source, and integration of more than one million lines before the start of data taking. We successfully orchestrate centralized integration at Fermilab of software developed over six physically separate institutions.

The survey continues to follow the spirit of the POSIX standards, and to incorporate them to the greatest extent feasible. ANSI-C, Extended F77 and C++ are the allowed programming languages. We include much public domain software into our central code repository and support infrastructure. As reported previously, other software standards in use are: X11, WWW and HTML, perl, TeX, CVS and RCVS, Fermilab UPS, tcl/tk, and an Object-Oriented Database for the survey's DBMS.

In this paper we report on our experiences and the perceived benefits in using and "enforcing" these standards. We report on further developments and directions we are taking for organizing and managing the software of the survey. In particular, we are now placing emphasis on developing tools for quality control and analysis and we present details of our tools and methods in these areas. The successful implementation and adoption of our methodologies within the collaboration gives us the confidence that new tools we develop can and will be applied to good effect across the breadth of the collaboration's software.

ABS_81

BR9737207

Offline Computing and Communications in the BES Collaboration

XinChou Lou (Texas)
BES Collaboration

This paper describes the offline computing, data handling and communications in the BEijing Spectrometer (BES) Collaboration.

Unlike other major High Energy Physics experiments at CERN, Fermi Lab and other laboratories, the BES experiment, centered at the Institute of High Energy Physics, in Beijing, P.R. China, has lacked funds to purchase or lease mainframe systems such as IBM or CRAY to facilitate the experiment's needs for event reconstruction and data handling. A distributed, RISC processor based Unix computing system and data farm provides the natural, economical solution to BES's long standing problem of shortage in computing power. A new Unix system is rapidly being built at IHEP in China for offline reconstruction. Several Unix sites have been developed in the US, each maintains a specific set of data and provides the collaboration with access to the data. Codes and constants are managed by CODEMAN simuntaneously at many sites. Software tools have been developed to facilitate batch job processing and tape handling automation. The steady improvement in computing and data handling in BES has enabled the collaboration to study a broad spectrum of physics topics recently. The low cost BES Unix systems serve as a example for small university groups as well as labs in developing countries.

ABS_48

BR9737208

Using WWW to improve software development and maintenance: Application of the LIGHT system to ALEPH programs

B.Rousseau (CERN)
A.Aimar, M. Aimar, A.Khodabandeh, P.Palazzi, M.Ruggier, P.Comas Illas (CERN)

Programmers who develop, use, maintain, modify software are faced with the problem of scanning and understanding large amounts of documents, ranging from source code to requirements, analysis and design diagrams, user and reference manuals, etc. This task is non trivial and time consuming, because of the number and size of documents, and the many implicit cross-references that they contain. In large HEP collaborations, where software and related documents are produced at various sites, the problem can be even more severe.

LIGHT, LIfe cycle Global HyperText, is an attempt to solve the problem using WWW technology. The basic idea is to make all the software documents, including code, available on the WWW, with all cross-references automatically established. For instance a click on an external subroutine call in the code takes you to its documentation, a click on a data element leads to the corresponding data definition, etc.

The first practical application of LIGHT is for JULIA, the reconstruction program of ALEPH. Programmers, documentation writers, maintainers of the data model and end users will be able to view through the web the entire Fortran source code, data definition, data design diagrams, as well as the JULIA, ALEPHLIB, CERNLIB, ADAMO and BOS manuals. All these documents are connected with hypertext links in such a way that, just using the mouse, helpful inspection can be performed.

For instance, from the code of a JULIA subroutine it is possible to navigate to: * the source code of calling and called subroutines, * the ALEPH data definition

files and design diagrams, * ALEPH Library manual and source code, and more. Hyperized indices and dependency trees help the navigation.

The LIGHT system is generic by design; it can be applied to projects that use languages and formats other than those employed in ALEPH. The system is made up of two software layers. The first consists of an extensible set of converters to HTML (the HyperText Markup Language used in WWW), one per source language or format, to ensure the automatic conversion of individual documents. These are plugged into the generic second software layer that automatically performs the hypertext interconnection among all the converted documents, according to the preferences expressed in configuration rules.

Other ongoing applications of LIGHT concern the interactive access to physics data through WWW, and the wider coverage of the software life cycle, according to standards such as PSS-05 for which LIGHT is an ideal complement. The JULIA Web generated with the LIGHT system exists now in the form of an advanced demonstrator containing 6000 HTML pages and more than 110000 hypertext links. At the time of the conference it will hopefully be in production.

We have implemented several converters for Fortran 77, ADAMO Data Definition Language, OMTool diagrams and FrameMaker documents. LaTeX manuals are converted using LaTeX2html. Converters for C and C++ are in progress. The specification of the interface to build new converters or adapt existing ones for LIGHT compatibility are being finalized.

More information about LIGHT, as well as several examples can be accessed on WWW at the following URL:

http://www.cern.ch/Light/

BR9737209

ABS_39

Code Management in a Large International Collaboration

T. GLANZMAN (SLAC)
T. Hung (SLAC), R. Jacobsen (U. California)

The BABAR collaboration is a large international initiative with over 450 members spanning 78 institutions in 10 countries. Work is now underway to develop the software for this experiment, an effort which will extend well past the commencement of data collection in 1999. This software development must reflect the highly distributed nature of the collaboration in that people working on code, databases and documentation need to fully participate from their home institutions. The challenges associated with this requirement are many, and include: a robust organization of the software such that files and directories may be readily located and accessed, new ones added, old ones retired; a mechanism to automatically and safely coordinate the updating of a master file repository; a degree of protection against accidental file/directory corruption; and, a mechanism to precisely identify and access a specific release of the entire set of production software. In addition, we must contend with the relatively poor network connections in place between certain collaborating institutions, and the need to support computing platforms from different vendors. The BABAR collaboration has developed a number of basic guidelines and identified a set of technologies to simplify the problem and limit the complexity of implementation. Important among these is the exclusive use of Unix as an operating system, reliance upon standard networking protocols such as TCP/IP and the desire to use existing commercial or freely available software where applicable. Obviously, the system must also be easy to use, extend and maintain. Software organization relies on a single directory tree hierarchy based upon a standard Unix root directory structure and allows for differences associated with different vendor architectures. Unix groups are used for directory protection within the NFS implementation, and ACLs within the AFS implementation. Coordinators for individual software "packages" have rights and responsibilities within a portion of this directory tree. Within high energy physics, software is in a continual state of development. BABAR has adopted the CVS package as the component to manage and regulate changes to individual files. CVS, however, only operates on a local file system, thus it was necessary to design and develop a means to extend this functionality to other institutions within the collaboration. The result, rCVS, is a continuing R and D project and we are gaining experience with the concept of distributing software across less-than-ideal network connections. A person running collaboration software, whether it be online data acquisition, offline reconstruction, simulation or an analysis needs some mechanism to determine just what is being used, along with the option of specifying a particular version (or combination of versions) of the software. This is production release management. A prototype mechanism has been installed and is currently in use.

D.4 OPEN SOFTWARE STANDARDS AND INTERCHANGEABILITY

ABS_95

BR9737210