The H1 Data Preservation Project

David M. South*, Michael Steder* on behalf of the H1 collaboration

*Deutsches Elektronen-Synchrotron, Hamburg, Germany

Future Governance of the H1 Collaboration

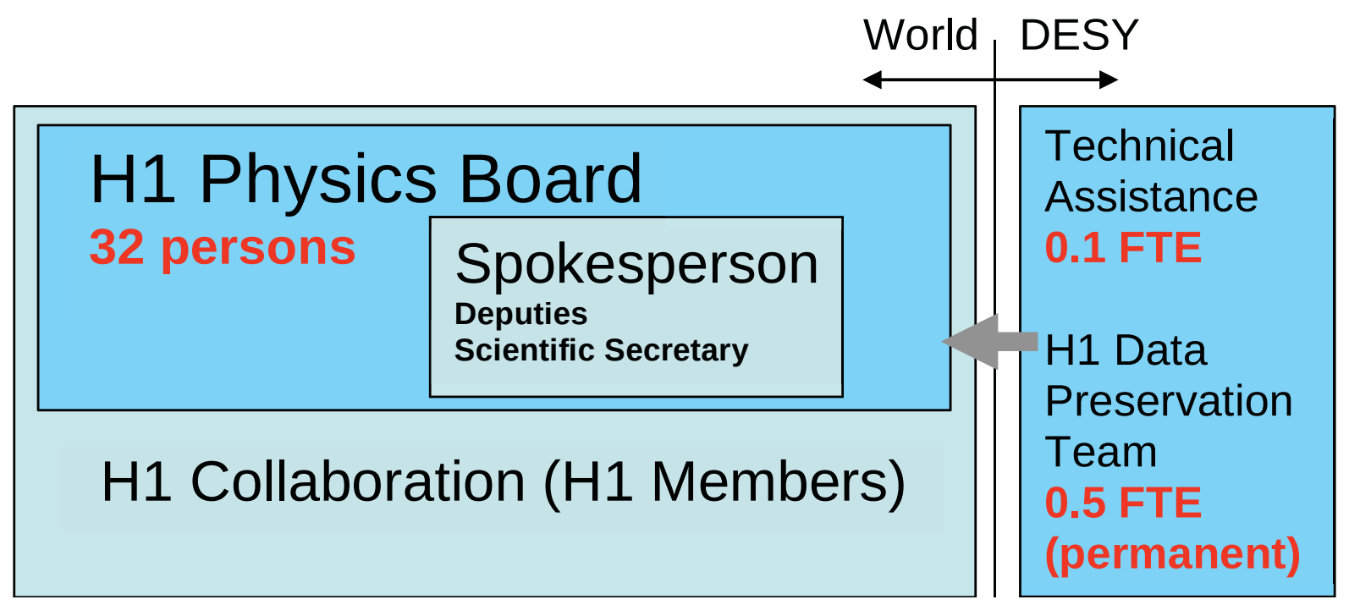

World
DESY
H1 Physics Board
32 persons
Spokesperson
Deputies
Scientific Secretary
H1 Collaboration (H1 Members)
Technical Assistance
0.1 FTE
H1 Data Preservation Team
0.5 FTE
(permanent)

Figure 1: The new organisational model of the H1 Collaboration, which will be adopted in July 2012.

A new model for long-term governance of the H1 Collaboration was adopted in 2011, where the transition from the current model will take place in July 2012. In the new model, the current H1 Collaboration Board (H1CB), which includes representatives from all participating institutes and the H1 Executive Committee, which is a smaller structure elected by the H1CB to meet more regularly, will be replaced by the H1 Physics Board (H1PB). The mandate of this board, which comprises a broad selection of H1 members from all physics and technical working groups is: to be the general contact point for H1 physics and data beyond the collaboration lifetime; to communicate with the host lab (DESY) and other experiments; to supervise the H1 data: to maintain contact with he global DPHEP initiative; and to overview further publications using H1 data. The new organisational model is illustrated in the *figure 1*.

H1 Data and MC Samples

The good and medium quality H1 raw data comprises around 75 TB and is the basic format to be preserved. A full set of Compressed Data Storage Tape (CDST) data for the 1996-2007 period is about 15-20 TB, and the analysis level files (H1OO) are around 4 TB. Other data, such as random trigger streams, noise files, cosmic-data, luminosity-monitor and other calibration data amounts to only a few TB. Standard MC sets for preservation will also be defined, where the total data volume likely to be similar to real data. The total preservation volume, including MC and non-collision data, is conservatively estimated to about 0.5 PB.

Data Preservation Models

As certain analyses may require the production of new simulated signals or even require a re-reconstruction of the real and/or simulated data, the H1 Collaboration plans to keep the full software chain from raw data to analysis level functional ('level 4'). It is clear that this level of preservation will necessarily include the full range of both experiment-specific and external software dependencies, although attempts to minimise the latter should be carried out in the initial step. However the clear benefit of such a model is that the full physics analysis chain is available and full flexibility is retained for future use.

Experiments		Preservation Model	Use Case	
	1	Provide additional documentation	Publication related info search	Cost, complexity and benefits
	2	Preserve the data in a simplified format	Outreach, simple training analyses	
	3	Preserve the analysis level software and data format	Full scientific analysis, based on the existing reconstruction	
	4	Preserve the reconstruction and simulation software as well as the basic level data	Retain the full potential of the experimental data	

Table 1: Levels of data preservation as suggested by the DPHEP study group.

Simulation, Reconstruction and Analysis Level Software

Simulation and Reconstruction Software

The H1 reconstruction and simulation software, which creates DSTs from the raw data is written mainly in Fortran, but also contains some C and C++. The MC simulation takes the generator files as input and passes them through GEANT 3, taking the relevant run conditions from a database, to produce MC events in the same format as the data - with some additional information. The same reconstruction software as used on the data is then applied to the simulated MC events. As new theory or new experimental methods are likely to be the prime reasons for re-analysing the H1 data, scenarios may arise where only a full preservation model will provide the necessary ingredients, for example if a cut in the current reconstruction turns out to have been too harsh, or a new simulation model, written in an alternative computing language, requires a new interface to the existing code.

Analysis Level Software

The majority of physics analysis performed by the H1 Collaboration is done using a common C++ analysis framework, H1OO. This has had huge benefits in terms of shared analysis code, expert knowledge and calibrations, working environments and, perhaps most importantly, handling the actual data, where the whole collaboration uses the same file format, and more often than not the same physical files. The common H1OO files comprise in reality of two persistent file formats: the *H1 Analysis Tag* (HAT), containing simple variables for use in a fast selection and the larger *micro Object Data Store* (mODS), which contains information on identified particles. A third H1OO file format, the *Object Data Store* (ODS) is accessed transiently during analysis and is equivalent in content to the DST. The H1OO framework is based on ROOT, and uses its functionality for I/O, data handling, producing histograms, visualisation and so on. ROOT also provides attractive solutions for code documentation, which are fully utilised by H1. Given the level of use in the HEP community, especially at the LHC, it is expected that ROOT will continue to be supported in the long term. Major development of the analysis software is essentially completed with the recent 4.0 release series, which was developed for and in parallel to DST 7.

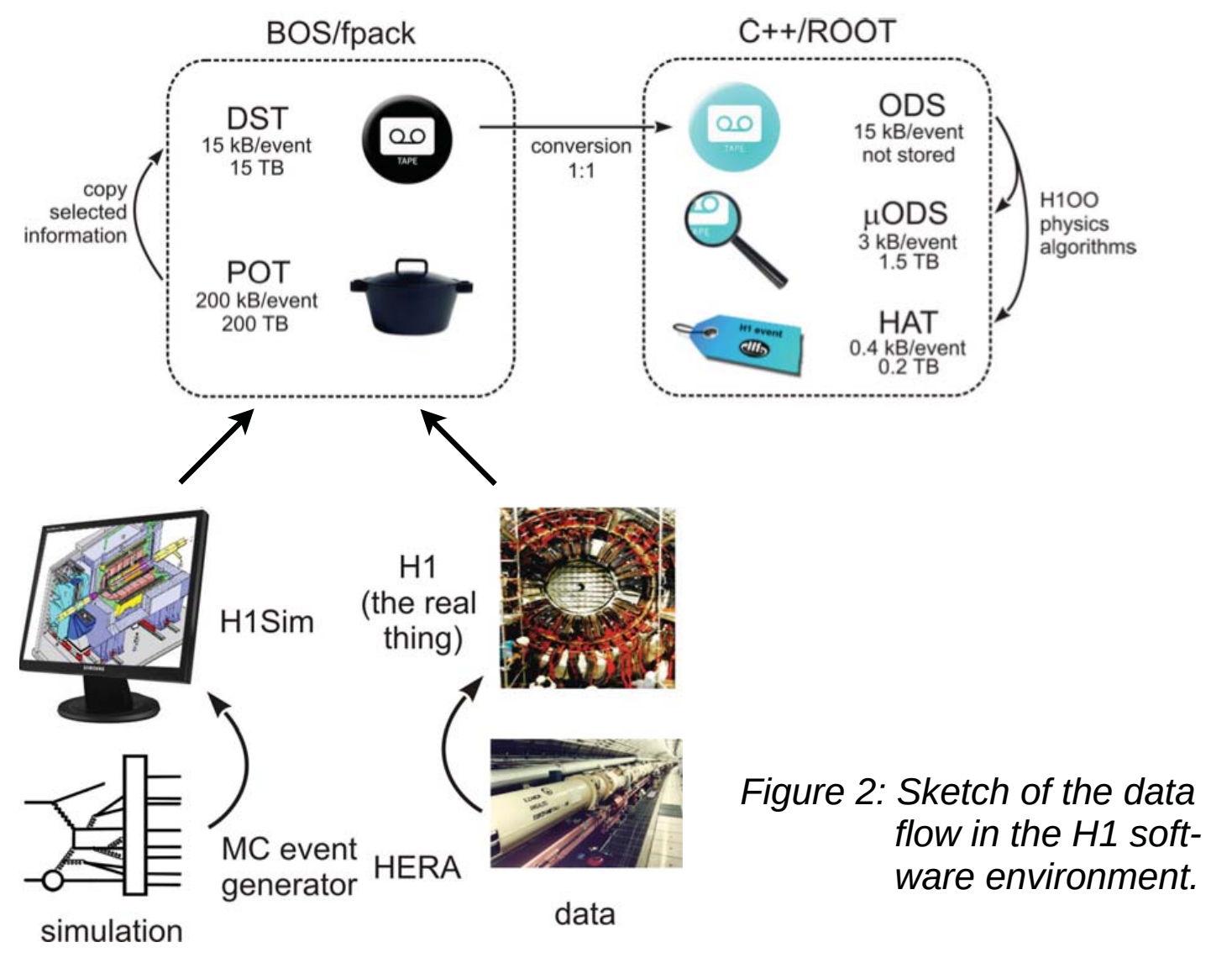

BOS/fpack
C++/ROOT
DST
15 kB/event
15 TB
POT
200 kB/event
200 TB
copy selected information
conversion 1:1
ODS
15 kB/event
not stored
μODS
3 kB/event
1.5 TB
HAT
0.4 kB/event
0.2 TB
H1OO physics algorithms
H1Sim
H1 (the real thing)
MC event generator
HERA
simulation
data

Figure 2: Sketch of the data flow in the H1 software environment.

Non-digital Documentation

A general survey of the state of the non-digital H1 documentation has been performed. There is a great deal of paper documentation: H1 physics and technical talks from pre-web days; detector schematics and blueprints; artefacts from the experimental hall like older logbooks. A future location large enough to store all the documentation for preservation has been secured in the DESY-Library. However, the cataloguing and organisation of large quantities of documentation is also a significant task that can only be done by someone with expert knowledge of the H1 Collaboration.

Cooperation with INSPIRE

The INSPIRE project has offered, via the DESY-Library, to aid the documentation effort and several pilot projects are underway with the HERA collaborations: including the ingestion of internal notes, digitisation of theses and electronically cataloguing the publication histories (preliminary results, T0 and referee reports, versions of the paper draft). The larger scale digitisation of older H1 documentation is also underway, although given the volume of material, this again has required prioritisation, where preference has been given to plenary meetings.

Figure 3: Jan scanning old log books, H1 storage in library basement.

Structure of the H1 Data Preservation Project

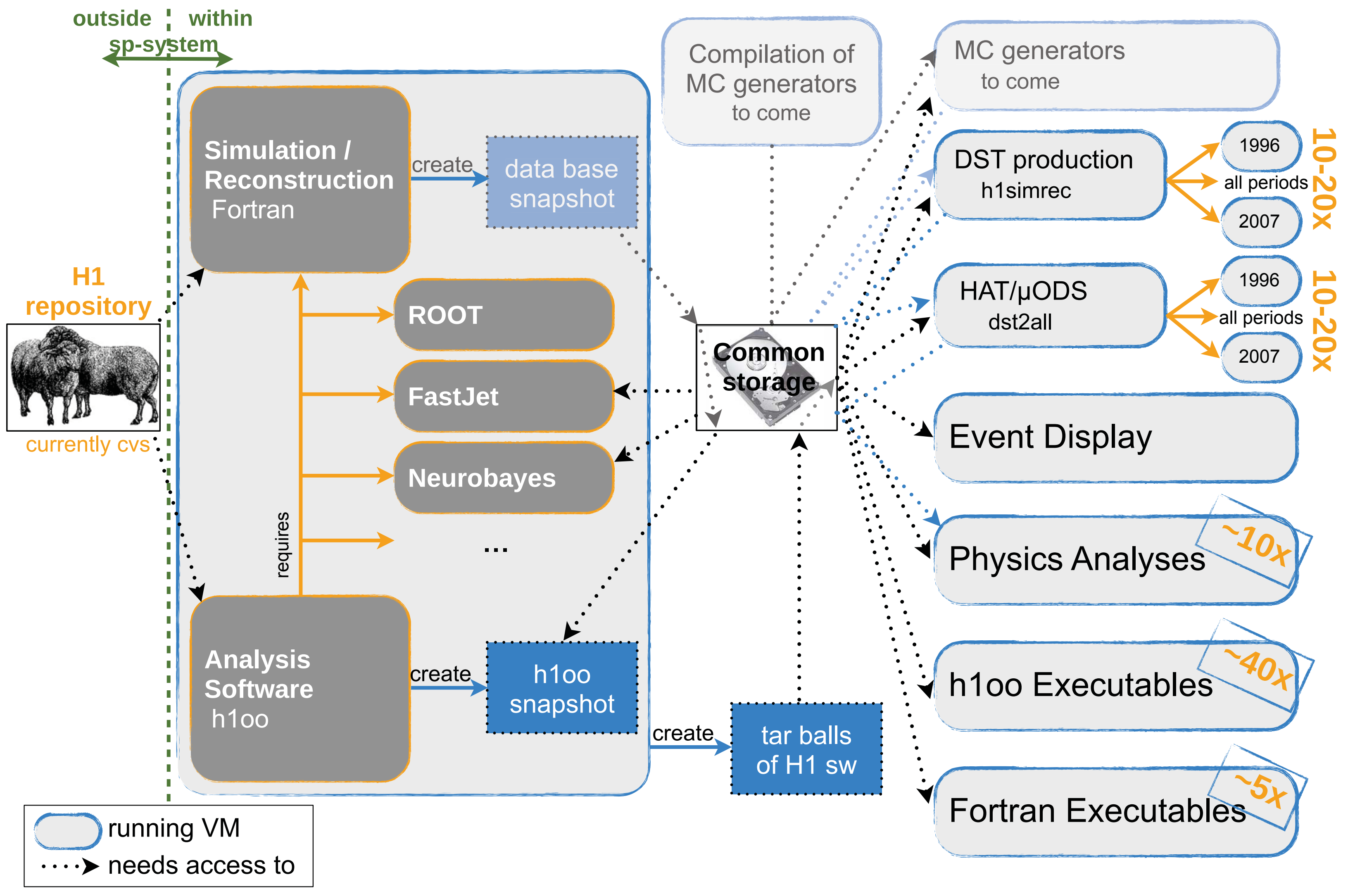
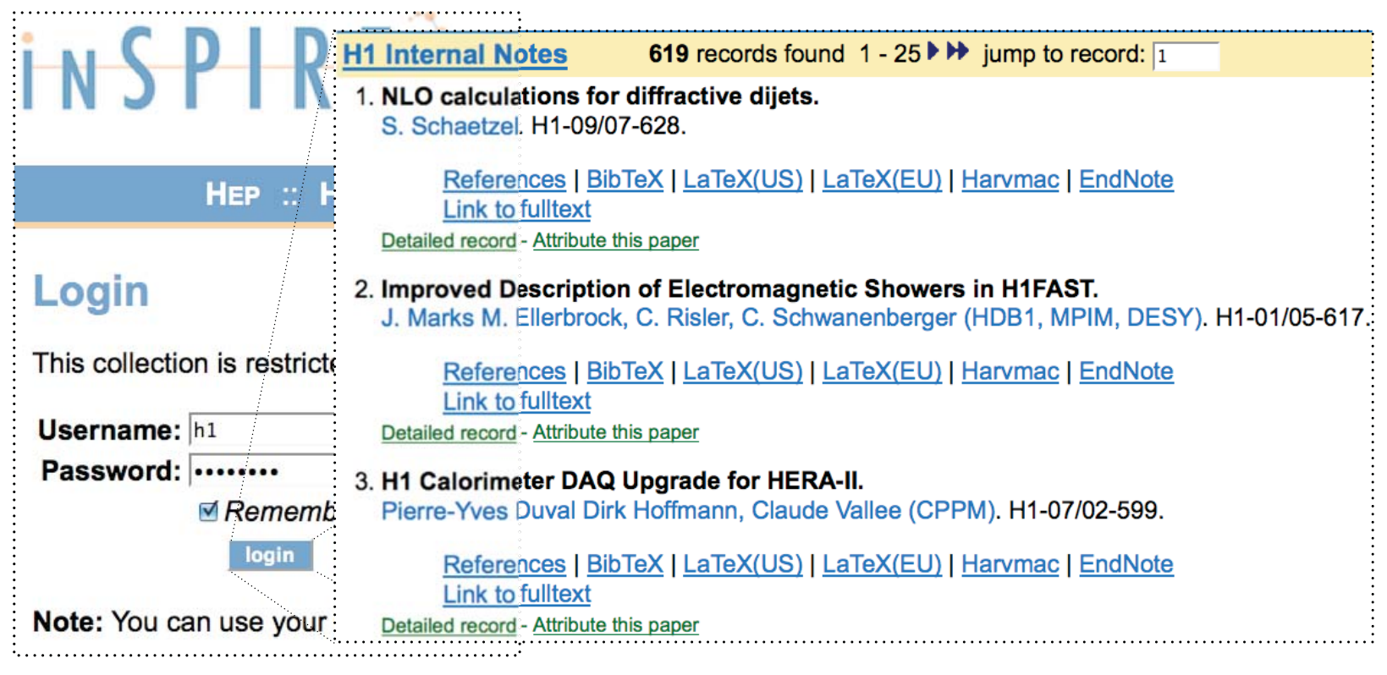

outside
within
sp-system
H1 repository
currently cvs
Simulation / Reconstruction Fortran
create
data base snapshot
ROOT
FastJet
Neurobayes
...
requires
Analysis Software h1oo
create
h1oo snapshot
create
tar balls of H1 sw
Common storage
Compilation of MC generators to come
MC generators to come
DST production h1simrec
1996
all periods
2007
10-20x
HAT/μODS dst2all
1996
all periods
2007
10-20x
Event Display
Physics Analyses
~10x
h1oo Executables
~40x
Fortran Executables
~5x
running VM
needs access to

Figure 4: Structure of the H1DP project. The left part summarises the compilation of the full reconstruction and analysis software chain, and its external dependencies. The right part indicates all envisaged validation tests, starting with MC generation/raw data via simulation/reconstruction to physics analyses.

Digital Documentation

A great deal of digital H1 documentation exists, mainly but not exclusively on the official webpages. This includes published papers and preliminary results, review articles and expert notes. In addition, talks from meetings, conferences, lectures, and university courses are also available. There are also many unpublished articles, such as H1-notes and the internal wiki pages that are extensively used by H1. Data quality information (physics and technical) and other electronic documentation like H1 software manuals and notes also contribute. In-house DDL documentation of Fortran software (h1banks) should be updated and/or completed. As mentioned above, the H1OO analysis level software benefits from the automatically generated ROOT documentation in HTML, but only if the code is correctly written, and any missing information should be addressed. Old online shift tools contain much metadata and are particularly vulnerable to loss. Such information has mostly not updated since July 2007 and electronic logbooks (shift, trigger and other detector components) and detailed run information contained in the system supervisor should be secured. Calibration files may still exist on old hardware: in excess of 20 online machines were employed during data taking.


INSPIR
HEP :: H
Login
This collection is restricte
Username: h1
Password: ••••••••
☑ Rememb
login
Note: You can use your

H1 Internal Notes 619 records found 1 - 25 ▶ ▶▶ jump to record: 1

1. NLO calculations for diffractive dijets.
S. Schaetzel. H1-09/07-628.
References | BibTeX | LaTeX(US) | LaTeX(EU) | Harvmac | EndNote
Link to fulltext
Detailed record - Attribute this paper

2. Improved Description of Electromagnetic Showers in H1FAST.
J. Marks M. Ellerbrock, C. Risler, C. Schwanenberger (HDB1, MPIM, DESY). H1-01/05-617.
References | BibTeX | LaTeX(US) | LaTeX(EU) | Harvmac | EndNote
Link to fulltext
Detailed record - Attribute this paper

3. H1 Calorimeter DAQ Upgrade for HERA-II.
Pierre-Yves Duval Dirk Hoffmann, Claude Vallee (CPPM). H1-07/02-599.
References | BibTeX | LaTeX(US) | LaTeX(EU) | Harvmac | EndNote
Link to fulltext
Detailed record - Attribute this paper

Figure 5: Screenshots of the H1 internal notes on Inspire (password protected).

Software Validation within the DESY-IT `sp-system`

In order to substantially extend the lifetime of the analysis capability it is beneficial to migrate to the latest software versions and technologies for as long as possible. In collaboration with DESY-IT a framework has been developed to automatically test and validate the software and data against such changes and upgrades to the environment, as well as changes to the experiment software itself. An illustration of this software preservtion system (sp-system) is given in *figure 6*. Technically, this is realised using a virtual environment with different configurations of operating systems (OS) and the relevant software, including any necessary external dependencies.

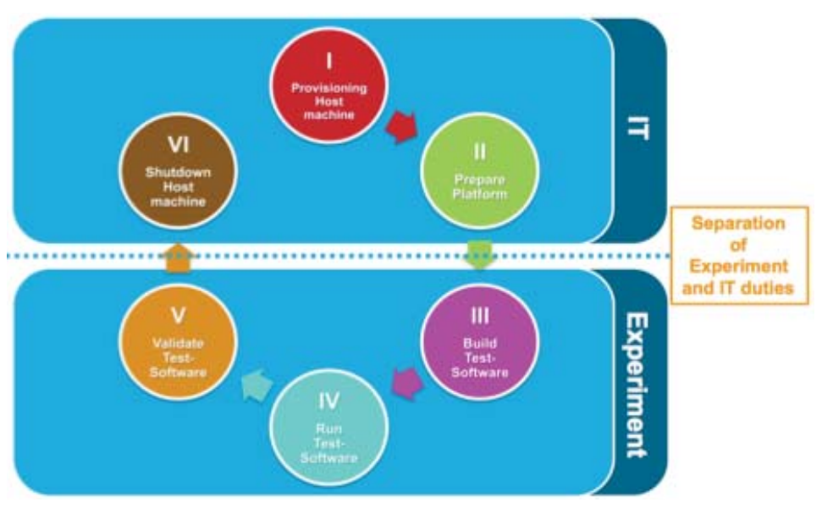

I Provisioning Host machine
II Prepare Platform
III Build Test-Software
IV Run Test-Software
V Validate Test-Software
VI Shutdown Host machine
IT
Experiment
Separation of Experiment and IT duties

Figure 6: An illustration of the basic idea behind the generic validation framework being developed at DESY.

H1 Data Preservation Project

The preliminary structure of the tests to be installed by the H1 experiment to validate the full analysis chain is shown in figure 4. The left part details the compilation of experimental and external software. This is considered as a series of tests, where the compilation of approximately 100 individual packages is carried out. The resulting binaries are stored as tar-balls on a central storage facility within the validation framework, where they are then accessible and used in the predefined tests, described on the right of the figure. These tests are wide reaching, examining all areas of the H1 software including among others file production ("DST production" and "HAT/μODS"), comparison of analysis histograms ("Physics Analyses") and execution of experiment specific tools and macros ("h1oo/Fortran Executables"). H1 estimates a total of around 250 tests are required, to successfully validate the complete analysis chain, although it should be noted the implementation is still within the development phase.

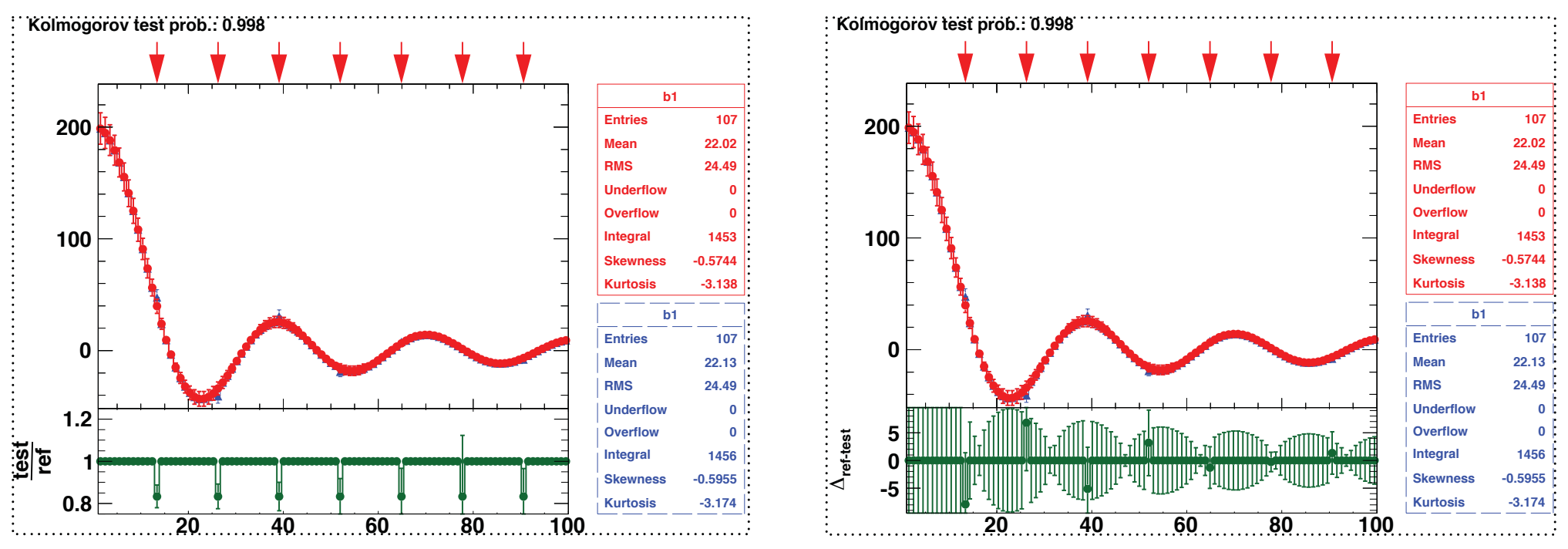


Figure 7: Output of the Root file validation tool (arbitrary histograms), along with the histograms, the ratio or absolute difference can be plotted.

Detailed Status of the H1 Data Preservation Project

Reference Operating System

The main OS used within H1 is Scientific Linux DESY 5 (SLD5), this OS is fully supported by DESY-IT and support is expected from the distributor until at least 2017. The migration to SLD5, which is the base level for the preservation project, was used to streamline the software and identify potential future problems. The default for H1 is SLD5/32-bit.

Migration to 64-bit Systems

The focus of the H1 Software Validation Project right now is on the full migration to 64-bit operating systems, starting with SLD5/64-bit. Compilation of all H1 simulation, reconstruction and analysis level software works with only minor modifications, external dependencies are included in the compilation as long as they are not centrally provided (e.g. Root, Neurobayes,…). Incorporating compilation of MC generators is on-going. Files can be produced and accessed, using libraries and executables created within the sp-system. First checks in the validation system yield very promising results and the sp-system allows SLD5/64-bit compatibility to be evaluated by the H1 collaboration.

Next Steps

As soon as SLD5/64-bit is fully validated (by the end of this year), the system will be rigorously tested against SLD6. This next generation OS will only be available in 64-bit, demonstrating the value of the current evaluation. Furthermore, lack of 32-bit support of next generation hardware might make use of a 64-bit OS inevitable well before 2017.

🟩 ok
🟨 ongoing
⬜ not yet done
🟥 problem

External Dependencies / Process	SL5 32bit	SL5 64bit									SL6 64bit
	Reference	ROOT				Cernlib		Fastjet	Neuro-bayes	Neuro-bayes	
	←	5.26	5.28	5.30	5.32	2005	2006	2.3.3	2008 0312	3.3.0	
Compilation of s/w	🟩	🟩	🟩	🟩	🟩	Use newer version	🟩	🟩	🟩	Centrally supported by IT, to be use soon	🟨
Generating MC files	🟩	🟩	🟩	🟩	🟩		🟩	🟩	🟩		⬜
Producing DST files	🟩	🟩	🟩	🟩	🟩		🟩	🟩	🟩		⬜
Producing h1oo files	🟩	🟩	🟩	🟩	🟩		🟩	🟩	🟩		⬜
Accessing h1oo files	🟩	🟩	🟩	🟩	🟩		🟩	🟩	🟩		⬜
Accessing ndb snapshot	🟩	🟩	🟩	🟩	🟩		🟩	🟩	🟩		⬜
Validation	🟩	🟨	🟨	🟨	🟨		🟨	🟨	🟨		⬜

Table 2: Status of the individual steps of the H1 software validation project for different external dependencies.

Display of Results and Book-keeping

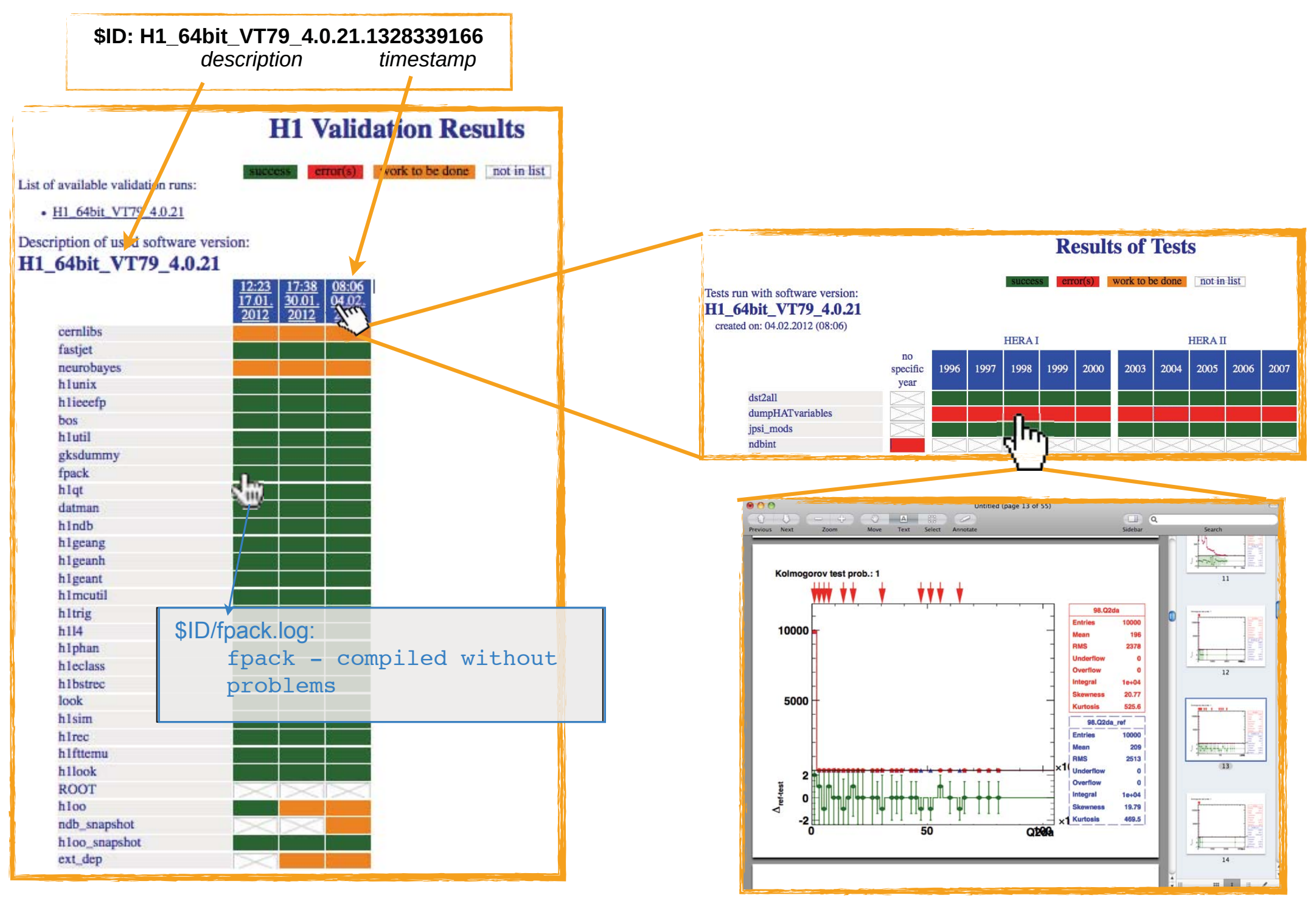
Every single job started in the sp-system is related to a unique ID, all scripts and input files used in the test as well as all output files are kept. This allows to validate all version against one another or even to reproduce the result in case that it's lost. In addition to this UID given by the sp-system, all jobs started in the H1DP project are tagged with a description, indicating which software versions were used, and the unix timestamp of the execution to ease the book-keeping.

Display of Software Compilation Result

A script-based webpage lists all available runs for a given description and indicates the status of the compilation for the individual packages with a colored table cell, which is linked to the corresponding output file.

Display of Validation Results

The headline cells of this table open another web page displaying the results of the various validation tests run for this version. Again the results are indicated by colored table cells, which are linked to additional information, like the created histograms or produced Root files.


$ID: H1_64bit_VT79_4.0.21.1328339166
description timestamp

H1 Validation Results
success | error(s) | work to be done | not in list
List of available validation runs:
- H1_64bit_VT79_4.0.21
Description of used software version:
H1_64bit_VT79_4.0.21
12:23 17.01. 2012 | 17:38 30.01. 2012 | 08:06 04.02. 2012
cernlibs, fastjet, neurobayes, h1unix, h1ieeefp, bos, h1util, gksdummy, fpack, h1qt, datman, h1ndb, h1geang, h1geanh, h1geant, h1mcutil, h1trig, h1l4, h1phan, h1eclass, h1bstrec, look, h1sim, h1rec, h1fttemu, h1look, ROOT, h1oo, ndb_snapshot, h1oo_snapshot, ext_dep
$ID/fpack.log:
fpack - compiled without problems

Results of Tests
success | error(s) | work to be done | not in list
Tests run with software version:
H1_64bit_VT79_4.0.21
created on: 04.02.2012 (08:06)
no specific year | HERA I: 1996 1997 1998 1999 2000 | HERA II: 2003 2004 2005 2006 2007
dst2all, dumpHATvariables, jpsi_mods, ndbint

Figure 8: Screenshots of the web pages displaying the result of the compilation for the individual packages (left part) and the results of the validation tests for a given version (right part).