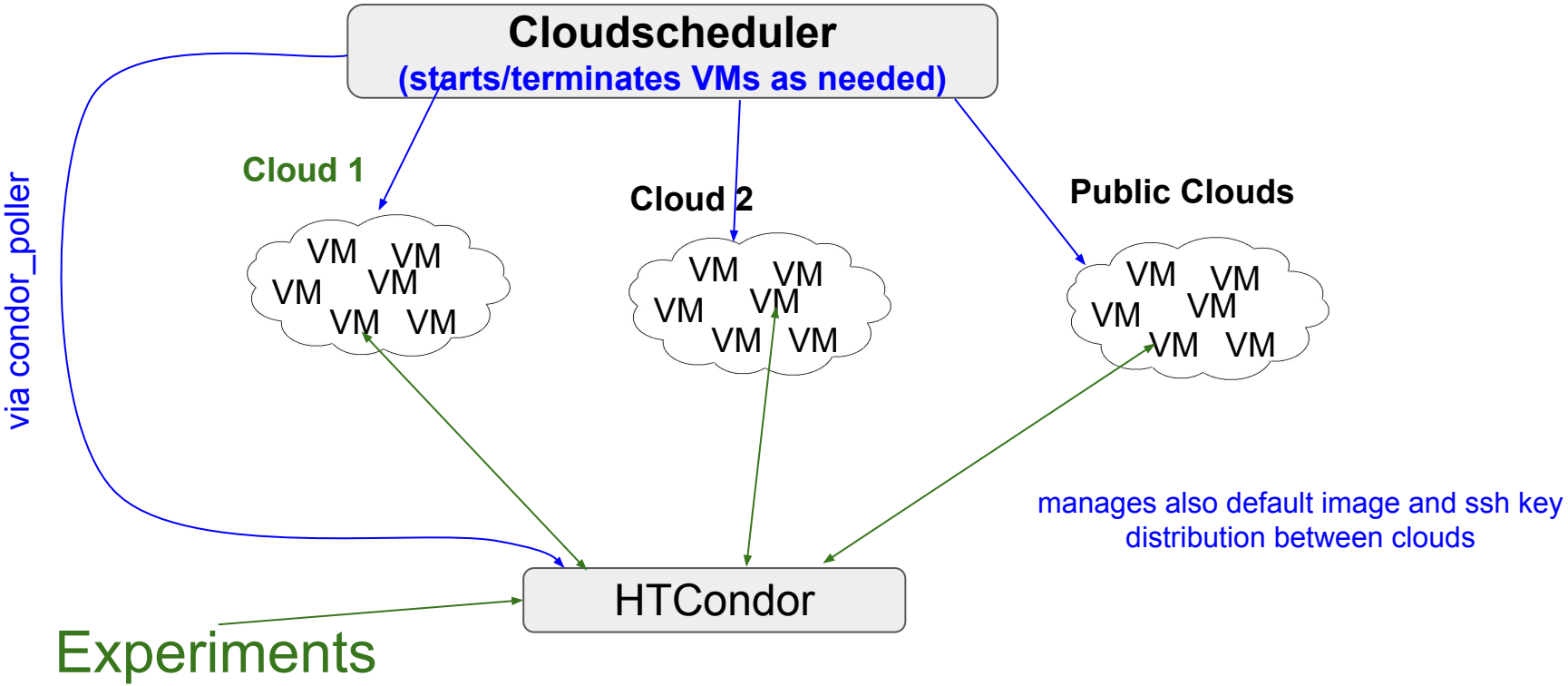
Environment frozen -> VMs need to be used

UVic: Cloudscheduler to run HEP jobs in VMs on clouds (https://csv2.heprc.uvic.ca/public/)

Using batch systems

Environment frozen -> VMs need to be used

UVic: Cloudscheduler to run HEP jobs in VMs on clouds (https://csv2.heprc.uvic.ca/public/)



Using batch systems

Environment frozen -> VMs need to be used
UVic: Cloudscheduler to run HEP jobs in VMs on clouds (https://csv2.heprc.uvic.ca/public/)

Previous batch systems: LSF (also assumed by analysis framework)
UVic batch system: HTCondor

---> wrapper scripts needed to be written to translate old commands and option to HTCondor commands
----> output of HTCondor commands need to be formatted as in the old commands

Very time consuming effort...

Analysis preservation

No analysis preservation done...

- analysis code depends on specific releases, which depend on specific OS,...
- improvements in new releases (tracking, PID, ...)

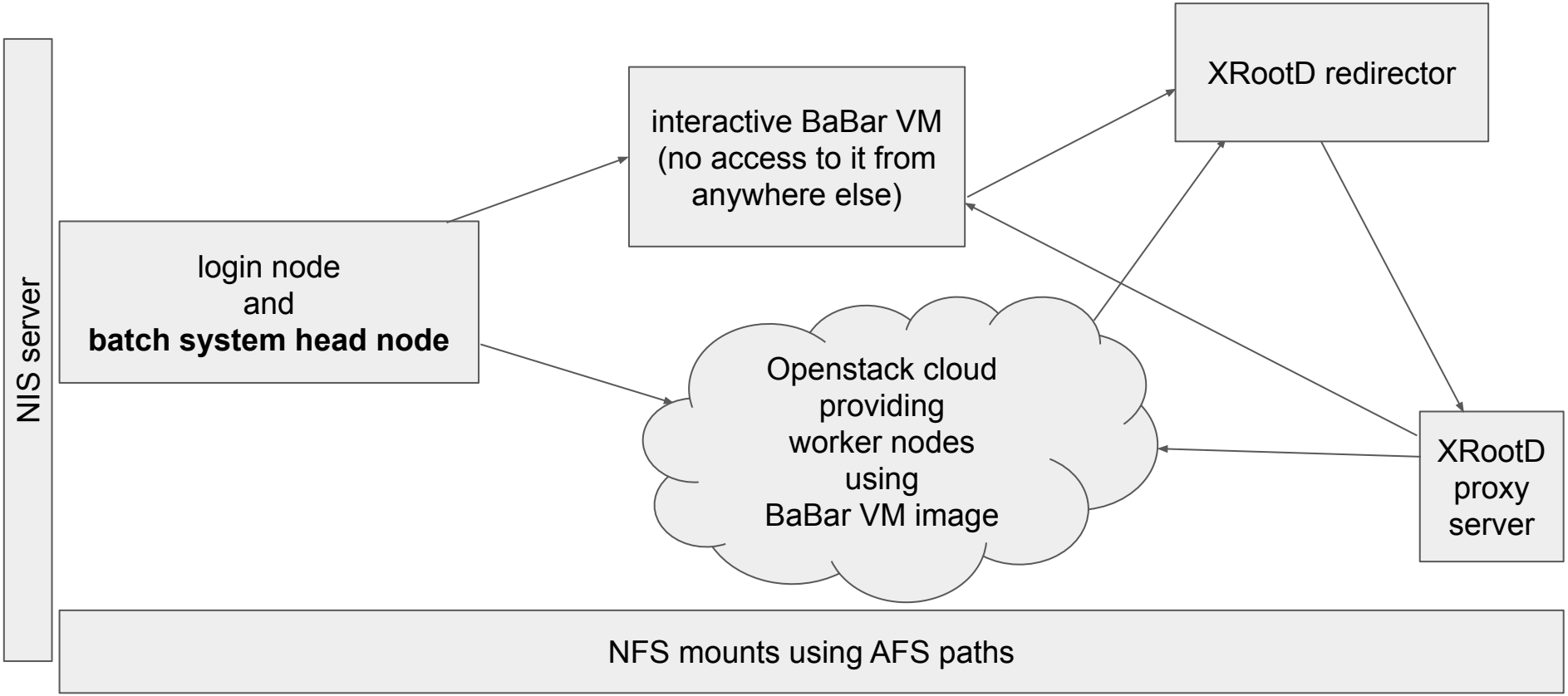
but...

- each analysis and all studies done need to be described in detail in a document
  - incl. software release and version of all packages, data and MC used,...

- used for internal review of analyses, but can also be used to repeat an analysis

These documents need to be preserved.

Hardware infrastructure



Hardware infrastructure

Hardware overview:

- XRootD proxy server: old machines
- XRootD redirector: VM on an old machine
- login machine: VM on an old machine
- BaBar interactive VM: VM on an old machine
- NIS server: VM on an old machine
- web server: on VM on an old machine
- babar wiki: VM on an old machine
- babar Hypernews: VM on an old machine
- NFS server: one new server, multiple old machines

old machine==out of warranty

Redundancy/Reliability:

- protect against disk failure
- protect against server failure

Documentation

- different systems used:
  - html web pages: in AFS within well defined directory structure, r/w rights via ACL, every BaBar user had a SLAC account; edit html files directly in AFS
  - Wiki: added ~2012 to have self contained system editable by anyone in the collaboration via web browser

- html web pages: visible to public or specific groups via .htaccess files, difficult to maintain content

- Wiki: visible only to BaBar members, easy to maintain content

Documentation preservation

documentation preservation: copy content to new web servers
(for html page and for wiki)

Documentation preservation

documentation preservation: copy content to new web servers
(for html page and for wiki)

Easy to do, but... issue again:

- often absolute URLs were used for links instead of relative paths
- some content dynamically created through db queries

- change URLs in html files relatively easy when keeping main structure the same
- db content not accessible and probably will never be - content in SLAC specific Oracle databases
  - SLAC infrastructure grew out of BaBar IT infrastructure
  - created static copies of most pages and content while db access was still possible
- having new content in html pages difficult
  - no user accounts on new web server system
  - on new analysis system only for active analysts get accounts

Freeze html content (pages are outdated) and have it no longer available to the public

Documentation preservation

documentation preservation: copy content to new web servers
(for html page and for wiki)

Easy to do, but... issue again:

- often absolute URLs were used for links instead of relative paths
- some content dynamically created through db queries

- change URLs in Wiki more complicated
  - content not stored in plain files but in mysql database
  - SLAC IT agreed to have on their web server redirects for BaBar URLs to the new server
  - people should change URLs manually when they come across links that go to SLAC

Wiki became main documentation for BaBar,
old html pages for historic purpose only,
new single public page for general information available.

Collaboration tools

- many different systems needed for the management of a collaboration and to have a communication between members
  - mailing lists
  - meeting pages
  - member lists
  - analysis management and documentations
  - review system for publications and talks
  - communication platform (Hypernews)

- all systems fully integrated into SLAC central systems and links between the systems
  - people's database
  - UNIX based authentication and ACLs used to access information
  - systems linked between each other (members database, analysis management system, working groups, mailing lists, Hypernews, email accounts)
  - BaBar specific scripts to query different db to display dynamic information
  - information, incl. binary data like pdf files, in different Oracle databases....

Collaboration tools

- SLAC based mailing lists ---> Caltech mailing lists
  - only created what is still needed

- old meeting agendas were HTML pages, registration based on SLAC systems

---> switch to use CERN Indico

- Hypernews was deeply integrated into SLAC
  - sending emails for posts to SLAC emails, notify SLAC systems in case of issues, people joining need SLAC UNIX account,... - but all content of posts in text files

---> moved Hypernews out of SLAC, made read-only, removed mailing features

-> **still readable and archive of any communication happened in the past**

---> replacement: CERN egoups

- also nicely integrated with CERN Indico for accessing BaBar meetings

Collaboration tools

- Analysis documents, notes, and Analysis metadata
  - old content archived to INSPIRE
  - new documents will be added too for long term preservation

<u>new system for active analyses and management:</u>

- Google drive folder for each analysis
  - for documents documents and other informations
- Google sheets for metadata of each analysis
- review done using CERN egroups (each analysis has its own)
- specific folders for SpeakersBureau, PublicationBoard,...

Knowledge preservation

- preserve old HyperNews system
  - communication and information exchange since the early years of BaBar
- preserve old HTML pages
- use self-contained information system (Wiki)

- **keep collaboration connected**
  - build community
  - mailing lists
  - CERN egroups
  - Collaboration meetings (virtual or in person)

Summary

- Physics data preservation and open access alone doesn't help for future possibility of using the data
  - one needs to **preserve data, analysis framework, and documentation**
  - other tools also important if the collaboration still needs to function at that stage, e.g. doing new analyses
  - for that reason, anyone can use the data, but in a "controlled way" when aiming for publication: 'BaBar Associates' open-access program

- large issues for infrastructure - too integrated into local systems
  - centralized and "non-free" databases with data of multiple groups
  - running services based on local accounts (Unix accounts)
  - important services running on personal accounts (cron jobs)
  - how information and code is written (absolute paths...)

- using systems that can be ported to other places (opensource) helps a lot

Conclusion

Using of Open Source software as much as possible helps.

Use relative path/URLs whenever possible.

Make sure no official production tasks run on individual accounts.

Keep collaboration connected (knowledge preservation).

Taking preservation into account during software/framework development helps.

Preservation efforts should start as early as possible and should assume anything currently in use will be lost.

Administration of active systems need manpower.

Freezing of framework and preserving in VM images gives independence.

Conclusion

Systems and organisations available to help with long term archival and preservation
DPHEP, OpenData portal, Inspire, ...

Data preservation status for BaBar

- BaBar is able to continue with data analyses after moving out of SLAC
  - learned a lot about framework and dependencies

- Real long term solution still missing...
  - What happens when collaboration stops to exist?
  - Old hardware may stop working
  - How long will GridKa continue to host the data?

Long term idea: Move to the Open Data portal at CERN

- Needs work/manpower
- Should start very soon, has to be ready when collaboration stops to exists

Redundancy/Reliability of hardware systems

login machine VM
NIS Server VM
interactive VM
XRootD redirector VM
---
hardware raid1 OS
ZFS mirror data disks

- spare server setup the same way
- ZFS send/receive

XRootD proxy server
---
hardware raid1 for OS
ZFS raidz3 data disks

- multiple servers available
- just cache, loose no data

Web documentation VM
Wiki VM
Hypernews (HN) VM
---
hardware raid1 for OS
ZFS raidz3 for data disks

- web content on NFS
- HN content on NFS
- images backed up
- daily mysql dump to NFS

4 NFS server:
NFS $HOME
NFS job output
NFS framework
NFS documentation
---
all use:
ZFS raidz2/3
hardware raid1 for OS

- spare server setup in the same way
- ZFS send/receive
- extra backup of framework and documentation