



UNIVERSIDAD MICHOACANA DE
SAN NICOLÁS DE HIDALGO



FACULTAD DE CIENCIAS FÍSICO MATEMÁTICAS

“MAT. LUIS MANUEL RIVERA GUTIÉRREZ”

*“Desarrollo e Instalación de un Centro TIER-3
para Procesamiento de Datos del LHC”*

TESIS

**QUE PARA OBTENER EL TÍTULO DE:
LICENCIADO EN CIENCIAS FÍSICO MATEMÁTICAS**

PRESENTA

Heber Orlando Arteaga Chávez

ASESOR

**Doctor en Ciencias con Especialidad en Física
Iván Heredia de la Cruz**

MORELIA, MICHOACÁN, AGOSTO 2014

Índice General

Dedicatoria	5
Agradecimientos	6
Resumen	7
Abstract	7
Capítulo 1	8
Fundamentos del cómputo GRID	8
1.1 Introducción	8
1.2 Antecedentes	9
1.3 Clúster.....	10
1.4 Mallas Computacionales o GRID	11
Capítulo 2	14
Antecedentes Históricos del CERN y el LHC	14
2.1 Antecedentes	14
2.2 CERN.....	14
2.3 El Gran Colisionador de Hadrones.....	16
2.4 Diseño del LHC.....	17
2.4.1 Detectores de Partículas del LHC	18
2.5 Compact Muon Solenoid (CMS).....	19
2.5.1 Capas que Conforman el CMS.....	19
2.5.2 Objetivos y Funcionamiento del CMS	21
Capítulo 3	23
Infraestructura de la GRID	23
3.2 La GRID en el experimento CMS	23
3.3 Modelo Jerárquico Implementado en CMS.....	24
3.3.1 Centro TIER-0.....	24
3.3.2 Centro TIER-1.....	25
3.3.3 Centros TIER-2	26
3.3.4 Centros TIER-3.....	27
3.4 Infraestructura y Servicios de OSG.....	27
3.4.1 Flujo de Trabajo.....	29

3.5 Seguridad de la GRID	31
3.5.1 Seguridad GRID en la Infraestructura Pilot	31
3.5.2 Seguridad Pilot-Based con Glexec.....	33
3.5.3 Certificados para Usuarios de la GRID	34
Capítulo 4	37
Instalación de la GRID.....	37
4.1 Repositorios OSG y EPEL	37
4.1.1 Instalación de los Repositorios de la Autoridad Certificadora.....	38
4.2 Instalación del Cliente Nodo de Trabajo (Worker Node Client) (WN)	39
4.3 Instalación de gLExec	40
4.3.1 Configuración de gLExec.....	40
4.4 Requerimientos para la Instalación del Computin Element (CE)	42
4.4.1 Instalación del Administrador de Recursos Torque	42
4.4.2 Instalación del CE	43
4.4.3 Detalles del Cliente OSG	44
4.5 Instalación del Storage Element (SE)	45
4.6 Validando el CE	46
4.6.1 Verificar que las VO son compatibles	49
4.6.2 Verificar que las todas las cuentas de UNIX fueron creadas y son accesibles....	49
4.7 Problemas en la Implementación.....	50
Capítulo 5	52
Instalación del Software de CMS	52
5.1 Sistema de Gestión de Trabajos de CMS	52
5.2 Instalación.....	52
5.3 Instalación de CRAB.....	54
5.3.1 Comandos Básicos	54
Capítulo 6	56
Análisis de Datos.....	56
6.1 Antecedentes del Bosón Z.....	56
6.2 Análisis de Datos CMS para Obtener la Distribución de Masa del Bosón Z.....	57
6.2.2 Herramientas de Análisis de Datos.....	58
Conclusiones	66

Referencias	68
--------------------------	-----------

Dedicatoria

Dedico de manera especial a mi madre pues ella fue el principal cimiento para la construcción de mi vida profesional, sentó en mí las bases de responsabilidad y deseos de superación. Agradezco a mis hermanos que estuvieron conmigo en las buenas y en las malas de manera incondicional.

Tuve la dicha de tener el mejor grupo de compañeros. Recuerdo el primer día de clases no conocía a ninguna persona del aula, y me di cuenta que eso no importaba que nosotros mismos somos los que vamos formando lazos de amistad día con día, al final de un examen y al aprobar juntos cada semestre.

Agradecimientos

Agradezco de manera especial a mis profesores por preparar sus clases con tanto entusiasmo. Sin duda ellos lograron cada semestre dejarnos inmersos en un mundo nuevo de intuición y deducción. Gracias a su apoyo incondicional pude superar los momentos más difíciles de mi vida. Cada profesor era distinto pero de alguna manera siempre lograban transmitir ánimo y seguridad para afrontar cualquier problema que se presentara.

Quiero agradecer especialmente al Dr. Iván Heredia de la Cruz por darme la oportunidad de trabajar con él, además por paciencia que me tuvo en la elaboración de este trabajo, por el apoyo moral y académico que me brindo desde el momento que lo conocí. Fue un verdadero gusto compartir con él esta etapa de mi vida.

Me siento especial en tener como sinodales a un excelente grupo de profesores de los cuales seguiré aprendiendo cada día porque están presentes en mi manera de pensar. Agradezco su tiempo y paciencia por revisar y aprobar este trabajo.

Gracias a la Lic. Azucena Chávez González y a su equipo de trabajo en la secretaria académica, por todo el apoyo brindado en cada tramite y sobre todo porque siempre me han recibido con una gran sonrisa.

Resumen

Se hace una breve introducción a la arquitectura del cómputo distribuido, mostrándola como un modelo de cómputo capaz de alcanzar los objetivos de la Física de Altas Energías. Se describen los procesos de instalación, configuración y operación del modelo computacional GRID. Se desplegó la infraestructura para implementar el modelo en un centro de supercómputo. Se expone un ejemplo del análisis de datos, donde se toma una submuestra de colisiones y se determina la masa del bosón Z, todo esto para conseguir el objetivo principal: contribuir para que la Universidad Michoacana de San Nicolás de Hidalgo sea oficialmente colaborador del experimento CMS.

Abstract

A brief introduction to the architecture of the distributed computing is done, showing the as a computing model able to achieve the objectives of the High Energy Physics. Installation processes, configuration and model GRID operation are described in detail. Infrastructure was deployed to implement the model on a supercomputer center. An example of data analysis is given, where we take a subsample of collisions and the Z boson mass is determined. All to achieve the main goal: to The Universidad Michoacán de San Nicolas de Hidalgo officially contributes to the CMS experiments.

Palabras Clave: Análisis de Datos, LHC, CERN, TIER-3, CMS.

Keyword: Data Analysis, LHC, CERN, TIER-3, CMS.

Capítulo 1

Fundamentos del cómputo GRID

1.1 Introducción

En este trabajo se describe la implementación del proyecto “Medición de propiedades de hadrones-b e instalación de un centro TIER-3¹ para el análisis de datos del experimentos CMS [1]²”. Éste constituye un esfuerzo de la Universidad Michoacana de San Nicolás de Hidalgo para implementar su propia infraestructura de cómputo distribuido (GRID) [2]³, para así brindar soporte a los proyectos académicos y de investigación de la facultad de Cs. Físico Matemáticas y el Instituto de Física y Matemáticas (IFM).

La primer fase consiste en implementar el nivel de GRID en un servidor y proponer a la UMSNH como centro TIER-3, logrando el acceso a los recursos de cómputo de las demás iniciativas nacionales a través del proyecto Open Science Grid (OSG)⁴. De esta manera podremos facilitar el uso de los recursos computacionales propios y trabajar con investigadores asociados, con la posibilidad de acceder a sus recursos de cómputo para la ejecución de Trabajos o Jobs.

Para la implementación de este proyecto aprovecharemos los recursos y el apoyo por parte de OSG, quienes dan soporte a la GRID en América. En el proyecto inicial se pretende utilizar únicamente un servidor; sin embargo, en un futuro será factible integrar

¹ Los centros TIER-3 forman parte de un grupo de investigación generalmente formado por universidades o institutos de investigación.

² El **Solenoid Compacto de Muones** (en inglés *Compact Muon Solenoid*, **CMS**) es uno de los dos detectores de partículas de propósito general del Gran Colisionador de Hadrones.

Página Web: <http://cms.web.cern.ch/>

³ La computación GRID es una tecnología innovadora que permite utilizar de forma coordinada todo tipo de recursos (entre ellos cómputo, almacenamiento y aplicaciones específicas) que no están sujetos a un control centralizado

⁴ Open Science Grid (OSG): es una comunidad global científica que usan el cómputo distribuido como herramienta para la investigación.

un clúster dedicado, con el software de análisis de datos de CMS y simulación con métodos tipo Monte Carlo (MC)⁵.

Para la implementación de la GRID del CERN⁶ en la Universidad Michoacana usaremos el sistema operativo Scientific Linux 6.0 [3] así como el software distribuido por OSG quienes mantienen una infraestructura GRID destinada al desarrollo académico y de investigación.

1.2 Antecedentes

La computación distribuida es un modelo de cómputo masivo, que coordina el almacenamiento y ejecución de aplicaciones específicas. A través de este modelo se solventan las limitantes económicas y de implementación de gigantescos laboratorios de cómputo destinados a los centros de investigación. Esto facilita a los científicos la ejecución y análisis de grandes cantidades de tareas en términos de recursos de cómputo.

La computación en racimo o *clúster* tiene sus orígenes a finales de la década de los 50. Y principios de los 60. La publicación de la ley de Amdahl⁷ en 1967 muestra el sustento formal para la tecnología de multiprocesadores y *clúster*. Un clúster es un conjunto de ordenadores unidos mediante una red de alta velocidad, de tal forma que el conjunto es visto como un único ordenador, más potente que los comunes de escritorio.

Para garantizar la seguridad en una GRID es necesario desarrollar interfaces de seguridad así como métodos de autenticación y encriptación. Los certificados digitales y llaves privadas son una opción que permitirá a los usuarios acceso a una conexión segura, mediante capas intermedias (Middleware⁸).

⁵ El método **Monte Carlo** abarca una colección de técnicas que permiten obtener soluciones de problemas matemáticos o físicos por medio de pruebas aleatorias repetidas. En la práctica, las pruebas aleatorias se sustituyen por resultados de ciertos cálculos realizados con números aleatorios.

⁶ Organisation Européenne pour la Recherche Nucléaire (CERN): Es una organización europea dedicada a la investigación en física de partículas.

⁷ Gene Amdahl, "Validity of the Single Processor Approach to Achieving Large-Scale Computing Capabilities", AFIPS Conference Proceedings, (30), pp. 483-485, 1967.

⁸ Middleware es un software que asiste a una aplicación para interactuar o comunicarse con otras aplicaciones, software, redes, hardware y/o sistemas operativos

A nivel mundial existen varias iniciativas de cómputo GRID de las cuales se destacan dos de carácter científico: la World Wide LHC (WLCG⁹), y la OSG de Norte América, la cual fue creada inicialmente para soportar el análisis de los datos obtenidos por el Gran Colisionador de Hadrones. Estos proyectos utilizan gLite [4]¹⁰ y Globus [5]¹¹ como sus capas intermedias de abstracción o middlewares

1.3 Clúster

El término clúster (Derivado del inglés clúster, “grupo” o “racimo”) se aplica al conjunto de equipos de cómputo unidos mediante una red de alta velocidad que interactúan para desarrollar actividades en común. Así un clúster es visto como un único e integrado recurso de cómputo que permite desarrollar tareas que demandan una gran capacidad de recursos.

En términos generales los clúster se clasifican de la siguiente manera.

- i. **High Performance Computing Clusters (HPCC):** Clúster de Alto Rendimiento. Ejecuta tareas que requieren grandes cantidades de recurso tales como memoria, procesamiento y almacenamiento
- ii. **High Availability (HA):** Clúster con la más alta disponibilidad y confiabilidad posible. Intenta mantener en todo momento la prestación del servicio, encubriendo fallos tanto de hardware como software.
- iii. **High Throughput:** Clúster de Alto volumen especializado en tareas de alta velocidad; es decir, procesa exitosamente la mayor cantidad de tareas en el menor tiempo posible.

Un clúster requiere los siguientes componentes básicos:

⁹La Worldwide LHC Grid (WLCG) es un proyecto global donde colaboran más de 150 centros de cómputo en 40 países, con infraestructura nacional e internacional. La misión de WLCG es proveer un proyecto de recurso de cómputo global capas de analizar 25 petabytes de información anual generados por el LHC.

¹⁰ GLITE, Lightweight Middleware for Grid Computing. Página Web: <http://glite.cern.ch/>

¹¹ Globus Toolkit. Página Web: <http://www.globus.org/toolkit/>

-
- i. **Nodos:** Pueden ser simples ordenadores o estaciones de trabajo (Workstation) con sistemas multiprocesador. Se especializan según la actividad a realizar al interior del clúster. El nodo principal interactúa directamente con los usuarios para procesar sus requisiciones distribuyendo el procesamiento hacia los nodos de trabajo.
 - ii. **Red de datos de Alta velocidad:** Conecta a los nodos del clúster como a los clientes con el clúster. Comúnmente las conexiones están separadas físicamente.
 - iii. **Sistemas Operativos:** Administran los recursos de los nodos y permiten la instalación de software que conforma el clúster.
 - iv. **Middleware:** Esta capa intermedia de software permite la implementación del clúster desde el punto de vista lógico. Interactúa entre el sistema operativo y las aplicaciones distribuidas.
 - v. **Aplicaciones Distribuidas:** Software de alto nivel que hace uso del clúster gracias a la abstracción provista por el middleware, tales como Torque¹² [6] y Condor [7]¹³.

El clúster que se implementara utilizará Torque como sistema de gestor de carga de trabajo especializado para tareas de cálculo intensivo.

1.4 Mallas Computacionales o GRID

Las mallas computacionales son el siguiente nivel a implementar para lograr la abstracción de los sistemas distribuidos con relación a los clústers. La GRID pretende hacer uso de los recursos esparcidos por todo el mundo compartiendo y explotando los clúster remotos de manera transparente y confiable.

¹² Middleware Torque página Web: <http://www.adaptivecomputing.com/downloading/?file=/torque/>

¹³ Middleware Condor Página Web <http://research.cs.wisc.edu/htcondor/>

El objetivo de desarrollar mallas computacionales consiste en integrar y optimizar el uso de recursos geográficamente distribuidos para el análisis de grandes bases de datos como si se tratara de un clúster local.

Las características generales de la GRID son las siguientes:

1. **En un ambiente heterogéneo:** Es posible que los ordenadores que integran la GRID sean de distintas arquitecturas, es decir, los elementos pueden ser de distinto hardware y sistema operativo, como Windows, Solaris, Linux o MacOS.
2. **Middleware:** Es el conjunto de software y servicios que son requeridos para ofrecer a los usuarios un entorno homogéneo.
3. **Sistema de seguridad ante fallos y control de tolerancia:** Provee mecanismos para detectar y solucionar los problemas de hardware y software permitiendo que los procesos finalicen de manera segura. Por otro lado el sistema global se encarga de registrar los errores para, así mismo, solucionarlos mediante la toma de decisiones automatizadas.
4. **Alta estabilidad:** El sistema GRID puede ser alterado con la finalidad de implementar mejoras a su capacidad de recursos sin que esto afecte el desempeño global o local.
5. **Altos niveles de seguridad:** Este entorno implementa servicios de seguridad mediante el uso de certificados y llaves privadas que permitan validar la autenticación de equipos, clústers y usuarios.

Los componentes de la GRID son semejantes a los descritos a nivel de clúster, pero con mejoras que administran el control de permisos, seguridad de acceso, balanceo de cargas y comunicación ambientes heterogéneos:

- i. **Nodo Central:** Se encarga de gestionar los recursos de la GRID recibiendo solicitudes de Jobs o Trabajos para posteriormente delegar su ejecución. Este nodo comúnmente se conoce como Computer Element (CE).

-
- ii. **Nodos de Trabajo (Worker Nodes):** Se encargan de procesar los trabajos delegados por el nodo central y posteriormente reportan los resultados para que el CE avise a los usuarios.
 - iii. **Nodos de Autenticación:** Validan la identificación de los usuarios mediante los certificados y claves privadas.
 - iv. **Nodos Cliente (Client Nodes):** Permite a los usuarios enviar sus trabajos o *Jobs* (en inglés) para que se ejecuten en el nodo de la GRID.
 - v. **Nodos de Almacenamiento:** Permiten disponer la capacidad de almacenamiento y bases de datos. También se conoce como Storage Element (SE).

Cada nodo del sistema posee sus propias normas de seguridad, acceso, y control de la disponibilidad de los recursos compartidos. Todo esto debe ser transparente a los usuarios, por lo que el sistema debe funcionar como una computadora virtual que internamente sea capaz de integrar las particularidades de cada nodo y presentarse al usuario como un único sistema.

Capítulo 2

Antecedentes Históricos del CERN y el LHC

2.1 Antecedentes

El Modelo Estándar (ME) de la Física de Partículas describe a todas las partículas conocidas y a sus interacciones, excepto las gravitatorias. Entre otras varias observaciones, el origen de la masa de las partículas es una pregunta abierta.

Por otro lado el gran colisionador de Hadrones (LHC) en el CERN [8] es una de las mayores instalaciones científicas jamás construida. Ésta ayudara a los físicos a responder las preguntas fundamentales de la física de partículas, así como a revelar algunos fenómenos novedosos que pueden ir más allá del ME.

El 4 de Julio de 2012 se descubrió en el LHC una partícula que es consistente con el bosón de Higgs. Recibe este nombre en Honor a Peter Higgs, quien en conjunto con colaboradores, propusieron en 1966 el hoy famoso mecanismo de Higgs para dotar de masa a leptones, quarks y bosones de norma del ME. Estudios realizados en el 2013 confirmaron su existencia y además se midieron preliminarmente algunas de sus propiedades tales como espín, paridad, fracciones de ramificación y acoplamientos con otras partículas del ME. Continuar con la recolección de datos en los experimentos CMS y ATLAS permitirán estar completamente seguros de que el bosón encontrado es el Higgs del ME. También se buscan partículas motivadas por otras teorías como Higgs múltiples, supersimetría, dimensiones extras etc. Se prevé que el LHC volverá a operar en 2015 a su máxima potencia de diseño, 14 TeV.

2.2 CERN

CERN es el Consejo Europeo para la Investigación Nuclear. Es uno de los centros de investigación científica con mayor prestigio a nivel mundial. Su misión es la investigación

en física fundamental y en particular en física de partículas; es decir, busca explicar la composición y funcionamiento del universo.

Los instrumentos utilizados en el CERN son los aceleradores de partículas y detectores que registran las colisiones. El CERN ha realizado importantes contribuciones al desarrollo de la física de partículas. Por ejemplo en 1973 con la cámara de burbujas Gargamelle. Se observó un proceso que requería la existencia de una partícula neutra portadora de fuerza débil. De esta manera se daría el descubrimiento histórico de los bosones W y el bosón Z, los portadores de la fuerza nuclear débil. Esto confirma la teoría unificadora de las fuerzas débil y electromagnética, la “teoría electrodébil”. En los años 90s, las medidas de precisión se realizaron con el gran colisionador Electrón-Positrón (LEP), incluyendo la medición de número de las familias de neutrinos. Estos descubrimientos fueron los hitos históricos de las pruebas del ME.

En el año de 1990 Tim Berners-Lee y Robert Cailliau propusieron un sistema de información distribuida. CERN anunció que la World Wide Web (WWW) sería una herramienta gratuita, a disposición de todo el mundo. La WWW impactó las tecnologías de la información y sin duda cambió la vida de cada persona.

Para el análisis de una enorme cantidad de datos recolectados de los experimentos, el CERN cuenta con un robusto centro de cómputo con un procesamiento de datos de gran alcance, además de instalaciones de almacenamiento masivo, dedicado principalmente a la investigación. La mayor parte de las actividades del CERN está dirigida al uso de herramientas tecnológicas para el desarrollo de la física. El LHC en funcionamiento normal produce más o menos unos 25 petabytes (PB)¹⁴ de datos anuales. Para hacer frente a los grandes desafíos para el análisis de datos del LHC, existe actualmente la World Wide LHC Computing GRID (WLCG), la cual tiene como objetivo almacenar, analizar y distribuir la inmensa cantidad de datos registrados por los experimentos de LHC. La misión de la WLCG es construir una infraestructura de almacenamiento de datos y análisis para la comunidad

¹⁴ Un petabyte es una unidad de almacenamiento de información cuyo símbolo es el PB, y equivale a 10^{15} bytes.

de la física de altas energías. En la actualidad, la WLCG combina los recursos informáticos de más de 100,000 procesadores de más de 170 sitios en 34 países. Gracias a la WLCG más de 8000 físicos de todo el mundo accedan en tiempo real a los datos de LHC para procesarlos. El concepto del WLCG u OSG para América, será abordado con profundidad en el capítulo 3.

2.3 El Gran Colisionador de Hadrones

El Gran Colisionador de Hadrones (LHC) es el más grande acelerador de partículas de altas energías ubicado en la frontera Franco-Suiza. En el año 2011, el LHC logró colisionar haces de protones con una energía de centro de masa de 7 TeV. Para el 2012 el LHC alcanzó una energía de 8 TeV. Se prevé que para 2015 alcance una energía de 14 TeV. El LHC multiplica por 4 la energía y por 20 la luminosidad alcanzada por el acelerador Tevatrón del Fermilab en E.U.A.

En la figura 2.1 se detalla los cuatro puntos de interacción o colisión, los cuales se distribuyen a lo largo del anillo acelerador y en donde se ubican grandes detectores: ATLAS (A Toroidal LHC Apparatus), CMS (Compact Muon Solenoid), LHCb (A Toroidal LHC Apparatus), ALICE (A Large Ion Collider Experiment). Estos detectores registran las partículas resultantes de las colisiones para probar diversas predicciones de la física de altas energías, tales como la existencia del bosón de Higgs o de nuevas partículas predichas por la supersimetría y otras teorías que van más allá del ME.

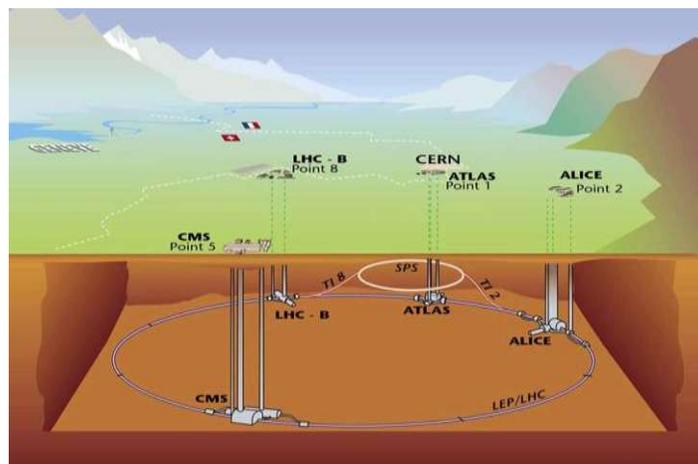


Figura 2.1: En esta figura se muestran los 4 puntos de interacción del LHC, así como el pre-acelerador SPS.

2.4 Diseño del LHC

Por razones económicas el LHC reutiliza el antiguo túnel de LEP de 27 km de circunferencia. Estas dimensiones, junto con el máximo valor del campo de los dipolos (encargados de curvar lo haces en el acelerador) y las cavidades radiofrecuentes (encargadas de acelerar los protones), limitan la energía de centro de masa de los haces de protones a 14 TeV.

Se requiere una energía muy alta pues solo una fracción de esta es realmente utilizada cuando ocurren las colisiones de los constituyentes de los protones (quarks y gluones), en los procesos que se quieren estudiar. La tabla 2.1 muestra los parámetros del diseño de LHC.

Parámetro	Valor
Energía en el centro de masas	14 TeV
Energía de inyección en el LHC	450 GeV
Numero de protones por paquete	1.15×10^{11}
Numero de paquetes en cada haz	2808
Luminosidad Nominal	$10^{34} \text{cm}^{-2} \text{s}^{-1}$
No. colisiones por segundo	600 millones
Vida media del haz (Beam Lifetime)	~10 h
Longitud de los paquetes	53mm
Radio del haz en el punto de interacción	15 μm
Tiempo entre colisiones	24.95 ns
No. de vueltas por segundo	11245
Frecuencia de cruce de haces	~31.6 MHz
Circunferencia	26,659 km
Campo magnético de los dipolos	8.3 T
Temperatura de operación de los dipolos	1.9 K

Tabla 2.1 Parámetros de diseño del LHC.

El complejo de aceleradores de CERN es una sucesión de máquinas que aceleran las partículas llevándolas a energías cada vez más altas, con velocidades cercanas a la velocidad de la luz de 299 762 458 m/s (c). La fuente de protones es una botella de gas de hidrogeno, que pasan por un campo eléctrico para despojar los de sus electrones, para así producir protones. La primera aceleración se lleva a cabo en el LINAC2, donde los protones salen con una energía de 50 MeV (33% de la velocidad de la luz). De ahí los protones pasan al Proton Synchrotron Booster, en donde además de aumentar la aceleración a 1.4 GeV (90% de la velocidad de la luz), estos son compactados en paquetes (bonches) de 100,000 millones de protones. Luego se transfieren al Proton Synchrotron (PS) donde la energía llega a 25 GeV (99.9% de la velocidad de la luz). Posteriormente los paquetes de protones son acelerados en el Súper Proton Synchrotron (SPS) donde la energía alcanza los 450 GeV (99.9998% de la velocidad de la luz). Finalmente los protones son transferidos a los dos tubos del LHC en donde la velocidad alcanza 99.9999991% de la velocidad de la luz, y la energía cinética llega a los 7 TeV. El LHC tiene unos 7 000 imanes refrigerados por helio líquido a menos 2 grados Kelvin. Los dos haces entran en colisión en cuatro puntos (ALICE, ATLAS, CMS y LHCb) donde la energía total de colisión es igual a 14 TeV.

2.4.1 Detectores de Partículas del LHC

Actualmente hay cuatro detectores:

- **ATLAS** (A Toroidal LHC Apparatus) y **CMS** (Compact Muon Solenoid):
En ambos proyectos están implicados más de 4000 científicos e ingenieros pertenecientes a 34 países. Son detectores multipropósito y sus principales objetivos son la investigación de física más allá del ME como la supersimetría, la exploración física en el rango de TeV e investigaciones relacionadas con el bosón de Higgs.
- **LHCb** (LHC Beauty Experiment): Se especializa en el estudio de física del quark botton (b). Uno de sus objetivos es la medida de parámetros de violación de

simetría de Carga-Paridad (CP), que están relacionadas con el excedente de materia sobre la antimateria en el universo.

- **ALICE** (A Large Ion Collider Experiment):

ALICE se encarga de estudiar las colisiones de iones pesados, como plomo-plomo con una energía de centro de masa de 5.5 TeV por núcleo. ALICE medirá las propiedades del plasma quark-gluón, que se cree existió en los primeros 30 milisegundos después del Big Bang.

2.5 Compact Muon Solenoid (CMS)

El CMS, es uno de los detectores de partículas de propósito general el cual tiene una forma cilíndrica con 21 metros de largo y 16 metros de diámetro. Es capaz de detectar múltiples aspectos de las colisiones de protones. Tiene un sistema que permite medir la energía y la cantidad de movimiento de fotones, electrones, muones y diversas partículas producto de las colisiones. La capa interna del detector está construida de silicio semiconductor y alrededor de esta se encuentra un calorímetro electromagnético de cristales centelladores, rodeado de un calorímetro hadrónico. El detector hadrónico y el calorímetro son tan compactos que quedan dentro del imán solenoide de CMS. Finalmente al exterior del imán, se encuentra el detector de muones.

2.5.1 Capas que Conforman el CMS

A continuación se describirá brevemente las capas de y componentes del CMS (Figura 2.2):

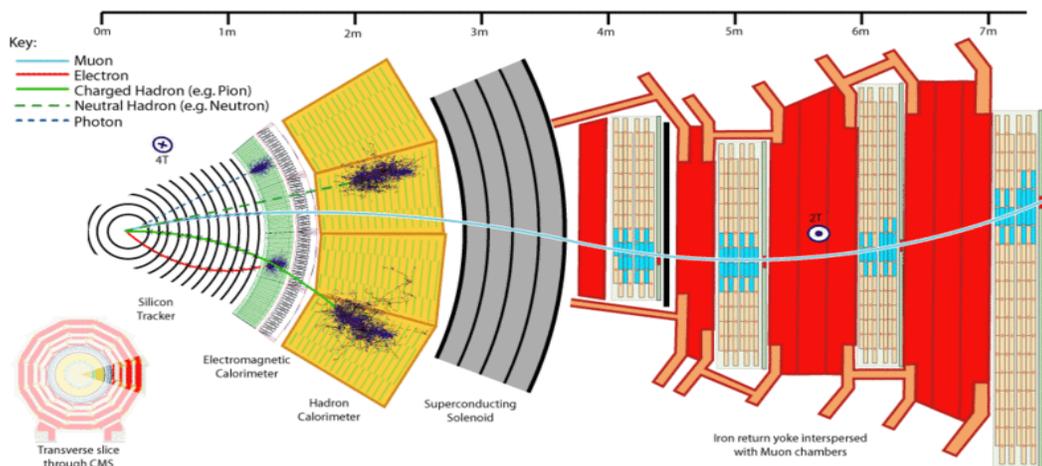


Figura 2.2 Esquema del detector de CMS.

-
- I. Región Central:** Los haces de protones colisionan en esta área. Los imanes de enfoque del LHC comprimen los haces, forzando a los protones que giran en sentido contrario a colisionar en el centro del detector. Los haces de protones se distribuyen en paquetes de 1.15×10^{11} .
- II. Tracker:** Está compuesto por finos pixeles de silicio de $125 \times 125 \mu\text{m}^2$ que permiten medir el rastro de las partículas cargadas con una resolución espacial de entre 10 a 15 μm de anchura. Se complementa además con tiras de silicio de 320 a 500 μm de anchura en regiones más alejadas del punto de interacción. El Tracker permite medir en varios casos el tiempo de vida de partículas inestables.
- III. Calorímetro Electromagnético:** Está construido por unos 80,000 cristales centelladores de tungstato de plomo (PbWO_4), que miden con precisión las energías de fotones y electrones. También tiene un detector de “prelluvia”, basado en sensores de silicio que ayudan a detectar las partículas que llegan al final de esa sección.
- IV. Calorímetro Hadrónico:** Está Formado por capas de material denso (bronce y acero) y capas de centelladores plásticos y fibras de cuarzo. Determinan la energía de los hadrones que la atraviesan, esto es, partículas como los protones, neutrones piones y kaones.
- V. Imán:** Al igual que muchos detectores de partículas, CMS tiene un gran imán solenoide. Este permite determinar el momento lineal de las partículas que lo atraviesan a partir del análisis de la curva que recorren en el seno del campo magnético. Mide 13 metros de largo y 6 de diámetro, y su núcleo superconductor de niobio-titanio está refrigerado criogénicamente con Helio líquido. Opera con un campo magnético de 3.8 T. La inductancia del imán es de 14 henrios y la

intensidad de corriente que lo atraviesa es de 19,500 A, con lo que almacena un total de 2,66 GJ, el equivalente a media tonelada de TNT.

VI. Detector de Muones: Para detectar la cantidad de movimiento de los muones, que en general son las únicas partículas que logran atravesar el imán, CMS usa tres tipos de detectores: tubos de deriva (*Drift Tubes*), cámaras de tiras catódicas (*Cathode Strip Chamber*) y cámaras de tiras resistivas (*Resistive Plate Chambers*).

2.5.2 Objetivos y Funcionamiento del CMS

El detector CMS ha sido diseñado para poder satisfacer los objetivos de física del LHC, resumidos de la siguiente manera:

- Identificación de muones: Resolución alta de la masa de di-muones y capacidad para determinar de manera contundente la carga de los muones con momento menor a 1 TeV/c
- Obtener buena resolución y eficiencia en la reconstrucción de trazas, lo que requiere de un detector de pixeles cercano a la región de interacción. Optimización de los procesos de filtrado y clasificación de los sucesos jets (chorros de partículas provenientes de un quark, gluón o leptón τ).
- Obtener buena resolución de la energía electromagnética y de la masa de di-fotones y di-electrones, cubriendo una amplia región espacial ($|\eta| < 2.5$)¹⁵. Aislamiento eficiente de fotones y leptones a altas luminosidades.
- Obtener buena resolución de la masa de di-jets y de la energía transversa faltante (E_T^{miss}), para lo que es necesario un calorímetro hadrónico hermético cubriendo una gran región espacial ($|\eta| < 5$) y con una segmentación lateral fina ($\Delta \eta \times \Delta \phi < 0.1 \times 0.1$)¹⁶

¹⁵ La pseudorapidez es una variable que se define a partir del ángulo polar θ (en donde el eje "z" se coloca en la dirección del haz de protones) como $\eta = -\ln(\text{tg}(\frac{\theta}{2}))$ y tiene la ventaja de que se transforma de forma aditiva bajo transformaciones de Lorentz a lo largo del eje z.

¹⁶ El ángulo azimutal ϕ es el ángulo en el plano transversal al haz de protones.

Las partículas producidas por las colisiones en general son inestables y se desintegran rápidamente en partículas más ligeras. Las partículas que atraviesan el detector CMS dejarán rastro de su trayectoria, permitiendo así identificarlas. Una manera de explicarlo, es como si en medio de la colisión tuviéramos una cámara fotográfica que nos permite analizar patrones y destellos de las trayectorias. Muchas de las colisiones no serán de interés porque describen partículas ya conocidas, sin embargo habrá colisiones que nos den información sobre nuevas partículas.

El detector CMS registra detalladamente una gran cantidad de sucesos y aproximadamente cada suceso ocupa 1.5 Mb/colisión. Un potente sistema de filtrado (mediante software y hardware) se encarga de seleccionar los sucesos interesantes para su análisis. El objetivo es obtener 100 de las 600 millones de colisiones que se producen cada segundo. En forma efectiva y en operación normal, el LHC mantiene colisiones durante 116 días en un año, es decir, 10^7 segundos. La producción de datos anual es de:

$$\text{Datos} = 1 \text{ año} \times 10^7 \text{ s/año} \times 100 \text{ Col./s} \times 1.5 \text{ MB/col} = 1.5 \text{ PB/año}$$

Esta es la cantidad de datos generados por el detector. Después de procesar la información y hacer respaldos de seguridad, se tendrá un total de 10 PB de datos anuales.

Capítulo 3

Infraestructura de la GRID

3.2 La GRID en el experimento CMS

La traducción de la palabra GRID al español es malla. Éste concepto es usado en redes eléctricas para referirse a líneas de corriente interconectadas entre sí. De manera que cuando encendemos un foco sabemos que esta energía proviene de distintas partes de la ciudad o del país, donde lo único que nos interesa es que haya corriente eléctrica al pulsar el interruptor.

El comentario anterior es una pequeña analogía de lo que podría representar la tecnología en cómputo GRID. Es decir que mediante una simple conexión de red, se tenga acceso a poderosos sistemas de cómputo, almacenamiento y análisis de datos. El objetivo final es que el proceso sea transparente para el usuario final tal y como nos resulta al encender una foco, en la comodidad de nuestra casa o en cualquier parte del mundo. En otras palabras, el cómputo GRID es una manera de compartir potencia de cálculo y recursos de almacenamiento a través de la red de Internet. Una red GRID debe cumplir una serie de características particulares:

- I. **Recursos compartidos:** Pongamos en contraste una GRID ante un clúster de datos. En ambos los recursos deben ser compartidos mediante una infraestructura común. La diferencia se encuentra en que el clúster pone a disposición sus recursos de manera local y la GRID a escala global.
- II. **Acceso seguro:** Hoy en día la seguridad por internet es realmente escasa, es por eso que es necesario implementar un sistema de acceso seguro mediante el uso de llaves privadas y certificados de autenticación.
- III. **Utilización de recursos:** El uso de recursos ha de ser balanceado y distribuido eficientemente entre todos los centros que componen la GRID.

-
- IV. Latencia:** La distancia entre dos centros no debe ser una desventaja para compartir datos o tareas, es por eso que se implementa tecnología de conectividad que debe ser del mismo orden de magnitud para cada uno de los centros de investigación.
 - V. Estándares abiertos:** Son necesarios para asegurar que los diferentes nodos puedan interactuar y que cualquiera puede contribuir al desarrollo de la GRID.

Como se mencionó antes, existen dos tipos de infraestructuras de GRID implementadas para cómputo científico; OSG (Open Science Grid) para America y WLCG (Worldwide LHC Computing Grid) para la Unión Europea.

3.3 Modelo Jerárquico Implementado en CMS

Considerando la necesidad de diseñar un proyecto capaz de procesar gigantescos datos de información, fue concebido el WLCG el cual es una infraestructura de cómputo que distribuye recursos de información en torno a la producción y el análisis de los experimentos en el LHC.

Este modelo fue implementado por CMS, en donde los servicios de cómputo están organizados por jerarquías. Cada centro tiene asignada varias tareas específicas dependiendo al nivel al que pertenece. El WLCG está compuesto por TIERS o NIVELES. Cada TIER está compuesta de varios centros de cómputo, los cuales proporcionan un conjunto específico de servicios tales como almacenaje, copias de seguridad, análisis y simulación de datos.

3.3.1 Centro TIER-0

La infraestructura TIER-0 [9] del CERN tiene como objetivos principales:

- Recolección de datos directamente del detector en formato RAW.
- Primera etapa de reconstrucción RECO Data.
- Brindar espacio de almacenamiento y copias de seguridad.
- Distribución de datos a los centros TIER-1.

TIER-0 es el centro de recolección de datos del CERN con capacidad de 30 petabytes y 65 000 procesadores. Toda la información obtenida en las colisiones del experimento LHC pasa a través de este complejo de servidores. CERN es responsable de proteger los datos en formato RAW (datos en crudo), procesándolos y entregando información significativa. El formato RECO (Reconstructed Data) se obtiene mediante software especializado, conteniendo así objetos físicos tales como trazas, vértices, jets y la identificación de algunas partículas. TIER-0 se encarga de distribuir los datos, reduciendo los costos de procesamiento ya que estas tareas de reconstrucción tienen un costo elevado en términos de recursos de cómputo. Full Event (FEVT) es la unión de los datos en formato RAW y RECO que son distribuidas a los centros externos.

El centro TIER-0 es responsable de distribuir los datos a los TIER-1. Por lo tanto, es necesario que exista una excelente comunicación entre las centrales. El servicio de transferencia de datos debe mantenerse entre los 5 Gb/s.

3.3.2 Centro TIER-1

Las funciones del centro TIER-1 están relacionadas con la re-reconstrucción de datos, brindando además un almacenamiento seguro. Estos centros tienen la responsabilidad de distribuir los datos a los centros TIER-2, recolectar y almacenar los datos simulados mediante Monte Carlo. De manera más detallada sus funciones principales son:

- Hacer disponible una segunda copia de los formatos RAW y RECO data.
- Forma parte del grupo de calibración.
- Elabora exhaustivas tareas de filtrado.
- Distribución de datos a los centros TIER-2.
- Recolecta de los centros TIER-2 los datos simulados mediante Monte Carlo, realiza copias de seguridad de estos y los distribuye a otros centros de análisis.

Los investigadores de CMS llevan a cabo tareas de selección y filtrado de datos obtenidos de los centros TIER-1, tomando en cuenta políticas de prioridades. Los TIERS-1 tiene la

capacidad de redistribuir los resultados obtenidos a distintos centros TIER-2 para completar los estudios de física y calibración. Es obligación de los TIER-1 proporcionar gran capacidad de cálculos y almacenamiento para el procesamiento de datos además de administrar un ancho de banda medio de 1 Gb/s para que los demás centros puedan acceder rápidamente a los datos. En la siguiente tabla (3.1) se detallan los principales centros TIER-1 del mundo.

PAIS	CENTRO TIER-1
Canadá	TRIUMF
Alemania	KIT
España	Port d'Informació Científica (PIC)
Francia	IN2P3
Italia	INFN
Países Nórdicos	Nordic Datagrid Facility
Nederland	NIKHEF / SARA
Taipéi	ASGC
Reino Unido	GridPP
USA	Fermilab-CMS
USA	BNL ATLAS

Tabla 3.1: Actualmente hay 11 centros TIER-1.

3.3.3 Centros TIER-2

Estos centros tienen las siguientes responsabilidades:

- Cubrir los requisitos de análisis de 20-100 físicos dependiendo de la capacidad del centro.
- Producción completa de Monte Carlo.
- Realizaran análisis sobre datos previamente filtrados además de calibración, alineamiento y varias operaciones relacionadas con el funcionamiento del detector.

-
- Proporcionar recursos de cómputo a distintas regiones geográficas

Los centros TIER-2 deben proporcionar capacidad de almacenamiento suficiente para guardar la cantidad de muestras replicadas de las demás TIER-2, así como de las procedentes del TIER-1 asociado. Se estima que la capacidad de almacenamiento debe ser de 200 TB de disco por centro, garantizando además una transferencia de datos de 1 Gb/s como mínimo.

3.3.4 Centros TIER-3

Los centros TIER-3 forman parte de un grupo de investigación generalmente formado por universidades o institutos de investigación. Estos centros no tienen responsabilidades específicas ya que proporcionan recursos y servicios a CMS de manera voluntaria que pueden suponer una contribución significativa a las necesidades de los experimentos.

3.4 Infraestructura y Servicios de OSG

Hoy en día la investigación consume grandes recursos de almacenamiento y procesamiento, lo cual genera la necesidad de tener una infraestructura de supercomputación. Este proyecto tiene como objetivo dar una solución a los problemas de procesamiento y análisis de datos que surgen en los diferentes grupos de investigación. De un proceso empírico y teórico la ciencia se dirige al proceso de simulación y manipulación de datos en donde son necesarias miles de horas de CPU, petabytes de almacenamiento y gigabytes por segundo en capacidad de comunicación.

Es aquí donde aparece la propuesta llamada GRID de cómputo, como respuesta a las necesidades de la ciencia de la actualidad. El software GRID consiste en una implementación de servicios que permiten acceso uniforme a los datos y recursos. Los usuarios pueden ser miembros individuales o a través grupos centrales de operación tales como Organizaciones Virtuales (VO). Son las VO las que otorgan prioridades, privilegios y, hasta en ciertos casos, la exclusividad en el uso de ciertos recursos.

Las componentes fundamentales del modelo de cómputo OSG son las siguientes:

-
- El Sistema de Gestión de Trabajos -Workload Management System (WMS).
 - El Sistema de Gestión de Datos -Data Management System (DMS).
 - El Sistema de Información -Information System (IS).
 - El Sistema de Autorización y Autenticación –Authorisation and Authentication System (AAS).
 - Varios servicios de instalación y monitorización.

El WMS administra los trabajos enviados por los usuarios, de tal manera que busca entre los recursos disponibles, aquellos que satisfacen los requisitos de análisis y simulación, y controla el proceso del envío de información. Finalmente tiene la capacidad de reportar el estado del trabajo y además permite la recuperación del trabajo terminado.

El DMS permite a los usuarios y al WMS realizar operaciones con los ficheros de datos, tales como moverlos entre distintos sitios, replicarlos y hasta eliminarlos. Estas operaciones se realizan haciendo uso de una serie de protocolos de transferencia de ficheros y consulta de bases de datos globales.

El IS proporciona información sobre los recursos disponibles en la GRID. Esta información es publicada por cada recurso y recopilada mediante un servicio de información global. El WMS hace uso de estos servicios para encontrar recursos disponibles. Así mismo el DMS hace uso del IS para localizar la disponibilidad de recursos de almacenamiento.

El AAS es responsable de la seguridad en la operación de la GRID. Mantiene una lista actualizada de los usuarios autorizados mediante certificados de seguridad, teniendo en cuenta sus posibles privilegios.

También hay disponibles una serie de recursos de monitoreo que vigilan el uso de los recursos disponibles. Esto permite a los usuarios tener información sobre sus trabajos en ejecución o simplemente para revisar el estado del trabajo.

3.4.1 Flujo de Trabajo

Para que un usuario pueda hacer uso de la GRID debe tener un certificado X-509 [10] el cual es un estándar en el sistema de criptografía de claves públicas. Este certificado es concedido por una Autoridad Certificadora (por sus siglas en ingles CA) aceptada por una organización virtual (por sus siglas en ingles VO). El objetivo de las CAs es garantizar la identidad de los usuarios comprobando así su derecho a poseer un certificado.

La sintaxis Job Description Language (JDL) nos permite enviar un conjunto de descripciones, requisitos y características del trabajo a procesar. Los ficheros que un trabajo requerirá durante su ejecución pueden ser enviados mediante la interface de usuario (UI). El conjunto de los ficheros enviados recibe el nombre de Input Sandbox. De igual manera los ficheros generados tras la ejecución del trabajo reciben el nombre de Output Sandbox.

Los siguientes servicios se ejecutan en un servidor conocido como Resource Broker (RB). Esta máquina incluye los servicios WMS y Network Server (NS) el cual acepta las peticiones que le llegan desde una UI para así autentificar y redirigir las peticiones al Workload Manager (WM).

Cuando el trabajo es enviado al WM para su ejecución, usa los servicios Matchmakin para encontrar los recursos que cumplan con los requisitos del fichero JDL. Posteriormente el WM interactúa con el IS y con un servicio que contiene el registro con la localización de los datos, conocido como Replica Location Service (RLS). La figura 3.1 muestra el esquema del flujo de trabajo.

Una vez que se han encontrado los recursos que cumplen con los requisitos especificados en JDL, el trabajo es enviado a un Computing Element (CE) para su ejecución. El CE es la interfaz de GRID con un clúster de cómputo. Está implementado en conjunto con nodos llamados Worker Nodes (WN), sobre un Sistema de Gestión de Recursos Locales –Local Resources Managment System (LRMS), y posteriormente el trabajo es enviado al GRID Gate (GG). El GG es responsable de aceptar los trabajos y distribuirlos para su ejecución. El

3.5 Seguridad de la GRID

En una infraestructura Pilot-Based los usuarios envían sus trabajos directamente a un sitio centralizado llegando a una cola de trabajo esperando para ser procesados. Este sistema es vulnerable ya que no utiliza métodos de autenticación.

La infraestructura Pilot/Gliden funciona de manera distinta. Al enviar la solicitud de trabajo, el centro remoto reconoce que hay una demanda de recursos de cómputo, entonces acepta la petición, autentifica al usuario y manda el trabajo a cola de espera para ser ejecutado. Todo este proceso requiere de una autenticación y autorización la cual se concede mediante una cuenta local Unix y las credenciales proxy del usuario.

Glexec es una herramienta implementada por OSG y la GRID [11] para brindar seguridad a la infraestructura Pilot-Based. El proceso es seguro ya que autentifica el proxy, valida la cuenta y trabajo del usuario, y de esta manera manda una autorización tal como lo haría un gatekeeper o cliente de acceso (mediante los servicios GUMS, ZAS o GRIDMAPFILE¹⁷) [12]. Además Glexec requiere de una serie de identificadores de grupo para fines de seguimiento

3.5.1 Seguridad GRID en la Infraestructura Pilot

En el modelo Pilot GRID, el cliente de acceso (GUMS, SAZ o GRIDMAPFILE) es responsable de autenticar y autorizar al usuario, convirtiéndolo en un usuario GRID y asignándole una identidad local. Cuando el Job llega al nodo de trabajo (es decir al recurso de cómputo) lo ejecuta sin olvidar que ese Job tiene asignada una identidad local. Puesto que diferentes usuarios tienen asignadas diferentes cuentas locales, los mecanismos de protección evitan que los Jobs de los usuarios entren en conflicto. (Vea la figura 3.2)

¹⁷ GRIDMAPFILE es un método de autenticación únicamente local mediante un archivo que contiene información de los certificados del usuario de la GRID.

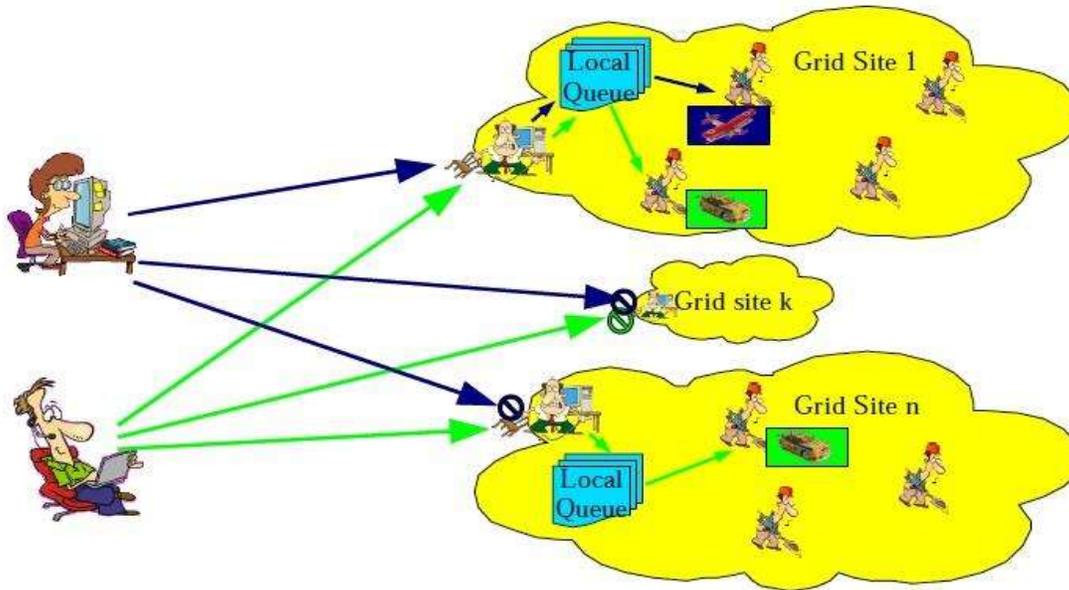


Figura 3.2 Seguridad Tradicional GRID.

Cuando se utiliza la infraestructura Pilot-Based WMS, el flujo de trabajo es diferente. Aquí los usuarios envían su Job a un sistema de lote específico al WMS. Cuando el Job acceso al WMS se activa la planta Pilot, enviando a su vez Jobs que permiten autenticar al usuario y finalmente el Job es enviado al nodo de trabajo.

Sin embargo, esta manera de operar tiene varios inconvenientes de seguridad. Las protecciones a nivel de sistema se pierden desde el punto de vista del usuario, ya que todos los usuarios tienen una identidad local piloto. Cualquier trabajo del WMS puede interferir con otros si se ejecuta en el mismo nodo de trabajo. Luego no hay protección que detenga a los usuarios mal intencionados, ya que la identidad del equipo local es la misma para el WMS. Por ende la identidad podría ser suplantada, rompiendo la seguridad en todos los sitios de la GRID.

Los problemas de seguridad mencionados anteriormente surgen por la ausencia de una herramienta que proporcione seguridad a los nodos de trabajo. Si esta herramienta existiera la infraestructura WMS sería una buena opción de seguridad en la GRID.

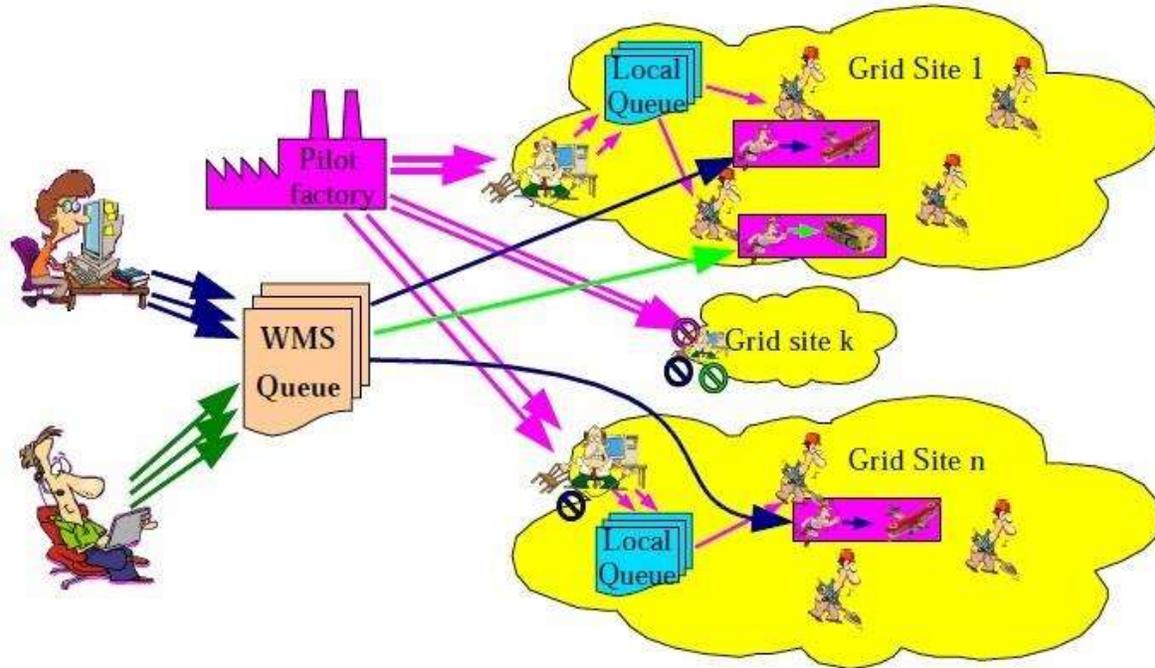


Figura 3.3 Infraestructura Pilot-Based WMS

3.5.2 Seguridad Pilot-Based con Glexec

Glexec es una herramienta de seguridad derivado de Apache que se puede instalar en los nodos de trabajo (Worker Node). GLExec es un ejecutable ligero el cual funciona otorgando un certificado X-509¹⁸ de manera temporal. Posteriormente el usuario es autenticado y autorizado para ejecutar la carga de trabajo, el cual no procede si antes no se ha asignado una identidad local. Algunas ventajas de la implementación de gLExec son:

- Los usuarios reciben protección a nivel sistema, protegiéndolos de interferencias con los Jobs de otros usuarios.
- Protege al WMS de usuarios malintencionados gracias a los certificados de usuario y la implementación de proxy temporal.
- Permite establecer políticas locales que garantiza que los trabajos no deseados no se ejecutaran en la GRID local. (Vea la figura 3.4)

¹⁸ En criptografía, X.509 es un estándar de la Unión Internacional de Telecomunicaciones para infraestructuras de claves públicas (en inglés, Public Key Infrastructure o PKI)

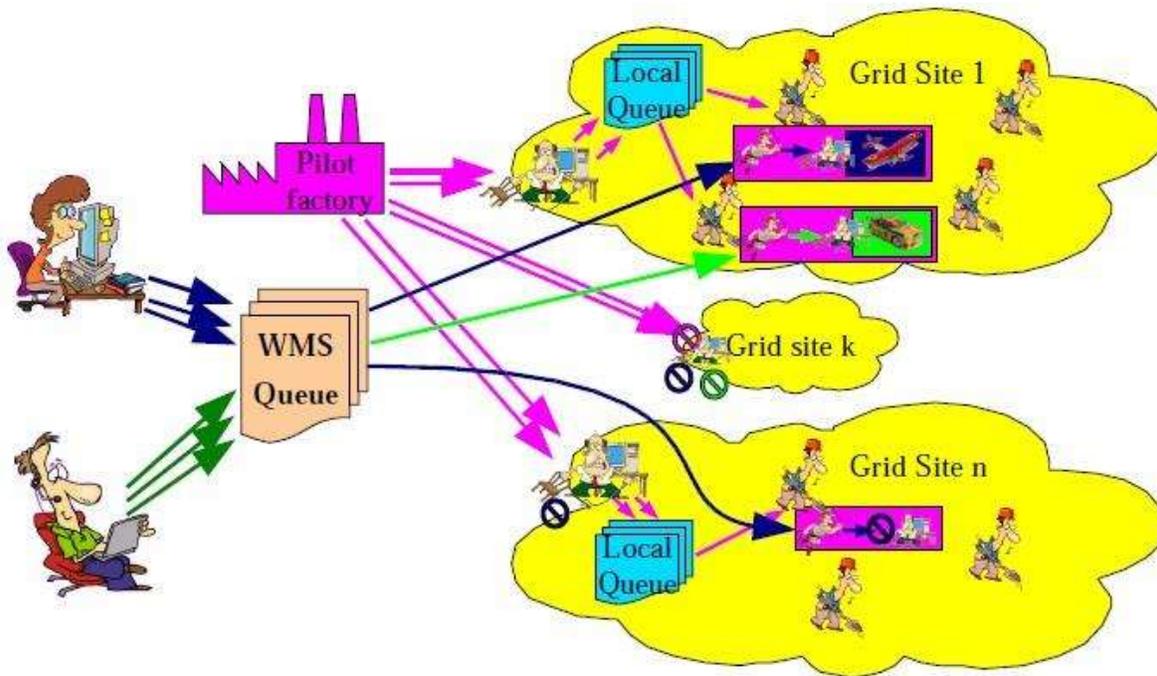


Figura 3.4 Infraestructura Pilot-Based con gLExec.

3.5.3 Certificados para Usuarios de la GRID

Vista la seguridad GRID desde una perspectiva más cercana podemos introducir el concepto de certificado de una manera más detallada. Por certificado nos referimos a una identidad digital X-509, la cual es una infraestructuras de claves publicas la cual específica, entre otras cosas formatos estándar para certificados de claves públicas y un algoritmo de validación. Su sintaxis, se define empleando el lenguaje ASN.1¹⁹ (Abstract Syntax Notation One) y los formatos de codificación más comunes como DER²⁰ (Distinguished Encoding

¹⁹ Es un protocolo de nivel de presentación en el modelo OSI (en inglés, Open System Interconnection 'sistemas de interconexión abiertos') es el modelo de red descriptivo, que fue creado por la Organización Internacional para la Estandarización (ISO) en el año 1980. Es un marco de referencia para la definición de arquitecturas en la interconexión de los sistemas de comunicaciones.

²⁰ La codificación DER es una manera de codificar en binario estructuras expresadas a través de la notación ASN.1.

Rules) y PEM²¹ (Privacy Enhanced Mail). Este certificado puede ser usado para acceder a un sistema informático, servicio o página web.

La entidad emisora de certificados es una organización llamada Certificate Authority (CA). Una vez que se tenga el certificado y esté por usarse, alguien debe decir si es válido o no. Esto implica que debe haber una “Cadena de Confianza”, de manera que el aceptante de su certificado confía en la CA que lo emitió.

Los certificados digitales se basan en tecnología Public Key Infrastructure (PKI)²². PKI utiliza un par de claves criptográficas (Públicas y Privadas), para cifrar y descifrar la información segura. Para ser precisos una credencial X-509 se compone de dos partes, el certificado público (que incluye una clave pública) y una clave privada.

Un principio central de esta tecnología es que la persona que solicita el certificado genera el par de claves Pública/Privada. La clave privada nunca es puesta a disposición de terceros. Cualquier información criptográficamente firmada con la clave privada del certificado se puede verificar con la clave pública del certificado. A su vez el certificado público es criptográficamente firmado por la CA. Así cuando una CA es de confianza, el destinatario tiene la seguridad de la identidad de la persona que tiene la llave privada. Un ejemplo de esto es que si se recibe un correo electrónico firmado criptográficamente por la clave privada del usuario, se puede comprobar de quien se trata mediante el certificado público de usuario (que contiene la clave pública).

Los certificados pueden pertenecer a diferentes tipos de entidades finales: las personas, host, servicios, y entidades emisoras. Los certificados de servicios se utilizan para autenticar hosts. Otros usos comunes incluyen firma de correo electrónico y la administración de acceso a sitios web restringidos. El uso primario de los certificados en la implementación será para iniciar las operaciones del cómputo GRID.

²¹ La codificación PEM no es más que la información codificada en DER, transformada en base 64.

²² PKI es un protocolo que trata de describir los procesos necesarios para la gestión de certificados digitales de claves públicas para el intercambio seguro de información, que permite firmar digitalmente un documento electrónico.

Para el primer uso, las credenciales se almacenaran normalmente en el directorio `~/globus`. El certificado que proporciona la CA es de extensión `*.p12` el cual tiene que ser convertido en `*.pem` mediante los siguientes comandos. Creamos el directorio `.globus`, y posteriormente movemos el archivo certificado al nuevo directorio.

```
mkdir .globus  
mv usercred.p12 .globus/  
openssl pkcs12 -in usercred.p12 -clcerts -nokeys -out $HOME/.globus/usercert.pem  
openssl pkcs12 -in usercred.p12 -nocerts -out $HOME/.globus/userkey.pem  
chmod 400 *.p12 *.pem
```

Es decir el certificado *usercert.pem* y la llave privada *userkey.pem* se pueden almacenar como archivos separados o agrupados por defecto en el archivo de extensión `.p12`. Toda herramienta de usuarios OSG funciona con ambos formatos, a menos que se especifique su uso separado. El formato compacto se usa principalmente para autenticarse en los sitios restringidos de la web de CERN.

Capítulo 4

Instalación de la GRID

4.1 Repositorios OSG y EPEL

En ésta etapa del proyecto usaremos un equipo con procesador i7 de tercera generación, el cual administra 4 núcleos de procesador con velocidad de reloj de 3.9 GHz. El sistema operativo será Scientific Linux 6.0 (SL6) el cual está instalado en un disco duro de estado sólido.

Los repositorios RPM (*Red Hat Package Manager*) requeridos por OSG deben ser instalados antes que cualquier Cliente o Nodo. Se debe instalar primeramente los Extra Packages for Linux Enterprise (EPEL). El procedimiento es el siguiente:

1. Si no están instalados los paquetes de EPEL pueden ser instalados mediante la siguiente instrucción:

```
rpm -Uvh http://download.fedoraproject.org/pub/epel/6/i386/epel-release-6-8.noarch.rpm
```

2. Instalación del plugins **yum-priorities**.

```
Yum install yum-priorities
```

3. Tenemos que asegurarnos que los plugins estén activos en el archivo de configuración: `/etc/yum.conf` agregando la siguiente línea.

```
Plugins=1
```

5. Instalación de los repositorios de OSG:

```
rpm -Uvh http://repo.grid.iu.edu/osg/3.1/osg-3.1-el6-release-latest.rpm
```

4.1.1 Instalación de los Repositorios de la Autoridad Certificadora.

Cuando instalamos el software con RPM, se tiene que decidir cómo se desea instalar los certificados de la Autoridad de Certificación (CA). Esto permitirá que el sistema GRID actualice el conjunto de certificados válidos y revocados. Se utilizará la configuración recomendada usando los certificados de OSG. Ver tabla (4.1.1.1)

Opciones	CA	Nombre	Línea comando (usuario root)
1	OSG	osg-ca-certs	yum install osg-ca-certs Recomendado
2	IGTF	igtg-ca-certs	yum install igtg-ca-certs
3	Vacío ²³	empty-ca-certs	yum install empty-ca-certs
4	Ninguno ²⁴	Any	yum install osg-ca-scripts

Tabla 4.1.1.1 Opciones de instalación de certificados.

Nota: si se usa la opción 1 o 2, será necesario ejecutar yum update con la finalidad de obtener la última versión de la CA.

Nota: Si se tiene instalado el servidor http de apache²⁵ será necesario reiniciarlo una vez que se haya ejecutado yum update, de otra manera se seguirá ejecutando una versión antigua de los certificados. La ubicación de los certificados, índices y archivos de configuración se detalla en la tabla 4.1.1.2.

Paquete	Archivo	Directorio
CA	Ubicación de CA	/etc/grid-security/certificates
CA	Índice de CA	/etc/grid-security/certificates/INDEX.txt
FETCH	Configuración	/etc/fetch-crl13.conf

Tabla 4.1.1.2 Ubicación de los certificados de la GRID

²³ *empty-ca-certs* indica que instalación de los certificados en el nodo sea realizada manualmente por el administrador del sistema local.

²⁴ *osg-ca-scripts* Indica que el administrador del sistema local proveerá un script que actualizará la descarga de los certificados

²⁵ El servidor HTTP Apache es un servidor web HTTP de código abierto.

4.2 Instalación del Cliente Nodo de Trabajo (Worker Node Client) (WN)

La instalación del WN se ejecuta con el siguiente comando:

```
yum install osg-wn-client
```

La instalación incluye la siguiente lista de aplicaciones y clientes:

- Bestman2 Client
- CA certificates
- dcache-srmclient
- dcap-tunnel-gsi
- dccp
- edg-gridftp-client
- Fetch CRL
- FTS client
- globus-url-copy(Cliente GridFTP)
- lcg-utils
- ldapsearch
- LFC
- MyProxy
- osg-system-profiler
- osg-version
- UberFTP
- vo-client
- wget

Si se requiere saber exactamente qué versión de nodo de trabajo está instalada se utiliza:

```
rpm -q --requires osg-wn-client
```

Luego se debe dejar el servicio activo mediante *cron*²⁶.

```
/usr/sbin/fetch-crl # Este Actualiza la CRLs  
/sbin/service fetch-crl-boot start # Inicia El servicio  
/sbin/service fetch-crl-cron start # Inicia el Cron  
/sbin/chkconfig fetch-crl-boot on #Habilita el Servicio tras cada reinicio  
/sbin/chkconfig fetch-crl-cron on #Habilita el Cron tras cada reinicio
```

²⁶ *Cron* es un administrador regular de procesos en segundo plano de Linux que ejecuta procesos a intervalos regulares.

4.3 Instalación de gLExec

Para continuar la instalación será necesario hacer la requisición de los certificados de host en la siguiente dirección: <https://oim.grid.iu.edu/oim/certificaterequesthost>. Para poder identificarnos en el portal será necesario haber tramitado una cuenta de usuario de la GRID del CERN.

El siguiente comando generara el archivo de llave privada `hostkey.pem` utilizando el nombre `umich.mx` como dominio de la GRID.

```
umask 077; openssl req -new -newkey rsa:2048 -nodes -keyout hostkey.pem -subj "/CN=umich.mx"
```

La llave privada se envía al grupo de administradores de la GRID. Quienes lo autorizan a través de la emisión del certificado `hostcert.pem`.

Tras finalizar el trámite anterior se debe que copiar el certificado y su llave privada como lo indica la siguiente tabla. Tabla 4.3

Certificado	Pertenencia	Dirección del certificado
Certificado del Nodo de Trabajo	root	/etc/grid-security/hostcert.pem
		/etc/grid-security/hostkey.pem

Tabla 4.3 Dirección de certificados

4.3.1 Configuración de gLExec

Las siguientes instrucciones son necesarias para completar la instalación de gLExec:

1. El contenido del archivo `/etc/glexec-conf` está configurado por defecto. El administrador de sistema puede modificarlo a su conveniencia.
2. El archivo de configuración `/etc/lcmads.db` es útil para asignar la siguientes propiedades de seguridad al servicio gLExec:
 - a) GUMS (Grid User Management System) actúa como servidor de autorización local. La configuración de `/etc/lcmads.db` es la siguiente:

```
--endpoint  
https://yourgums.yourdomain:8443/gums/services/GUMSXACMLAuthorizationServicePort"
```

-
- b) SAZ (Site Authorization Service) permite controlar el acceso de usuarios a cualquier recurso de red mediante un mecanismo de autenticación local y remota. La configuración de `/etc/lcmaps.db` es la siguiente:

```
--endpoint  
https://yoursaz.yourdomain:8443/saz/services/SAZXACMLAuthorizationServicePort"
```

- c) En el caso de no usar GUMS ni SAZ entonces puede usar la configuración: `GRIDMAPFILE` el cual es un método de autenticación únicamente local mediante un archivo que contiene información del usuario de la GRID. La configuración de `/etc/lcmaps.db` es la siguiente:

```
verifyproxy -> gridmapfile  
gridmapfile -> glexctracking
```

4.4 Requerimientos para la Instalación del Computin Element (CE)

El siguiente procedimiento describe la instalación y configuración del Computing Element (CE). Los requerimientos para la instalación son: un dominio host y los certificados de integración a la GRID (estos están asociados al CE y al Host). Los certificados de la GRID deben almacenarse en la siguiente ruta *“/etc/grid-security/hostcert.pem”* y *“/etc/grid-security/hostkey.pem”*. En el proceso de instalación se crearan cuentas de usuario las cuales estarán interactuando con el nodo Resource Broker. Finalmente es necesario que se configure el Firewall local universitario para que se abra el rango de puertos necesarios para el buen funcionamiento del servicio del nodo principal. Los puertos asignados son:

NOMBRE DEL SERVICIO	PROTOCOLO	PUERTOS	INBOUND	OUTBOUND
GLOBALUS, CONDOR	TCP	20000-24999	•	•
GRAM	TCP	2119	•	
GRID FTP	TCP	2811	•	
HTCondor-CE CLIENTE	TCP	9619	•	
HTCondor-CE SHARED	TCP	9620	•	
STORAGE RESOURCE MANAGER	TCP	8080	•	
STORAGE RESOURCE MANAGER	TCP	8443	•	
GSISSH	TCP/UDP	22	•	•
MYPROXY	TCP	7512	•	
VOMS	TCP	15001-15003	•	
SQUID	TCP	3128	•	•
SQUID MONITOR	UDP	3401	•	
BOSCO SUBMINT HOST	TCP	11000	•	
SSH	TCP	22	•	•

Tabla 4.4 Puertos de operación

4.4.1 Instalación del Administrador de Recursos Torque

El CE trabajara en conjunto con el administrador de recursos y trabajos. Se usara Torque el cual permitirá tener control sobre los trabajos enviados al nodo principal. Para instalarlo se hará lo siguiente:

Se descargara Torque version 5.x desde la página del proveedor o bien mediante el portal GitHub:

```
http://www.adaptivecomputing.com/support/download-center/torque-download/  
git clone https://github.com/adaptivecomputing/torque.git -b 5.0.0 5.0.0
```

Se requerirán algunas paqueterías para la correcta instalación de torque. Para instalarlas siga el siguiente procedimiento:

```
yum update  
yum install libxml2-devel openssl-devel gcc gcc-c++ boost-devel
```

Una vez descargado el paquete se procederá a configurar la instalación:

```
tar -xzvf torque-5.0.0.tar.gz  
cd torque-5.0.0/  
./configure  
make  
make install
```

Una vez instalado se debe configurar el nombre del servidor y asignarlo al host del CE. El archivo de configuración se encuentra en `/var/spool/torque/server_name`.

```
echo <higgs.cms.umich.mx> > /var/spool/torque/server_name
```

Se inicializará el servicio mediante la ejecución del script `torque.setup`.

```
./torque.setup root
```

4.4.2 Instalación del CE

Para la instalación del nodo principal ejecutaremos el siguiente comando:

```
yum install osg-ce-pbs
```

La configuración de seguridad para el CE debe modificarse a conveniencia por el administrador en el archivo siguiente `/etc/edg-mkgridmap.conf`. Por defecto el archivo esta configurado de la siguiente manera:

```
#### GROUP: group URI [lcluser]
#
#-----
# USER-VO-MAP mis MIS -- 6 -- Rob Quick (rquick@iupui.edu)
group vomss://voms.grid.iu.edu:8443/voms/mis mis
#-----
# USER-VO-MAP osgedu OSGEDU -- 24 -- Rob Quick (rquick@iupui.edu)
group vomss://voms.grid.iu.edu:8443/voms/osgedu osgedu
```

4.4.3 Detalles del Cliente OSG

Este paquete es necesario que se instale en el nodo de trabajo ya que es utilizado por los usuarios para enviar Jobs, transferencia de datos e interactuar con la GRID.

```
yum install osg-client
```

De manera general la instalación incluye:

- 1.- Globus Toolkit: provee clientes y herramientas para la autorización de Jobs y transferencia de datos.
- 2.- Certificados CA: certificados de confianza de la OSG.
- 3.- Lista de certificados revocados y herramientas de seguridad.
- 4.- Cliente HTCondor: administra el flujo y envío de Jobs.
- 5.- Clientes que permiten el acceso al Storage Element Manager (SEM).

El nombre de los clientes y paquetes son los siguientes:

- Bandwidth Test Controller (bwctl) client
- GSI OpenSSH client
- Globus GRAM clients

-
- Globus certificare utilities (Incluyendo grid-proxy-init)
 - Network Diagnostic Tool (NDT)
 - Nmap (Scanner de Puertos)
 - One-Way Ping (owamp) client
 - lcg-info
 - lcg-infosites
 - osg-cert-scripts
 - osg-discovery
 - osg-system-profiler
 - osg-version

Si se quiere saber qué versión está instalada ejecutamos el siguiente comando:

```
rpm -q --requires osg-client
```

4.5 Instalación del Storage Element (SE)

El nodo del SE es un conjunto de servicios encargados de dar permisos de almacenamiento a aplicaciones y usuarios para una recuperación futura. Por tanto, todos los Jobs guardados en el SE son de solo lectura y no pueden ser modificados salvo que sean remplazados por datos nuevos. El proceso de instalación y configuración es el siguiente:

```
yum install osg-se-hadoop-namenode
yum install osg-se-hadoop-secondarynamenode
yum install osg-se-hadoop-datanode
yum install osg-se-hadoop-gridftp
yum install lcmaps-plugins-gums-client
yum install lcmaps-plugins-basic
```

Para continuar con la configuración será necesario designar una partición para el sistema de archivos FUSE (File System in User Space).

```
mkdir /mnt/hadoop
mount /mnt/hadoop
```

Para un correcto funcionamiento será necesario crear los usuarios que operaran en la GRID. Es importante designar áreas de trabajo para cada usuario. Y los procesos de seguridad y permisos serán equivalentes a los comandos de administración de usuarios de Linux.

Se crearan directorios de VO en los siguientes comandos:

```
mkdir /mnt/hadoop/cms
mkdir /mnt/hadoop/dzero
mkdir /mnt/hadoop/sbgrid
mkdir /mnt/hadoop/fermigrid
mkdir /mnt/hadoop/cmstest
mkdir /mnt/hadoop/osg
```

En el siguiente comando se crearan áreas de trabajo para el usuario:

```
mkdir -p /mnt/hadoop/cms/store/user/nombredeusuario
```

Se debe asignar a cada usuario a un grupo de trabajo.

4.6 Validando el CE

El Computing Element (CE) permite ejecutar Jobs en la GRID. Es necesario validar la autenticación de los usuarios y verificar que los mecanismos de seguridad estén funcionando correctamente. Para ello se realizan los siguientes pasos:

1. Creación del proxy con el comando voms-proxi.init.

```
voms-proxy-init -voms GLOW
```

```
Enter GRID pass phrase for this identity:
```

```
Your identity: /DC=org/DC=doegrids/OU=People/CN=Alain Roy 424511
```

```
Creating temporary proxy ..... Done
```

```
Contacting glow-voms.cs.wisc.edu:15001 [/DC=org/DC=doegrids/OU=Services/CN=glow-
voms.cs.wisc.edu] "GLOW" Done
```

```
Creating proxy ..... Done
```

Se debe sustituir la cadena “GLOW” por la Virtual Organization (VO) en este caso es CMS.

- 2. Se usará el comando siguiente para asegurar que se tiene acceso al host designado para la universidad.**

```
globusrun -a -r cms.umich.mx  
GRAM Authentication test successful
```

- 3. Se validará la ejecución de Jobs.**

- i. Se creará el proxy (Ver punto 1.0).
- ii. Se ejecutará un Job de manera local.

```
globus-job-run fermicloud081.fnal.gov /bin/hostname  
cms.umich.mx
```

- iii. Se enviará un Job de manera local.**

```
globus-job-run fermicloud081.fnal.gov/jobmanager-pbs /bin/hostname  
cms.umich.mx
```

- 4. Validación de transferencia de archivos desde y hacia el CE.**

- i. Crear el proxy (ver punto 1.0)**
- ii. Copiar un archivo al CE**

```
globus-url-copy  
file:///proc/cpuinfo gsiftp://fermicloud081.fnal.gov/tmp/cpuinfo-deleteme
```

Si se ejecuta correctamente terminara de manera silenciosa.

iii. Copiar un archivo desde el CE.

```
globus-url-copy gsiftp://fermicloud081.fnal.gov/proc/version file:///tmp/version-
deleteme

cat /tmp/version-deleteme

Linux version 2.6.18-238.19.1.el5 (gcc version 4.1.2 (Red Hat 4.1.2-50)) #1 SMP Fri Jul 15
00:48:58 EDT 2011

rm -f /tmp/version-deleteme
```

5. Validando las VO's

Una vez validado el CE se tiene que decidir cuales VO's tendrán permiso de acceder, verificando lo siguiente:

- i. Configurar el acceso a las VO's deseadas.
- ii. Que fueron creadas las cuentas apropiadas para cada usuario.
- iii. Que las áreas de almacenamiento fueran definidas y que son accesibles por los miembros de la VO.

Una vez que los criterios anteriores fueron definidos podemos estar seguros que los miembros de la VO tienen la intención de colaborar y podrán acceder a los servicios prestados por el nodo. Para finalizar será necesario ejecutar los siguientes servicios que hacen referencia al tipo de seguridad implementada (GRIDMAPFILE).

```
/usr/sbin/edg-mkgridmap
/usr/bin/gums-host-cron
/sbin/service edg-mkgridmap start
/sbin/service gums-client-cron start
```

4.6.1 Verificar que las VO son compatibles

Los archivos `/var/lib/osg/supported-vo-list` (para `edg-mkgridmap`) y `/var/lib/osg/gums-undefined-accounts` (para GUMS) muestra la lista de VO que acepta el SITE.

4.6.2 Verificar que las todas las cuentas de UNIX fueron creadas y son accesibles

El siguiente archivo `/var/lib/osg/undefined-accounts` muestra la lista de usuarios permitidos en el SITE. Estas cuentas no están dadas de alta en el sistema operativo. Si las cuentas se dan de alta de manera local entonces este archivo debe estar vacío.

4.7 Problemas en la Implementación

Un Firewall (Corta fuegos en español) es parte de un sistema o una red que está diseñada para bloquear el acceso no autorizado y permite, limitar, cifrar y descifrar el tráfico de la red en base a un conjunto de normas y criterios.

Los Firewall pueden ser implementados en hardware o software, o una combinación de ambos. Se utilizan con frecuencia para evitar que los usuarios de Internet no autorizados tengan acceso a redes privadas conectadas a Internet, especialmente intranets. Todos los mensajes que entren o salgan de la intranet pasan a través del Firewall, que examina cada mensaje y bloquea aquellos que no cumplen los criterios de seguridad especificados. También es frecuente configurar dispositivos a una red llamada zona desmilitarizada o DMZ, en la que se ubican los servidores de la organización que deben permanecer accesibles desde la red exterior.

En el proceso de “validación del CE” se presentaron problemas en torno a la seguridad que brinda el sistema de Firewall de la universidad. Un ejemplo se encuentra en la sección 4.2 en la Instalación del cliente nodo de trabajo, cuando se ejecuta el comando `/usr/sbin/fetch-crl`. Este comando se encarga de descargar una lista de certificados revocados por el CERN. La GRID puede operar si esta lista de certificados está actualizada. La lista se actualiza haciendo conexión directa a servidores del CERN a través de puertos específicos. Tales puertos de salida están bloqueados por el servicio de Firewall de la universidad, lo cual impide el funcionamiento de la GRID. Este problema se solucionó de manera temporal instalando un servidor PROXY-SQUID²⁷ el cual se encargara de descargar los certificados a través de un proxy externo.

Respecto a los puertos de comunicación entrante y saliente, el cliente encargado de enviar y recibir trabajos (GLOBUS) requiere que los puertos 20000-24999 estén abiertos en ambos sentidos. Por política de seguridad de la Universidad no se pueden abrir más de

²⁷ Un servidor proxy permite a otros equipos conectarse a una red de forma indirecta a través de él. Cuando un equipo de la red desea acceder a una información o recurso, es realmente el proxy quien realiza la comunicación y a continuación traslada el resultado al equipo inicial.

3 puertos consecutivos, ni de entrada ni de salida. Será necesario que en un futuro el servidor GRID sea asignado al DMZ y la seguridad mediante un firewall local para el nuevo equipo de supercómputo de la Universidad Michoacana.

El tercer tipo de problema es respecto a que CMS, y propiamente el CERN, debe aprobar el proyecto para autorizar a la Universidad Michoacana como TIER 3. Una vez terminado este proceso se liberará el software de CMS para su uso local.

Capítulo 5

Instalación del Software de CMS

5.1 Sistema de Gestión de Trabajos de CMS

CMS ha desarrollado varias herramientas para automatizar y facilitar la gestión de trabajos de análisis, de producción Monte Carlo y de reprocesamiento de los datos, haciendo transparente para el usuario el uso de los comandos Grid que ejecutan las operaciones más básicas. Para la gestión de trabajos de análisis, la herramienta que se ha desarrollado en CMS recibe el nombre de CMS Remote Analysis Builder (CRAB) [13]. Esta herramienta permite crear los Jobs, enviarlos a la Grid, monitorizar su estado, recuperar los Jobs procesado, reenviarlos en caso de fallo o cancelarlos mientras se encuentran en ejecución.

5.2 Instalación

El software de CMS puede ser instalado de manera local. El administrador tiene que definir los directorios de trabajo. Será necesario crear un grupo de usuarios y dar de alta una cuenta que administre las versiones de software de CMS. Ésta cuenta no puede ser el usuario root, de lo contrario nadie más que root tendrá acceso a las versiones. Se creó la cuenta “cmssgm” y se definió el directorio de trabajo “/CMS/osg/app”. A continuación se describirá el proceso de instalación, configuración de directorios y comandos básicos del software de CMS.

En primera instancia se realizará el proceso mediante el usuario cmssgm. Corremos el script de la siguiente ruta el cual activará variables de entorno útiles en el proceso de instalación:

```
source /CMS/osg/users/cmssgm/cms_login.sh
```

Este último a su vez ejecuta el siguiente script:

```
/CMS/osg/app /cmssoft/cms/cmsset_default.sh
```

el cual asigna las siguientes variables:

```
export OSG_APP=/CMS/osg/app
export SCRAM_ARCH= slc6_amd64_gcc472
export LANG="C"
```

Estas variables pueden ser modificadas manualmente de acuerdo a la arquitectura y compilador que se requiera utilizar.

Definidas las variables de entorno se continúa con la instalacion.

```
sh -x $OSG_APP/bootstrap.sh setup -path $OSG_APP -arch $SCRAM_ARCH >&
$OSG_APP/bootstrap_$SCRAM_ARCH.log
```

El siguiente script define las variables que permitirán instalar el software de CMS mediante el comando apt-get.

```
source $OSG_APP/$SCRAM_ARCH/external/apt/*/etc/profile.d/init.sh
```

Posteriormente se podrá consultar las versiones disponibles del software de CMS.

```
apt-cache pkgnames |grep CMSSW_5
```

Una vez que encontramos la version procedemos a instalar mediante apt-get:

```
apt-get update
apt-get install CMSSW_5_X_Y
```

Por ejemplo si se instaló la version CMSSW_5_3_11, se puede crear un directorio de trabajo con los siguientes comandos:

```
/home/cmssgm/work/
source /CMS/osg/users/cmssgm/cms_login.sh
cmsrel CMSSW_5_3_11
cmsenv
```

5.3 Instalación de CRAB

CRAB es una herramienta de interacción de la GRID [14] con los usuarios en el User Interface (UI). CRAB da soporte a cualquier programa ejecutable basado en el software oficial del experimento CMS, y con cualquier modulo o librería incluyendo las propias del usuario.

A continuación se describirá el proceso de instalación.

Se creó un directorio de instalación del CRAB en `/CMS/osg/users/cmssgm/CRAB/`. La versión que se ha compilado es CRAB 2.10.2, la cual se obtuvo de la siguiente manera:

```
wget http://cmsdoc.cern.ch/cms/ccs/wm/www/Crab/Docs/CRAB_2_10_2.tgz
tar -xvzf CRAB_2_10_2.tgz
```

Luego se ejecutó el archivo de configuración:

```
cd YOUR_DIR/CRAB_2_10_2/
./configure
```

el cual crea los script `cms_ui_env.sh` y `crab.sh` en el directorio `/CMS/osg/users/cmssgm/CRAB/`.

5.3.1 Comandos Básicos

Para definir el ambiente de CRAB será necesario ejecutar el siguiente script:

```
source /afs/cern.ch/cms/ccs/wm/scripts/Crab/crab.csh
```

El comando para crear un nuevo Job se encarga de consultar la disponibilidad de la GRID, y prepara los archivos y librerías especificadas en el archivo de configuración `crab.cfg`.

```
crab -create -cfg archivo.cfg
```

Una vez que ejecutamos el comando anterior se hará la petición para identificar al usuario que está haciendo la petición de ejecución del Job.

Con el siguiente comando será posible especificar el envío de uno o más Jobs:

```
crab -submit -c <project_dir_name>
```

Una vez enviado el Job será necesario monitorear los avances del proyecto, y se utilizará el siguiente comando:

```
crab -status -c <project_dir_name>
```

La recuperación de un Job finalizado será mediante el siguiente comando:

```
crab -getoutput -c <project_dir_name>
```

Finalmente para copiar el trabajo completo y terminado del Storage Element (SE) de la GRID será necesario ejecutar

```
crab -copyData -c <project_dir_name>
```

Capítulo 6

Análisis de Datos

6.1 Antecedentes del Bosón Z

Debido al gran éxito de la electrodinámica cuántica para el caso de la interacción electromagnética en los años 50, los científicos intentaron desarrollar una teoría similar para la interacción débil. La teoría culminó con la aparición de la teoría que unifica al electromagnetismo con la interacción débil. Por su trabajo en la teoría electrodébil Sheldon Glashow, Steven Weinberg, y Abdus Salam recibieron el premio Nobel de física (1984). La teoría electrodébil postuló entonces la existencia de los bosones W para explicar la desintegración beta, y también postuló la existencia del bosón Z para interacciones neutras. El mecanismo de Higgs, dota de masa a los bosones W y Z y además predice la existencia del bosón de Higgs. El descubrimiento de los bosones W y Z se realizó con la construcción del Súper Protón Synchrotron en el CERN, durante una serie de experimentos dirigidos por Carlo Rubbia y Simon Van Der Meer. Ambos científicos recibieron el premio Nobel de física en 1984 por su descubrimiento.

En la relatividad especial la relación de energía-momento es una ecuación que relaciona la energía y el momento de una partícula con su masa en reposo. La ecuación es la siguiente

$$E^2 - \vec{P}^2 c^2 = m^2 c^4 \quad (6.1)$$

Utilizando los cuadrimomentos:

$$P^\mu = \left(\frac{E}{c}, P_1, P_2, P_3 \right) = \left(\frac{E}{c}, \vec{P} \right) \text{ y } P^\mu = \left(\frac{E}{c}, -P_1, -P_2, -P_3 \right) = \left(\frac{E}{c}, -\vec{P} \right) \quad (6.2)$$

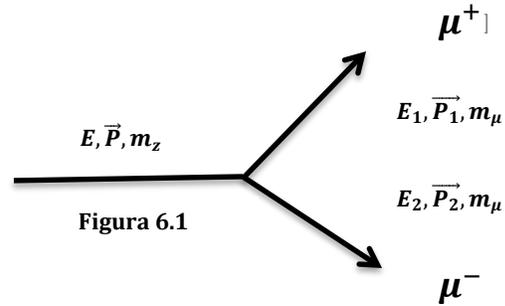
y haciendo el producto interno:

$$\sum_{\mu=0}^4 P^\mu P_\mu = \left(\frac{EE}{c^2} + P_1(-P_1) + P_2(-P_2) + P_3(-P_3) \right) = \frac{E^2}{c^2} - \vec{P}^2 = m^2 c^2 \quad (6.3)$$

Se recupera la ec. 6.1.

La relación es válida para la desintegración de un bosón Z a dos muones. (Vea figura 6.1).

$$m_Z^2 c^2 = P^\mu P_\mu = (P_1^\mu + P_2^\mu)(P_\mu^1 + P_\mu^2) \quad (6.4)$$



En donde en la última igualdad se utiliza la conservación de la energía y el momento de la relatividad especial.

Dado que la masa de ambos muones es la misma, luego al hacer el producto se tiene que:

$$m_Z^2 c^2 = m_\mu^2 c^2 + 2\left(\frac{E_1 E_2}{c^2} - \vec{P}_1 \cdot \vec{P}_2\right) + m_\mu^2 c^2 \quad (6.5)$$

Así la masa del bosón Z queda en términos de la masa, energía y momento de los muones:

$$m_Z = \sqrt{2m_\mu^2 + \frac{2}{c^2}\left(\frac{E_1 E_2}{c^2} - \vec{P}_1 \cdot \vec{P}_2\right)} = 91.1876 \text{ (21) GeV}/c^2 \quad (6.6)$$

6.2 Análisis de Datos CMS para Obtener la Distribución de Masa del Bosón Z

En esta sección se mostrarán los procesos de análisis de datos del software desarrollado para el experimento CMS. A continuación se detallarán los comandos básicos, procedimientos de filtrado de datos, transformación de archivos mediante la herramienta analyzeBasicPat, el envío de trabajos mediante CRAB y finalmente la creación de histogramas mostrando los resultados de un análisis estadístico.

Se establecerá un área de trabajo:

```
mkdir CMSSGM
cd CMSSGM
cmsrel CMSSW_5_3_11
```

6.2.2 Herramientas de Análisis de Datos

El análisis de datos en CMS es una tarea complicada ya que involucra varios procesos y es necesario hacer comparaciones entre diferentes tipos de análisis para obtener conclusiones. Para maximizar la productividad se han desarrollado un conjunto de herramientas de análisis. Una de ellas es el Physics Analysis ToolKit (PAT). PAT es una capa de análisis de alto nivel, su objetivo es satisfacer las necesidades de del análisis de CMS, proporcionando facilidad de uso a usuarios principiantes y avanzados.

Comenzaremos ejecutando el siguiente script `simple_PAT_MC_cfg.py` el cual procesara el archivo de datos `CMSDataAnaSch_Data_536.root` que contiene una submuestra de colisiones, limitado a un máximo de 1000 eventos.

```
cmsRun simple_PAT_MC_cfg.py
```

Fragmento del Script :

```
## Importando Paqueterías PAT.
from PhysicsTools.PatAlgos.patTemplate_cfg import *

## Ejecución del proceso
process.p = cms.Path(rocess.patDefaultSequence)

## Aquí definimos la fuente de datos.

process.GlobalTag.globaltag = 'START53_V7G::All'
process.source.fileNames =
['file:/afs/cern.ch/cms/Tutorials/TWIKI_DATA/CMSDataAnaSch_RelValZMM53
6.root']

##Aquí se define el archive procesado

process.maxEvents.input = -1          ##( -1 to run on all events)
process.out.fileName = 'simple_PAT_MC.root'      ##(e.g.
'myTuple.root')
```

El comando anterior creara un archivo con nombre simple_PAT_MC.root, el cual será leído por la terminal root con la siguiente línea de comandos.

```
root -l
gSystem->Load("libFWCoreFWLite.so");
AutoLibraryLoader::enable();
gSystem->Load("libDataFormatsFWLite.so");
gROOT->SetStyle ("Plain");
gStyle->SetOptStat(111111);
TFile::Open("simple_PAT_MC.root");
TBrowser b;
```

Una vez ejecutado el comando TBrowser b se mostrara en pantalla una interfaz la cual tiene como función entregar histogramas de los datos procesados. (Vea figura 6.2).

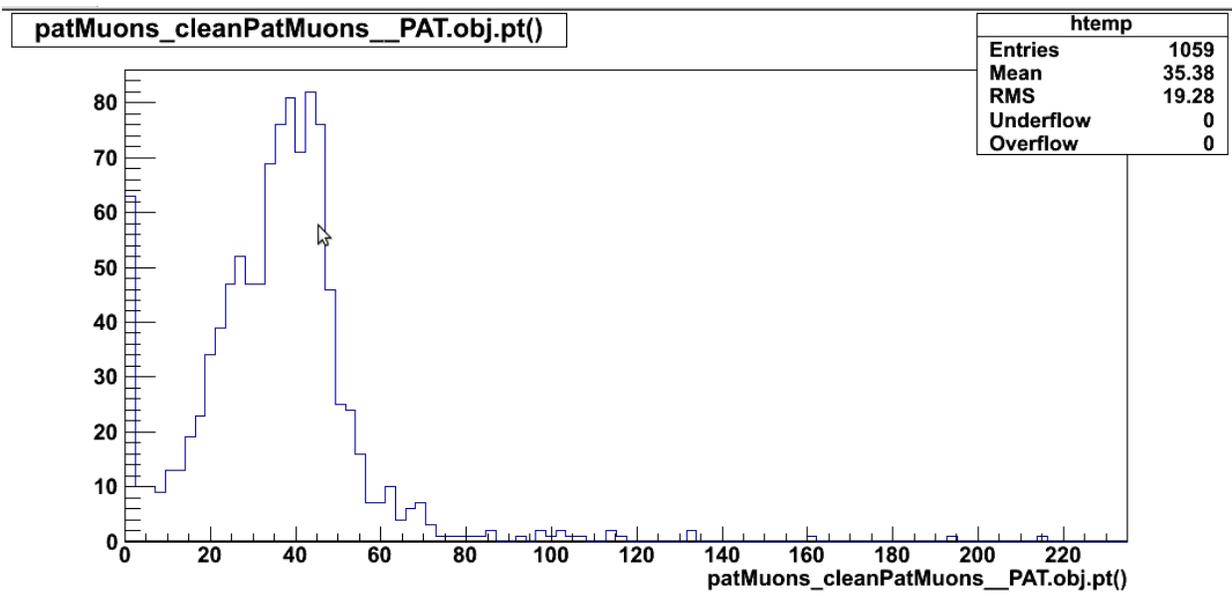


Figura 6.2 Momento transverso de muones en una submuestra de datos.

A continuación se mostrará el proceso de filtrado de eventos producidos en la colisión. El siguiente script llamado `simple_PAT_MC_MuEle_cfg.py` está diseñado para filtrar únicamente los eventos producidos por muones y electrones:

```
## Importando Paqueterías PAT.
from PhysicsTools.PatAlgos.patTemplate_cfg import *

## Ejecución del proceso pero con filtrado de datos,
process.p = cms.Path(rocess.patDefaultSequence)

##Definimos la fuente de datos.

process.GlobalTag.globaltag = 'START53_V7G::All'
process.source.fileNames =
['file:/afs/cern.ch/cms/Tutorials/TWIKI_DATA/CMSDataAnaSch_RelValZMM53
6.root']

##Definimos la salida.

process.maxEvents.input = -1          ## ( -1 to run on all events)
process.out.fileName = 'simple_PAT_MC.root'      ## (e.g.
'myTuple.root')
```

Ejecutamos el script y esto producirá la salida `simple_PAT_MC_MuEle.root`.

```
cmsRun simple_PAT_MC_MuEle_cfg.py
```

Utilizaremos la paquetería FWLite para obtener la gráfica de masa de la partícula Z.

```
git cms-addpkg PhysicsTools/FWLite
git cms-addpkg PhysicsTools/UtilAlgos
git cms-addpkg DataFormats/FWLite
git cms-addpkg DataFormats/Common
git cms-addpkg FWCore/FWLite
git cms-addpkg DataFormats/MuonReco
git cms-addpkg DataFormats/PatCandidates
scram b ## Compilara os paquetes
```

El visualizador de eventos CMS Event Display también conocido como Fireworks, nos permite visualizar los eventos físicos en las colisiones. Es posible ejecutarlo localmente en

cualquier dispositivo con Sistema Operativo Linux o Macintosh. Puede descargarse mediante:

```
wget http://cern.ch/cmsshow/cmsShow-5.2-3.linux.tar.gz
```

Ejecutaremos el Fireworks de los siguientes archivos *root* los cuales permitirán observar las trazas de partículas al momento de la colisión.

```
cmsShow /afs/cern.ch/cms/harteaga/TWIKI_DATA/CMSDataAnaSch_Data_536.root
```

```
cmsShow /afs/cern.ch/cms/harteaga/TWIKI_DATA/CMSDataAnaSch_RelValZMM536.root
```

La siguiente imagen muestra la reconstrucción de los archivos *root* anteriores. Se puede observar en el tracker la traza de partículas cargadas hasta el momento en que decaen por completo. En el siguiente nivel se encuentra el calorímetro electromagnético encargado medir la energía de fotones y electrones. En el siguiente nivel se encuentra el calorímetro hadrónico el cual determina la energía de protones, neutrones, piones y kaones. Finalmente se encuentra el detector de muones.

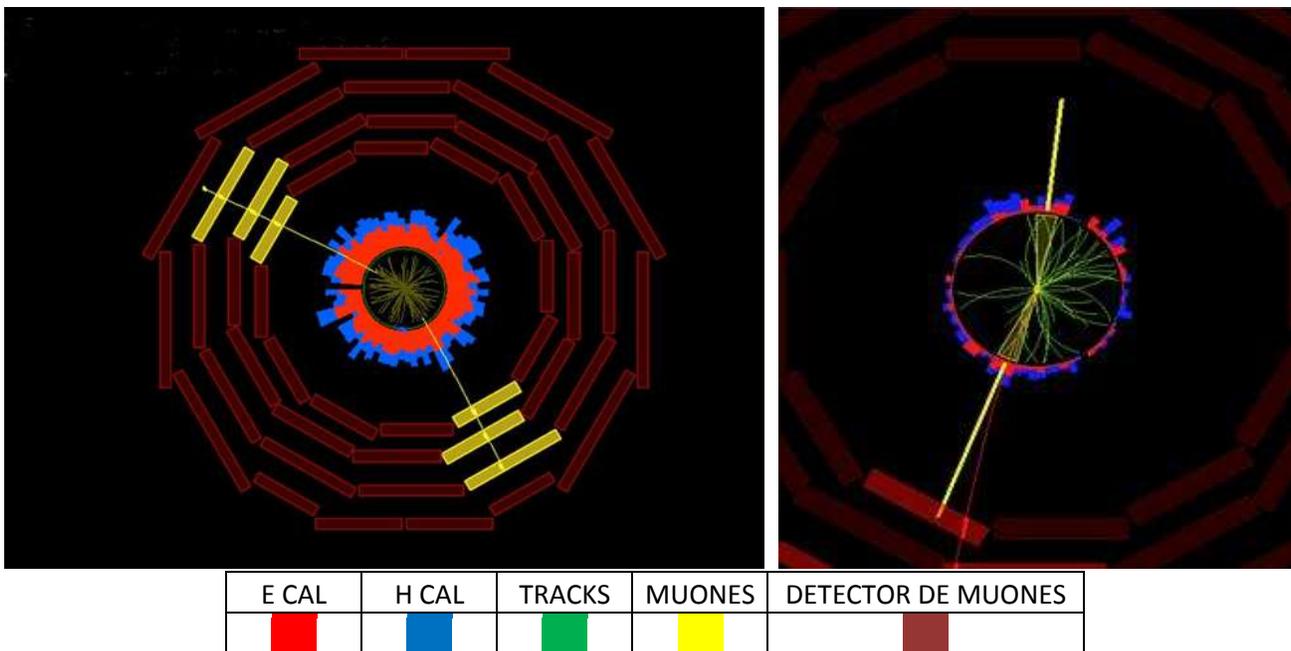


Figura 6.3 Decaimiento del bosón Z a dos muones.

Se agregara el paquete de datos PhysicsTools / PatExamples el cual contiene la clase *EDAnalyzer* que permitirán convertir las trazas en magnitudes físicas.

```
git cms-addpkg PhysicsTools/PatExamples
```

Posteriormente ejecutamos:

```
cmsRun MyZPeak_cfg.py
```

De esta manera se creará el archivo myZPeakCRAB.root el cual ya ha sido filtrado y procesado.

Iniciamos la terminal de root y ejecutamos la visualización, la cual estará ajustada a una distribución gaussiana:

```
Root -l  
TFile f("myZPeakCRAB.root");  
f.cd("analyzeBasicPat");  
gStyle->SetOptFit(111111);  
mumuMass->Fit("gaus");
```

La salida será la gráfica de distribución de masa de la partícula Z. (Vea figura 6.4).

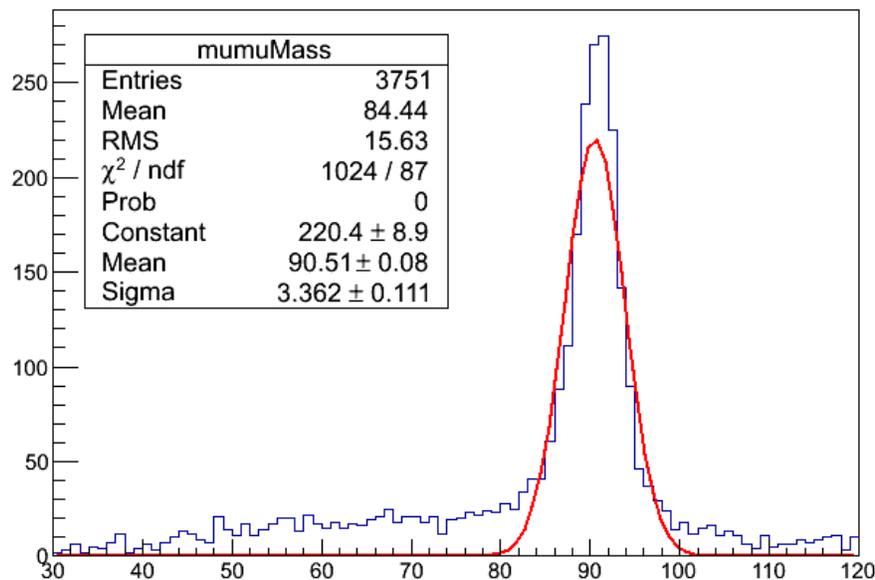


Figura 6.4 Espectro de masa de dos muones de 30 a 120 GeV ajustado a una gaussiana simple.

Para lograr un mejor ajuste se usará como distribución la convolución de la distribución Gaussiana y la distribución relativista de Breit-Wigner. Las cuales son las siguientes:

Distribución Gaussiana:

$$G(x; \mu, \sigma) = \frac{1}{\sqrt{2\pi}\sigma} \exp\left[-\frac{(x - \mu)^2}{2\sigma^2}\right]$$

Distribución relativista Breit-Wigner:

$$B(m; M, \Gamma) = N \cdot \frac{2}{\pi} \cdot \frac{\Gamma^2 M^2}{(m^2 - M^2)^2 + m^4 (\Gamma^2 / M^2)}$$

Convolución B*G:

$$P(m) = \int B(m'; M, \Gamma) \cdot G(m - m'; \mu, \sigma) dm'$$

Definiremos el rango de los datos del archivo myZPeakCRAB.root, y serán definidos filtros y parámetros de las distribuciones anteriores mediante el script FitZPeak.C.

```
{
#include "TF1.h"
#include "TH1.h"
#include "TMath.h"

gROOT->LoadMacro("BW.C");
gStyle->SetOptFit(111111);

TFile f("myZPeakCRAB.root");

f.cd("analyzeBasicPat");
Z_mass=(TH1F*)mumuMass->Clone();

int division = mumuMass->GetNbinsX();
//float massMIN = mumuMass->GetBinLowEdge(1);
//float massMAX = mumuMass->GetBinLowEdge(division+1);

float massMIN = 30.0;
float massMAX = 120.0;

float BIN_SIZE = mumuMass->GetBinWidth(1);
```

```

////////////////////////////////////
//Para la distribución Gaussiana//
////////////////////////////////////

TF1 *func = new TF1("mygauss",mygauss,massMIN, massMAX,3);
func->SetParameter(0,1.0);   func->SetParName(0,"const");
func->SetParameter(2,5.0);   func->SetParName(2,"sigma");
func->SetParameter(1,95.0);   func->SetParName(1,"mean");

Z_mass->Fit("mygauss","QR");
TF1 *fit = Z_mass->GetFunction("mygauss");

////////////////////////////////////
//Para la distribución Breit-Wigner//
////////////////////////////////////

TF1 *func = new TF1("mybw",mybw,massMIN, massMAX,3);
func->SetParameter(0,1.0);   func->SetParName(0,"const");
func->SetParameter(2,5.0);   func->SetParName(1,"sigma");
func->SetParameter(1,95.0);   func->SetParName(2,"mean");

Z_mass->Fit("mybw","QR");
TF1 *fit = Z_mass->GetFunction("mybw");

    fit->SetLineColor(4);
    fit->SetLineWidth(3);

Z_mass->SetMarkerStyle(20);
Z_mass->SetMarkerSize(1.0);
Z_mass->SetMarkerColor(2);
Z_mass->SetLineWidth(2.0);
Z_mass->SetXTitle("Z Mass (in GeV/c^{2})");
// Z_mass->GetXaxis()->SetTitleOffset(1.0);

Z_mass->Draw("PE0");
// c1->Print("myZmass_BWfitted.eps");
// c1->Print("myZmass_Gausfitted.eps");

```

Para visualizar el histograma ejecutamos:

```
root -l FitZPeak.C
```

La salida será la gráfica de distribución de masa de la partícula Z. (Vea figura 6.5).

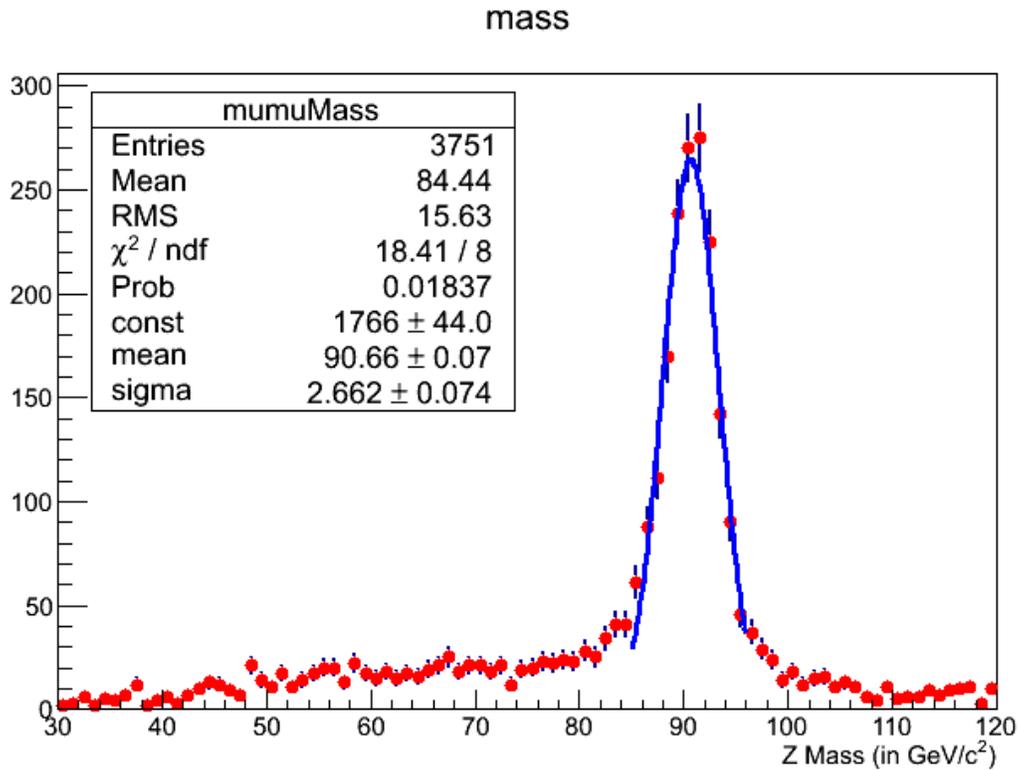


Figura 6.5 Histograma de la distribución de masa del bosón Z.

En conclusión, el resultado de haber usado una pequeña submuestra de los datos de colisión del LHC y un filtro de dos muones es:

$$m_z = 90.66 \pm 0.07 \frac{\text{GeV}}{c^2} \quad (6.7)$$

El cual puede compararse con el valor promedio mundial de masa del Z de $91.1876 \pm 0.0021 \frac{\text{GeV}}{c^2}$. El resultado de (6.7) es preliminar, no está corregido por escalas de momentos y no incluye incertidumbres sistemáticas.

Conclusiones

Para poder continuar con los objetivos de física de altas energías, es necesario procesar y analizar una gran cantidad de datos del orden de decenas de petabytes por año. Los modelos de cómputo clásico de las universidades dedicadas a la investigación, no resultan adecuados para gestionar el análisis de datos. El CERN desarrolló un modelo de cómputo (GRID) capaz de dejar de lado la distancia entre laboratorios, y así reducir costos en inversión y soporte técnico, además de asegurar un alto nivel de seguridad.

Los métodos de seguridad publicados por la OSG son a través de certificados de usuario, los cuales son verificados en cada interacción. Por otro lado se configuró el servidor de autenticación GRIDMAPFILE el cual es un método de autenticación únicamente local. Se eligió así porque de esta manera el administrador del sistema siempre tendrá el control de los usuarios que están enviando Jobs a la GRID local. La GRID implementada en la Universidad Michoacana cumple con los estándares y políticas de seguridad publicados por el CERN.

Una vez que la integridad de los nodos de trabajo fue verificada, se procedió a documentar el método de instalación del software de análisis de datos (CMSSW). La reconstrucción de las colisiones demanda gran cantidad de recurso de cómputo y de algoritmos de seguimiento que involucran procesos de filtrado a miles de trayectorias. En el ejemplo de la reconstrucción del bosón Z se realizó un filtro de las trayectorias de muones y electrones, para posteriormente trabajar únicamente con los muones y así determinar la distribución de masa.

Cabe mencionar que los métodos usados para determinar la masa del bosón Z son similares a los que se utilizaron para determinar las propiedades del bosón de Higgs.

Con este trabajo contribuimos de manera significativa a la incorporación de la Universidad Michoacana al proyecto CMS para que los procesos de análisis e investigación sean efectuados en las instalaciones de la universidad por estudiantes y profesores, sin necesidad de hacer inversiones millonarias.

Finalmente, se desea despertar el interés en los profesores y estudiantes mostrando los avances tecnológicos para optimizar la investigación en Física de Altas Energías. Los beneficios del cómputo distribuido tendrán repercusión en el desarrollo científico en áreas como educación, medicina, bioinformática, nanotecnología, ingeniería y meteorología, por mencionar algunas.

Referencias

- [1] The Compact Muon Solenoid Technical Proposal. Collaboration. CMS. Marzo de 2010.
- [2] Grid File Access Library. Página Web: Consultado en enero de 2014.
http://grid-deployment.web.cern.ch/grid-deployment/documentation/LFC_DPM/gfal/.
- [3] Scientific Linux. Página Web: Consultado en enero de 2014.
<https://www.scientificlinux.org/>.
- [4] Pacini., Fabrizio. Job Description Language Attributes Specification for the gLite middleware (submission through Network Server. CERN Engineering Data Management. 2008.
- [5] The Globus Alliance. Página Web: Consultado en marzo de 2014. <http://www.globus.org>.
- [6] Adaptive Computing. Página Web: Consultado en febrero de 2014.
<http://www.adaptivecomputing.com/products/open-source/torque/>.
- [7] USCMS. Página Web: Consultado en febrero de 2014.
http://www.uscms.org/uscms_at_work/computing/status/status_condor.shtml.
- [8] The CERN Web Page . Página Web: Consultado abril 2014 <http://home.web.cern.ch/>.
- [9] The TIER Computing Model. Página Web: Consultado en abril de 2014.
“<https://twiki.cern.ch/twiki/bin/view/CMS/>”.
- [10] R. Housley, W. Ford y D. Internet Public Key Infrastructure, X.509 Certificate and CRL Profile. The Internet Society. Abril 2002.
- [11] Kesselman, I. Foster and C. “The Grid: Blueprint for a New Computing Infrastructure”. 2012.
- [12] Lorch M, Kafura D, Fisk I, Keahey K. Authorization and Account Management in the Open Science Grid Grid Computing. Mayo de 2005.
- [13] Corvo, M. CRAB. Mayo de 2014.
- [14] Foster I and Kesselman C. The Grid: Blueprint for a New Computing Infrastructure. San Francisco 2013.