Technical Report GriPhyN-2001-xx
www.griphyn.org

GriPhyN Year 2 – CMS Project Plan

21 December 2001

Developed by members of the GriPhyN Project Team

Editors: Mike Wilde and Rick Cavanaugh
wilde@mcs.anl.gov, cavanaug@phys.ufl.edu

1 Introduction

The CMS GriPhyN efforts during Year 1 consisted primarily of researching the grid requirements for CMS [GriPhyN 2001-1], [GriPhyN 2001-16] and in studying the current status of CMS software, as it relates to the production of simulated CMS data and the future analysis of that data. Such research, which has proven vital to the planning of tasks, will continue in Year 2 as CMS refines and redefines its software environment.

One important consideration is that the immediate CMS demand for software and grid functionality outweighs the current need for scalability. As a result, this project plan details CMS activities for GriPhyN project Year 2, in which we plan to integrate virtual data GriPhyN research results and functionality into the production of CMS simulated data and begin to experimenting with applying virtual data concepts to the problem of CMS data analysis.

1.1 Pertinent CMS Software

While the CMS software is continuing to develop at a rapid pace, it is already highly functional. Indeed, successfully integrating virtual data technologies into an already advanced software environment presents a particular challenge to the CMS GriPhyN team. As there are several packages with which to contend, a brief introduction to some of the pertinent CMS software is given here:

- Simulation of physics events and the CMS detector: Loose collections of Fortran applications, collectively known as CMKIN, simulate different types of physics events resulting from proton-proton collisions. Subsequently, a single Fortran application, CMSIM, simulates the "raw" response of the CMS detector to the physics events produced from the CMKIN application. A typical simulation of 500 events requires approximately 8 hours of processor time (on 1 GHz machine) and produces a 50 GB flat file.
- Reconstruction of physics events: An OO framework, ORCA/CARF, is currently used to reconstruct the "raw" information contained in a physics event and translate it into data, which is

useful for physicists to analyze. The resulting data is stored in an Objectivity database. The input to this reconstruction framework can either be real data as taken from the actual CMS detector or, a simulated data file as produced by the CMKIN/CMSIM applications.

- Mass production of simulated CMS data: A major goal of CMS is the production of large sets of simulated data. The current production of simulated data follows a simple pipeline: (1) input parameters are given to CMKIN/CMSIM which produce a "raw" simulated data file, (2) this data file is copied from a flat file format to an Objectivity database using ORCA/CARF, and finally (3) ORCA/CARF is used to transform the "raw" data into "reconstructed" data and store it in an Objectivity database. IMPALA/BOSS consists of a set of shell scripts, which facilitate the mass production of simulated CMS data by performing input parameter bookkeeping, as well as job submission and tracking. To date, CMS has produced approximately 20 million simulated data events using various versions of IMPALA/BOSS.

This brief CMS software summary will change during Year 2 as the CMS Core Application Software team continues the development process. However, the basic structure of the simulated data production chain is expected to remain relatively constant until sometime in Year 3, when the current Fortran based simulation of the CMS detector will be replaced by an OO based simulation.

1.2 Outline

This document is outlined as follows. First, a broad overview of the GriPhyN activities in CMS is given in Section 2. Section 3 is devoted to a more detailed description of the development tasks foreseen for Year 2 in support of the high-level goals of Section 2. This is followed by a discussion of test bed development in Section 4. A brief conclusion is given in Section 5. Finally, a detailed work plan for Year 2 is presented in the Appendix. The provided work plan is, nevertheless, subject to change as the year evolves.

2 Project Overview

This section presents a broad overview of CMS GriPhyN. It is important to understand that new, and perhaps unexpected, developments will occur during the course of the year. Hence, a substantial amount of time will be budgeted to monitor the progress of CMS software and toolkit development as well as emerging technology from other grid projects and to adapt the CMS GriPhyN Year 2 activities accordingly.

2.1 Goals

The currently identified high-level goals for CMS GriPhyN for year 2 are:

- Show the utility of GriPhyN technology as a basis for enhancing the robustness and reproducibility of distributed computing, by integrating grid components more deeply into the CMS production software, with the results of the integration being used either in real production or in challenge demos.
- Gain and demonstrate the commitment of CMS to the evaluation, testing, integration, and use of GriPhyN technologies.
- Create a test bed in which both GriPhyN and PPDG CMS activities can be conducted.
- Forge joint activities and tighter coordination between PPDG, European Data Grid, TeraGrid, and GriPhyN (as CMS is part of all four grid projects).
- Apply GriPhyN virtual data research to CMS simulation production. Deploy the virtual data mechanism and use it to provide automated production as well as a GriPhyN laboratory.
- Integrate the virtual data catalog into an important production application, and demonstrate the benefits of detailed data derivation tracking and large-scale data re-derivation
- Instrument and measure the use of virtual data and request planning mechanisms to gather data and feedback for further CS research.
- Apply preliminary GriPhyN research results and execution planning and scheduling mechanisms (for example, using DAGMan) to CMS production.

Name	Affiliations	Role
James Amundson	PPDG FNAL CMS	CMS Physicist; MOP Developer
Paul Avery	GriPhyN UFL CMS	GriPhyN PI; UFL CMS Physics Lead
Lothar Bauerdick	GriPhyN FNAL CMS	US CMS Computing Coordinator
Dimitri Bourilkov	GriPhyN UFL CMS	CMS Physicist
Rick Cavanaugh	GriPhyN UFL CMS	CMS Physicist
Greg Graham	FNAL CMS	CMS Physicist; IMPALA Developer
Koen Holtman	GriPhyN CIT CMS	CMS Computer Scientist
Iosif Legrand	PPDG CIT-CERN-CMS	CMS Physicist
Vladimir Litvin	GriPhyN CIT-CMS	CMS Physicist
Harvey Newman	GriPhyN PPDG CIT-CMS	CMS Physicist; CMS GriPhyN Lead
Rajesh Rajamani	Condor UW-CS	Computer Scientist, CMS Application Support
Jorge Rodriquez	GriPhyN UFL-CMS	CMS Physicist
Conrad Steenberg	PPDG CIT-CMS	CMS Physicist
Jens Voeckler	GriPhyN UC-CMS	Computer Scientist
Edwin Soedermadji	iVDGL CIT-CMS	CMS Computer Scientist
Suresh Singh	GriPhyN CIT-CMS	CMS Computer Scientist
Julian Bunn	GriPhyN CIT-CMS	CMS Physicist
Takako Hickey	PPDG CIT-CMS	CMS Computer Scientist

Table 1. CMS GriPhyN Team. This table lists the people expected to contribute to CMS work in GriPhyN over Year 2. While many people are directly associated with GriPhyN, several are affiliated with other grid projects, or with CMS. This inter-project team make-up is considered important for the integration and use of tools developed outside of GriPhyN. It is hoped to expand this collaboration to include members of the European DataGrid and the CMS DataGrid.

- Start exploring the CMS analysis process, creating prototypes of GriPhyN-based analysis systems that can lead the way to live science use in project year 3.

2.2 Activities

Personnel involved in the CMS GriPhyN activities in Year 2 are listed in Table 1. The activities for this team during Year 2 will consist of test bed development, integration of VDT technology into CMS simulation production, and the prototyping of using VDT technology in the CMS analysis process. These are described in the sections below.

2.2.1 Production of CMS Simulated Data

The main GriPhyN CMS effort for Year 2 is to integrate virtual data and request-planning mechanisms into the IMPALA/BOSS production system of CMS simulated data. As part of this effort, several job management mechanisms need to be brought together into a single coherent entity. This will be accomplished by extending IMPALA/BOSS with GriPhyN VDT components as well as prototype virtual data technologies. Valuable research feedback to the GriPhyN project is anticipated as VDT enhanced versions of IMPALA/BOSS are evaluated by CMS. During Year 2 and beyond, we intend to integrate several GriPhyN Grid technologies into the IMPALA/BOSS job management, including: replication, replica location service, reliable file transfer, the Community Authorization Service, and prototypes of the GriPhyN Job Execution Planner.

The benefits to CMS of this effort represent increased functionality: easier, more automated job submission; easier recalculation of derived data products; accurate tracking of data derivation; ability to re-derive latter stage outputs of the production data pipeline without re-calculating the earlier phases, in cases where the later-stage processing programs require changes. We intend to highlight ease of grid usage as a major benefit to CMS, especially in the automated handling of failures and complex grid configuration and usage details. We also intend to explore how to effectively utilize the GDMP publish-subscribe paradigm within CMS production.

The benefits to GriPhyN of this effort include: live testing of a fundamental GriPhyN concept in intensive production for real users; user feedback on the value of the virtual data paradigm and usability of the tool

set; measurement of virtual data process effectiveness; capture of live logs of the detailed activity and resource utilizations of the production process for further CS research.

2.2.2 Analysis of CMS Data

This activity explores the vital later-phase of CMS data analysis. Such analyses search massive numbers of events for signature patterns offering evidence for various proposed theories of nature. Once CMS is online and recording live data from the detector, analysis activities will be the dominant use of computing resources.

In a typical analysis, a physicist selects events from a large database consisting of small sized event classification information, or TAG data. Using the TAG data, the physicist gathers the full reconstructed event from various sources including, if necessary the creation of fully reconstructed versions of those events if they do not already exist on some "convenient" storage system. A typical analysis might look something like the following:

- Search through 10^9 TAG events (~100 GB), select 10^6 of those events and fetch the full "raw" event data for the selected events (~1 TB)
- Invoke a user defined reconstruction of the full "raw" event data and make a user defined set of Analysis Object Data (~100 GB) and TAGs (~100 MB)
- Analyze the user defined Analysis Object Data (AOD) and TAG datasets interactively, extracting a few hundred candidate signal events (~10 MB).
- Histogram the results and visualize some of the "interesting" events.

We seek to create virtual data techniques to track data dependencies for the files and/or objects in this process from TAG schemas and TAG databases (or tables) back to the reconstructed event sets and possibly back to the raw data.

We will build tools for this type of user-driven fine-granularity physics dataset extraction and transport over the grid, driven by an easy-to-understand interface that reduces the difficulties of marshalling distributed grid resources. This effort will exploit newer collection management features developed by ORCA/CARF team at CERN.

We expect to further explore the impact of workloads from end-user physics data analysis on the grid system, by prototyping distributed end user analysis tools, demos, and pilot facilities.

2.2.3 Test Beds

To facilitate the research, development, and integration of virtual data technology into the CMS software environment, several potential grids may be used for CMS GriPhyN Year 2 activities:

- GriPhyN Test Grid: an envisaged, shared GriPhyN experimental grid meant to be used by all experiments, for the initial stages of prototyping.
- US-CMS Test Grid: a CMS-only, experimental test bed (currently under construction), to be used by CMS GriPhyN and CMS PPDG collaborators to prototype tools for CMS production of simulated data and distributed analysis.
- CMS Data Grid: a CMS-only test bed (currently under construction) to be used by CMS collaborators to prototype tools for CMS production of simulated data and distributed analysis.

One of the early goals will be to integrate the US-CMS Test Grid with the CMS Data Grid. This will allow the CMS GriPhyN team to remain synchronized and integrated with future CMS developments.

3 Development Tasks

The primary focus of CMS activities in GriPhyN during Year 2 will be the development of virtual data tools for the production of Monte Carlo simulated CMS data and virtual data tools for the analysis of CMS data. These software packages will rely both upon existing tools from the current Virtual Data Toolkit as well as contribute new tools, which are general enough in nature, into future versions of the Virtual Data Toolkit.

In support of these activities, efforts will be directed towards the establishment of a catalog infrastructure including: Replica Catalogs (RC), Virtual Data Catalogs (VDC), and Meta-Data Catalogs (MDC). Significant progress towards the development of a prototype VDC has already been accomplished by Voeckler using a PostGreSQL database coupled with a Perl interface. The catalog tracks the dependencies of data files and transformations between data files. As such, it is able to regenerate any (missing or deleted) derived data file on demand.

While CMS does not currently use virtual data concepts, CMS has detailed several future needs related to virtual data [GRIPHYN 2001-16]. Using the experience gained by integrating the VDC with current CMS production and analysis needs (see below), CMS-GriPhyN plans to work with CS-GriPhyN to further develop virtual data concepts by taking the following as work items during Year 2:

- Work Item 1: Develop and prototype the concept of "grid uploaded" files and algorithms (i.e. transformations between data files) by leveraging existing technology in GDMP and extending it. Such files and transformations would exist in a grid wide replica catalog and be distinguished by Unique Identifiers (UID).
- Work Item 2: Interface the current VDC with GDMP and an extended GDMP so that virtual data products can be materialized from "grid uploaded" files and transformations. Each materialized virtual data product would receive a UID and entry into the "replica catalog" and/or the VDC. Platform dependencies and their relation to virtual data product UIDs will be investigated.
- Work Item 3: Develop prototypes for Consistency Management of the replica catalog over a grid.

3.1.1 Tools for the Production of Monte Carlo Simulated CMS Data

A tool for the distributed production of CMS Monte Carlo simulated data, known as MOP, is currently under development from the Particle Physics Data Grid. MOP (which is based upon Globus, Condor-G, and DAGMan) is loosely integrated with a set of shell scripts, known as IMPALA, that are currently used by CMS for Monte Carlo production as well as the Grid Data Movement Package, or GDMP. Currently these tools do not employ virtual data concepts.

Over the course of Year 2, CMS-GriPhyN will augment MOP and IMPALA with virtual data tracking using the Virtual Data Catalog. This will involve decomposing the job submission and bookkeeping logic of IMPALA into parameters and transformations which are specific to CMS and logic which is more general to batch job execution planning in the form of abstract Directed Acyclic Graphs (DAGs). In addition, the distributed job execution logic of MOP will be embedded into the VDC to facilitate virtual data materialization in a grid environment. In order to fine tune these concepts and synchronize with CMS production efforts, two challenge problems a proposed over the next year:

- Challenge Problem 1: Produce 50,000 Monte Carlo fully simulated CMS events (including pileup) using the VDC on a USCMS Test Grid (see below). This should expose any technical and concept ional modifications required to use the VDC in realistic situations.
- Challenge Problem 2: Fulfill one (or several) official Monte Carlo Production request(s) from CERN on a USCMS Test Grid. This will demonstrate the feasibility of using the VDC in "real world" CMS production activities.

The aim of this effort is two-fold: 1) provide an ever more autonomous environment for CMS Monte Carlo production by enabling automatic error recovery, rigorous bookkeeping, and transparent production at different CMS grid sites and 2) provide valuable insight into virtual data concepts for future prototyping.

3.1.2 Prototypes for the Analysis of CMS Data in a Remote Environment

Virtual data as it applies to scientific analysis of CMS data has only recently been considered. It is currently unclear whether CMS physicists will employ a single monolithic analysis tool, or use a standardized set of analysis tools, or even use different sets of analysis tools. As a result, CMS research into different data analysis paradigms will continue to be monitored by CMS-GriPhyN throughout Year 2.

Given that CMS requires that physicists have the option of defining their own sets of data products (files, objects, etc) for scientific analysis, it important to begin the process of tracking virtual data products as they relate to end-user data analysis. In order to probe this and to facilitate whatever analysis tool(s) that

may be used in the future by CMS, a remote data server (known as Clarens) is being developed by Steenberg to enable analysis of CMS data distributed over a wide-area network. Clarens is based on a Client/Server approach and provides a framework for remote data analysis. Communication between the client and server is conducted via XML-RPC over a scalable SSL-encrypted HTTP transport provided by the Apache web server. Currently the server is only linked to the standard CMS analysis C++ libraries, but various server functions are envisioned. The client can be end-user specific and implementations currently exist for several data analysis tools including: C++, Python (for use in the Lizard analysis environment), PHP (a web portal), The Java Analysis Studio, and a Python plug-in for SciGraphica. This allows the user full access to remote CMS data via a choice of analysis packages as well as the web.

During Year 2, Steenberg plans to grid-enable Clarens by taking full advantage of the Virtual Data Toolkit including: the Virtual Data Catalog for tracking CMS data, the Globus Security Infrastructure for authentication, and Grid-ftp for CMS data movement:

- Work Item 4: Provide a remote interface to a useful subset of VDT 1.0 functionality, including authentication, file movement, and job scheduling and monitoring.
- Work Item 5: Provide a remote interface to the VDC.

As a joint endeavor with efforts in the virtual data tracking of CMS Monte Carlo production, the following data challenge problem is proposed:

- Challenge Problem 3: Remotely analyze 50,000 events using Clarens integrated with the VDC as used in Challenge Problem 1.

Finally, investigations into more fine grained data collections at the object level and their relation to a VDC will also be done during Year 2.

4 Infrastructure Tasks

CMS GriPhyN is currently building a US-CMS Test Grid, in cooperation with the Particle Physics Data Grid, at five initial sites: The California Institute of Technology, Fermi National Accelerator Laboratory, The University of California-San Diego, The University of Florida, and The University of Wisconsin-Madison. The US-CMS Test Grid is expected to be operational in January 2002.

The initial goals of the test grid will be to produce a platform, which enables CMS grid software development, and which probes policy issues related to Certificate Authorities. In addition, early integration with the CMS Data Grid (also expected to come online in January 2002) will be aggressively pursued. This will facilitate a closer technical cooperation between CMS and GriPhyN.

4.1.1 Software Components

To ensure interoperability with other GriPhyN grid efforts, the US-CMS Test Grid is based upon the GriPhyN Virtual Data Toolkit Version 1.0, which includes Condor 6.3.1, Globus 2.0 beta, and GDMP 2.0. In addition to the packages contained in the Virtual Data Toolkit, Condor-G 6.3.1 and Objectivity/DB will be installed.

- Work Item 6: Install and verify the Virtual Data Toolkit 1.0 and Condor-G 6.3.1 at each grid site.
- Work Item 7: Install Objectivity 6.1 at each grid site

The entire suite of CMS software is complex with many external software package dependencies and dynamically linked shared object libraries. As a result, a CMS software Distribution After Release (DAR) package has been developed at Fermilab. DAR files are essentially "tar balls" consisting of particular versions of CMS executables packaged with the required CMS execution environment (shared object libraries, etc). It is planned to distribute and install DAR files to each grid site, enabling grid users to have access to several versions of CMS software.

- Work Item 8: Install an initially agreed upon CMS DAR file at each grid site

To facilitate sophisticated CMS software development, dynamic installation of personalized CMS software will be investigated and implemented via DAR files.

- Work Item 9: Investigate "on-the-fly" DAR file installation via a remote user.

4.1.2 Platform Requirements

Currently, the CMS software environment supports the Red Hat 6.1, 6.2 and Solaris operating systems. To ensure compatibility with the CMS software environment as well as the VDT 1.0, the test bed will be entirely composed of machines running Red Hat 6.2.

4.1.3 Security and Resource Sharing Policies

Each CMS-GriPhyN and CMS-PPDG registered user will receive an account at each of the five grid sites. Currently, the account management for the US-CMS Test Grid is de-centralized. During the course of Year 2, the CMS GriPhyN team will monitor developments in the European Data Grid Project regarding account management.

The test bed will initially use the Globus Certificate Authority for the distribution of user and gatekeeper authentication certificates. However, as ESnet is expected to provide a Certificate Authority in April 2002, the US-CMS Test Grid will migrate to the ESnet Certificate Authority when it becomes available.

- Work Item 10: Monitor the progress of the ESnet Certificate Authority and migrate to ESnet issued GSI authentication certificates as they becomes available

A mechanism for centralized accounting of resource sharing and authorization has not been addressed. The CMS GriPhyN team will monitor progress with the Community Authorization Service (CAS), currently being developed by Globus. However, CAS is considered a long-term solution. In the absence of such a mechanism, resource sharing and accounting procedures will be studied and implemented on a site-by-site basis.

4.1.4 Integration with other Grid Projects

The integration of the US-CMS Test Grid with other Grid Projects of the same scope is considered important. In particular, preliminary dialog has already begun with the CMS Grid Integration Task Group concerning how best to integrate the CMS Data Grid and the US-CMS Test Grid. This will provide a higher likelihood that tools developed by CMS GriPhyN will more directly benefit and be accepted by CMS.

- Work Item 11: Determine and resolve any possible conflicts between the US-CMS Test Grid and the CMS Data Grid.

5 Conclusions

The CMS GriPhyN activities in Year 2 will involve three primary research areas: (1) development and integration of virtual data technology into the production of CMS simulated data, (2) research and development of virtual data functionality in remote analysis of CMS data, and (3) development of the necessary test bed infrastructure needed to develop, test, and integrate virtual data into the CMS grid environment.

A great deal of effort has already been invested in understanding the current and future virtual data needs of CMS. This investigative effort will continue during Year 2 so that GriPhyN and CMS remain synchronized with respect to milestones and software architecture. Nevertheless, additional liaison efforts between CMS and GriPhyN are required to ensure that virtual data technologies developed by GriPhyN are not only found to be useful in CMS, but actually integrated into CMS grid technology as well. The integration of the US-CMS Test Grid and the CMS Data Grid will assist this effort.

6 Appendix: Snap-shot of current Year 2 Work Plan (subject to change)

1	CS work on Distributed Virtual Data Catalogs
1.1	Develop Replica Catalog
1.1.1	Investigate concepts of "uploaded" files and algorithms
1.1.2	Prototype a Grid-wide RC which catalogs files and algorithms
1.1.3	Investigate Consistency Management Techniques with respect to file existence
1.2	Integrate the VDC/L with the RC
1.3	Integrate VDC with mop_submitter
2	Virtualize CMS Distributed Monte Carlo Production
2.1	Integrate CMS Monte Carlo Production Pipeline with the VDC/L
2.1.1	Investigate "on-the-fly" creation of Objectivity database files and how to catalog them in the VDC
2.1.2	Integrate VDC directly with DAR (tar ball of the CMS software and environment)
2.1.2.1	Understand the architecture of DAR
2.1.2.2	Produce a VDL file for a DAR pipeline
2.1.3	Include Pile-up in VDL file for Monte Carlo Production
2.1.4	Start a test Monte Carlo Production using the VDC with no validation checks
2.1.5	Produce 10 000 events using the VDC
2.1.6	Validation of Monte Carlo Simulated Data
2.1.6.1	Understand how to validate a produced data file
2.1.6.2	Determine additional steps in the production pipeline to validate each produced data file
2.1.6.3	Integrate additional validation steps into the production pipeline
2.1.7	Start actual Monte Carlo Production using the VDC with automatic validation checks
2.1.8	Produce 50 000 events using the VDC
2.2	Develop VAMPALLA: Virtual dAta iMPALLA
2.2.1	Develop Test Version of a Virtual Data IMPALA
2.2.1.1	Understand the architecture of IMPALA
2.2.1.2	Understand the architecture of MC_Runjob
2.2.1.3	Understand the architecture of BOSS
2.2.1.4	Determine in which initial layer the VDC should be integrated
2.2.1.5	Write interface between the IMPALA integration Layer and the VDC
2.2.1.6	Deploy Test version of VD-IMPALA for Monte Carlo Production
2.2.1.7	Receive Comments
2.2.1.8	Determine a more optimal architecture for a Virtual Data IMPALA

2.2.2 Virtual Data IMPALA Version
2.2.2.1 Virtual Data IMPALA Version 1
2.2.2.2 Virtual Data IMPALA Version 2
2.2.2.3 Virtual Data IMPALA Version 3
2.2.2.4 Virtual Data IMPALA Version 4
2.2.2.5 Virtual Data IMPALA Version 5
2.2.2.6 Virtual Data IMPALA Version 6
2.2.2.7 Virtual Data IMPALA Version 7
2.2.2.8 Virtual Data IMPALA Version 8
2.2.2.9 Virtual Data IMPALA Version 9
2.2.2.10 Virtual Data IMPALA Version 10
2.2.2.11 Virtual Data IMPALA Version 11
2.2.2.12 Virtual Data IMPALA Version 12
2.2.2.13 Virtual Data IMPALA Version 13
3 Virtualize CMS Remote Data Analysis
3.1 Integrate Clarens with VDT 1.0
3.1.1 Authenticate connection using GSI
3.1.2 Demonstrate GSI authentication at GriPhyN CS-meeting
3.1.3 Expose the GDMP C++ API to remote clients
3.1.4 Demonstrate text version of GDMP C++ API to remote clients
3.1.5 Demonstrate GUI version of GDMP C++ API to remote clients
3.1.6 Expose the Globus Toolkit API to remote clients
3.1.7 Demonstrate text version of Globus API to remote clients
3.2 Integrate Clarens with VDC/L
3.3 Remotely analyze 50 000 events using Clarens, the VDC and the VDT
3.4 Release Clarens 1.0 beta
3.5 Receive feedback on Clarens 1.0 beta from users
3.6 Interface Clarens with Root
3.6.1 Provide a Clarens data server for the Root I/O Package
3.6.2 Provide a Clarens data client for the Root Analysis Package
4 US CMS Test Beds
4.1 Establish Test Grid (based on VDT 1.0 and Globus CA)
4.1.1 Agree on an Initial Account Management Policy
4.1.2 Agree on Software Base
4.1.3 Commission Caltech Site

4.1.3.1 Identify Machines for US CMS Test Bed

4.1.3.2 Provide initial user accounts

4.1.3.3 Deploy Software

4.1.3.3.1 Install VDT 1.0

4.1.3.3.1.1 Condor 6.3.1

4.1.3.3.1.2 ClassAds 0.9

4.1.3.3.1.3 DAGMan 6.3.1

4.1.3.3.1.4 Globus 2.0 beta

4.1.3.3.1.5 GDMP 2.0

4.1.3.3.2 Install Condor-G 6.3.1

4.1.3.3.3 Install Objectivity 6.1

4.1.3.3.4 Install initial version of DAR

4.1.4 Commission Fermilab Site

4.1.4.1 Provide initial user accounts

4.1.4.2 Deploy Software

4.1.4.2.1 Deploy VDT 1.0

4.1.4.2.1.1 Condor 6.3.1

4.1.4.2.1.2 ClassAds 0.9

4.1.4.2.1.3 DAGMan 6.3.1

4.1.4.2.1.4 Globus 2.0 beta

4.1.4.2.1.5 GDMP 2.0

4.1.4.2.2 Deploy Condor-G 6.3.1

4.1.4.2.3 Deploy Objectivity 6.1

4.1.4.2.4 Deploy initial version of DAR

4.1.5 Commission UCSD Site

4.1.5.1 Provide initial user accounts

4.1.5.2 Deploy Software

4.1.5.2.1 Deploy VDT 1.0

4.1.5.2.1.1 Condor 6.3.1

4.1.5.2.1.2 ClassAds 0.9

4.1.5.2.1.3 DAGMan 6.3.1

4.1.5.2.1.4 Globus 2.0 beta

4.1.5.2.1.5 GDMP 2.0

4.1.5.2.2 Deploy Condor-G 6.3.1

4.1.5.2.3 Deploy Objectivity 6.1

4.1.5.2.4 Deploy initial version of DAR

4.1.6 Commission Florida Site

4.1.6.1 Provide initial user accounts

4.1.6.2 Deploy Software

4.1.6.2.1 Deploy VDT 1.0

4.1.6.2.1.1 Condor 6.3.1

4.1.6.2.1.2 ClassAds 0.9

4.1.6.2.1.3 DAGMan 6.3.1

4.1.6.2.1.4 Globus 2.0 beta

4.1.6.2.1.5 GDMP 2.0

4.1.6.2.2 Deploy Condor-G 6.3.1

4.1.6.2.3 Deploy Objectivity 6.1

4.1.6.2.4 Deploy initial version of DAR

4.1.7 Commission Wisconsin Site

4.1.7.1 Provide initial user accounts

4.1.7.2 Deploy Software

4.1.7.2.1 Deploy VDT 1.0

4.1.7.2.1.1 Condor 6.3.1

4.1.7.2.1.2 ClassAds 0.9

4.1.7.2.1.3 DAGMan 6.3.1

4.1.7.2.1.4 Globus 2.0 beta

4.1.7.2.1.5 GDMP 2.0

4.1.7.2.2 Deploy Condor-G 6.3.1

4.1.7.2.3 Deploy Objectivity 6.1

4.1.7.2.4 Deploy initial version of DAR

4.1.8 Verify inter-site Test Grid functionality

4.1.9 Deploy version of the SC2001 MOP demo

4.1.10 Deploy version of the SC2001 VDC demo

4.2 Investigate currently available technologies for centralized New Account Registration

4.3 Integrate Test Grid with CMS Data Grid

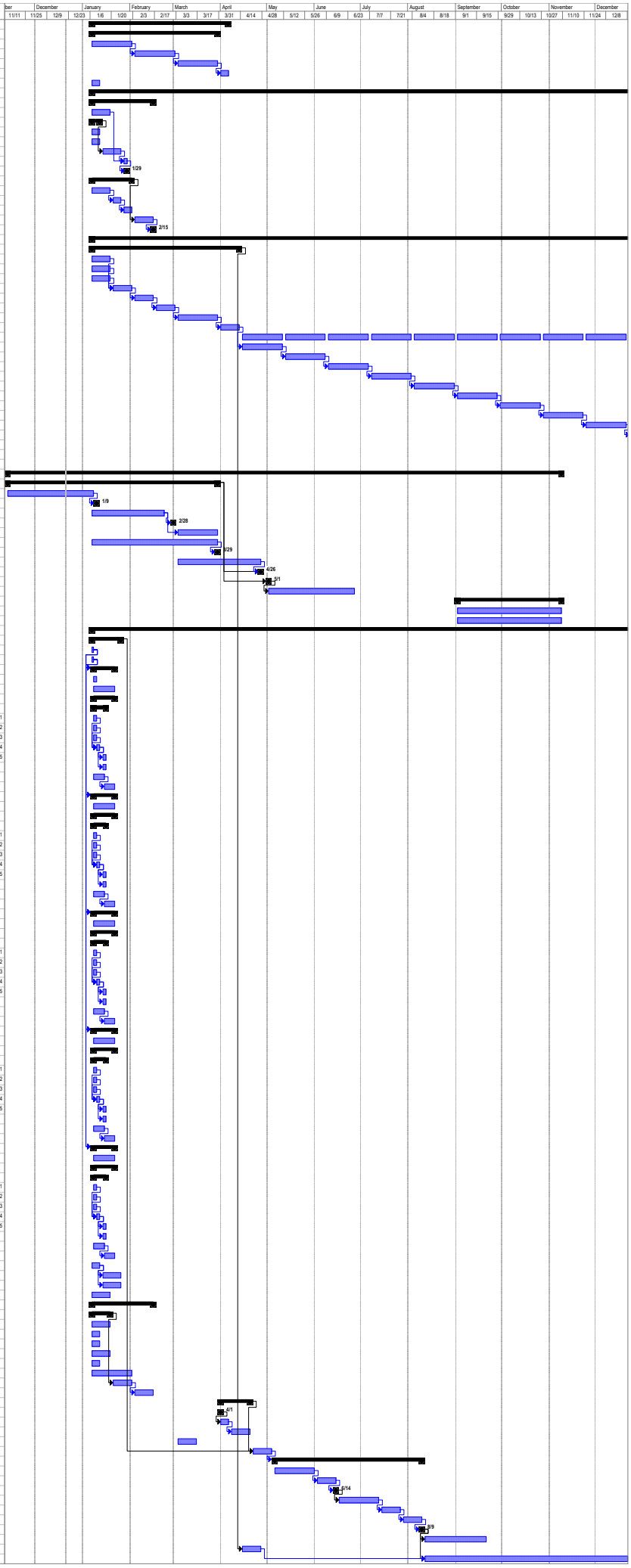
4.3.1 Determine any infrastructure requirements and resolve possible conflicts

4.3.1.1 Resolve any Certificate Authority issues

4.3.1.2 Agree to use a common version of Globus and Condor

4.3.1.3 Determine if AFS is required as a shared file system

4.3.1.4 Detail any required software

4.3.2	Agree on an Account Management Policy
4.3.3	Implement any required compatible infrastructure
4.3.4	Create user accounts according to agreed policy
4.3.5	Verify Inter-grid test bed functionality
4.4	Upgrade Test Grid to use the ESnet Certificate Authority
4.4.1	ESNet CA comes online
4.4.2	Replace the globus-gatekeeper certificate at each site
4.4.3	Each user submits a new grid-mapfile entry to each site
4.5	Upgrade Test Grid to VDT 2.0
4.6	Perform a CERN request for Monte Carlo Production using MOP
4.7	Establish Production Grid (based on VDT 2.0 and ESNet CA)
4.7.1	Develop US-CMS Production Grid Plan, Version 1
4.7.2	Circulate US-CMS Production Grid plan for feed back
4.7.3	US-CMS Production Grid Plan approved
4.7.4	Allocate machines and/or purchase machines for US-CMS Production Grid
4.7.5	Install VDT 1.0 on US-CMS Production Grid machines
4.7.6	Verify US-CMS Production Grid functionality
4.7.7	US-CMS Production Grid Operational
4.8	Commence regular Monte Carlo Production using IMPALA/MOP on US-CMS Production Grid
4.9	Perform a CERN request for Monte Carlo Production using a Virtual Data IMPALA on Test Grid
4.10	Commence regular Monte Carlo Production using Virtual Data IMPALA

ID	WBS
1	1
2	1.1
3	1.1.1
4	1.1.2
5	1.1.3
6	1.2
7	1.3
8	2
9	2.1
10	2.1.1
11	2.1.2
12	2.1.2.1
13	2.1.2.2
14	2.1.3
15	2.1.4
16	2.1.5
17	2.1.6
18	2.1.6.1
19	2.1.6.2
20	2.1.6.3
21	2.1.7
22	2.1.8
23	2.2
24	2.2.1
25	2.2.1.1
26	2.2.1.2
27	2.2.1.3
28	2.2.1.4
29	2.2.1.5
30	2.2.1.6
31	2.2.1.7
32	2.2.1.8
33	2.2.2
34	2.2.2.1
35	2.2.2.2
36	2.2.2.3
37	2.2.2.4
38	2.2.2.5
39	2.2.2.6
40	2.2.2.7
41	2.2.2.8
42	2.2.2.9
43	2.2.2.10
44	2.2.2.11
45	2.2.2.12
46	2.2.2.13
47	3
48	3.1
49	3.1.1
50	3.1.2
51	3.1.3
52	3.1.4
53	3.1.5
54	3.1.6
55	3.1.7
56	3.2
57	3.3
58	3.4
59	3.5
60	3.6
61	3.6.1
62	3.6.2
63	4
64	4.1
65	4.1.1
66	4.1.2
67	4.1.3
68	4.1.3.1
69	4.1.3.2
70	4.1.3.3
71	4.1.3.3.1
72	4.1.3.3.1.1
73	4.1.3.3.1.2
74	4.1.3.3.1.3
75	4.1.3.3.1.4
76	4.1.3.3.1.5
77	4.1.3.3.2
78	4.1.3.3.3
79	4.1.3.3.4
80	4.1.4
81	4.1.4.1
82	4.1.4.2
83	4.1.4.2.1
84	4.1.4.2.1.1
85	4.1.4.2.1.2
86	4.1.4.2.1.3
87	4.1.4.2.1.4
88	4.1.4.2.1.5
89	4.1.4.2.2
90	4.1.4.2.3
91	4.1.4.2.4
92	4.1.5
53	4.1.5.1
94	4.1.5.2
95	4.1.5.2.1
96	4.1.5.2.1.1
97	4.1.5.2.1.2
98	4.1.5.2.1.3
99	4.1.5.2.1.4
100	4.1.5.2.1.5
101	4.1.5.2.2
102	4.1.5.2.3
103	4.1.5.2.4
104	4.1.6
105	4.1.6.1
106	4.1.6.2
107	4.1.6.2.1
108	4.1.6.2.1.1
109	4.1.6.2.1.2
110	4.1.6.2.1.3
111	4.1.6.2.1.4
112	4.1.6.2.1.5
113	4.1.6.2.2
114	4.1.6.2.3
115	4.1.6.2.4
116	4.1.7
117	4.1.7.1
118	4.1.7.2
119	4.1.7.2.1
120	4.1.7.2.1.1
121	4.1.7.2.1.2
122	4.1.7.2.1.3
123	4.1.7.2.1.4
124	4.1.7.2.1.5
125	4.1.7.2.2
126	4.1.7.2.3
127	4.1.7.2.4
128	4.1.8
129	4.1.9
130	4.1.10
131	4.2
132	4.3
133	4.3.1
134	4.3.1.1
135	4.3.1.2
136	4.3.1.3
137	4.3.1.4
138	4.3.2
139	4.3.3
140	4.3.4
141	4.3.5
142	4.4
143	4.4.1
144	4.4.2
145	4.4.3
146	4.5
147	4.6
148	4.7
149	4.7.1
150	4.7.2
151	4.7.3
152	4.7.4
153	4.7.5
154	4.7.6
155	4.7.7
156	4.8
157	4.9
158	4.10