Centro de Investigaciones Energéticas, Medioambientales y Tecnológicas
(CIEMAT)
Madrid

Departamento de Física Atómica, Molecular y Nuclear
Universidad Complutense de Madrid

Computación Grid para el experimento CMS del LHC

Memoria de tesis presentada por

D. José Caballero Béjar

para optar al grado de Doctor en Ciencias Físicas.

Dirigido por el

Dr. D. José María Hernández Calama

Madrid, Septiembre de 2007

Índice general

Prólogo

Una de las descripciones más completas de los constituyentes fundamentales de la materia y sus interacciones a altas energías es el Modelo Estándar de Partículas. Para probar su validez se han realizado todo tipo de experimentos, desde la medida sobre la violación de la conservación de la paridad en átomos a una transferencia de momento efectiva de $Q^2 \sim 10^{-10}$ GeV, hasta la colisiones en grandes aceleradores de haces de electrones y positrones, protones y antiprotones o electrones y protones, hasta una escala de varios cientos de GeV. Ejemplos de estos grandes colisionadores son LEP y SPS en el CERN, Tevatron en Fermilab o SLC en Standford.

Todas las medidas que se han realizado en estos experimentos están en gran concordancia con las predicciones del Modelo Estándar y han sido validadas con gran precisión. Sin embargo, varias cuestiones fundamentales permanecen aún sin explicar. El papel de la gravedad no está incluido en el modelo, por ejemplo. No explica ni el número ni las propiedades de las partículas elementales (introducidos como parámetros). Por ejemplo, el modelo no justifica por qué las cargas eléctricas del electrón y el protón son exactamente iguales en valor absoluto, por qué las fuerzas de las diferentes interacciones gauge son tan diferentes, el número de generaciones es tres, las anomalías de los sectores quarkónico y leptónico (aparentemente independientes) se cancelan, de dónde vienen las masas de los fermiones o por qué se rompe la invariancia CP y por qué en la escala de 246 GeV. Entre sus predicciones teóricas una nueva partícula, el bosón de Higgs, aún no ha sido observada. El Modelo Estándar introduce un campo escalar (el campo de Higgs) que rompe la simetría electrodébil a través de un mecanismo de ruptura espontánea y confiere masa a las partículas mediante la interacción entre éstas y dicho campo escalar. El bosón de Higgs sería la excitación de este campo escalar, y su existencia es de gran importancia pues validaría este método como el proceso que confiere las masas de las partículas. Otra deficiencia del modelo es que diverge a altas energías. Está claro que el Modelo Estándar sólo proporciona una descripción parcial de la naturaleza y varios modelos alternativos han sido propuestos. Algunos ejemplos son los modelos de dimensiones extra, technicolor, leptones pesados o los modelos de supersimetría (SUSY).

Con objeto de clarificar estas cuestiones aún abiertas, así como de validar o descartar los modelos alternativos propuestos, se está construyendo en el CERN un gran colisionador protón-protón, conocido como Large Hadron Collider (LHC). Operará a una energía en el centro de masas de 14 TeV, con una luminosidad nominal de 10^{34} cm^{-2}s^{-1}. Unos valores tan altos no tienen precedentes en los experimentos de Física de Partículas, por lo que el LHC supone un reto, no sólo científico sino también tecnológico.

Uno de los cuatro experimentos que operarán en el colisionador es CMS (Compact Muon Solenoid). De carácter multipropósito, está diseñado principalmente para encontrar el bosón de Higgs en un amplio rango de energías. CMS está preparado para operar en el más alto rango de luminosidades de LHC.

Los procesos de interés que se estudiarán en el LHC poseen una sección eficaz de producción muy baja. Por esto, y para poder acumular una gran cantidad de datos de procesos relevantes durante el tiempo de vida del colisionador, LHC operará a una gran frecuencia de cruce de haces, y acumulará una gran luminosidad integrada. Como consecuencia, el LHC producirá un volumen de datos muy superior al de los experimentos precedentes. Se estima que, sólo durante su primer año de funcionamiento, el volumen de datos generados será del orden de varios millones de Gigabytes. Procesar esta cantidad de información requerirá el uso de varias decenas de miles de PCs con la capacidad de procesamiento de un computador medio actual y

una organización novedosa de los recursos de computación y de los sistemas de procesamiento y análisis de datos.

El modelo de computación clásico de los experimentos de Altas Energías, basado principalmente en la acumulación de la mayor parte de los recursos computacionales en el laboratorio donde está instalado el acelerador, se muestra claramente inadecuado. Por una variedad de razones es difícil concentrar la ingente cantidad de recursos que se van a poner en juego en una única localización. Recursos, no sólo de hardware sino también, lo que es más importante, de personal cualificado, infraestructuras y servicios de soporte. En LHC, los institutos que componen los experimentos aportan localmente los recursos de computación, y dichos experimentos han diseñado un modelo computacional donde todos estos recursos, distribuidos geográficamente, están interconectados mediante redes de Internet de gran ancho de banda. Una nuevo conjunto de tecnologías, las llamadas **tecnologías Grid**, se encargan de operar estos recursos de manera coherente y transparente. Para el almacenamiento, transporte, procesamiento y análisis de los datos registrados por los experimentos de LHC se ha desarrollado el llamado LHC Computing Grid (LCG). El éxito en el desarrollo, integración y operaciones del modelo de computación de CMS es crucial para el análisis de los datos. Sin un sistema de computación Grid eficiente será imposible alcanzar los objetivos de física que el experimento se ha propuesto, entre ellos, el descubrimiento del bosón de Higgs del Modelo Estándar.

Este trabajo describe las tareas que se han llevado a cabo para el desarrollo, integración, test, despliegue y operación de este modelo computacional para el experimento CMS. Se ha desplegado toda la infraestructura necesaria para implementar el modelo en los centros españoles colaboradores en el experimento CMS. Las actividades en las que se ha participado de forma activa, descritas en esta memoria, son:

- **Desarrollo**: tanto de los sistemas de gestión de trabajos como del sistema de transferencia de datos. El primer punto incluye la migración completa del sistema de producción de datos Monte Carlo al entorno Grid y la colaboración en la creación de un segundo sistema de producción mejorado.
- **Integración**: mediante la participación en diversos ejercicios de computación ("challenges") diseñados para testear, a escala y complejidad crecientes, los recursos, servicios y componentes Grid.
- **Operaciones**: mediante la producción masiva de datos Monte Carlo, transferencias masivas de datos y diversas tareas de filtrado, reconstrucción y análisis de muestras.

Mediante estas actividades y ejercicios se ha podido implementar de forma eficaz toda la infraestructura de computación necesaria para el experimento. Se han obtenido conclusiones que han demostrado ser de gran utilidad para la mejora del software oficial de análisis del experimento, consiguiendo que sea capaz de operar más eficientemente en un entorno de trabajo ampliamente distribuido. También se ha contribuido para dotar a los centros de computación españoles para CMS del entorno Grid necesario para la generación de datos simulados, el procesamiento de datos reales y simulados, y el análisis de los mismos.

En el primer capítulo de esta memoria se hace una breve introducción al acelerador LHC y se describe el experimento CMS. Dada la enorme luminosidad del acelerador y la gran capacidad de filtrado del sistema online de trigger del experimento, se almacenará anualmente una ingente cantidad de datos de procesos relevantes. Se hace indispensable un sistema de computación capaz de procesar de forma eficaz esos datos.

El segundo capítulo presenta con detalle los elementos que componen LCG, el entorno de computación Grid para el LHC. Se describen las componentes necesarias para la gestión de los trabajos, la gestión de los datos y el procesamiento y análisis de los mismos. Una vez conocidos los elementos disponibles, se muestra la forma en que CMS ha organizado estos elementos en una estructura jerárquica de centros de computación: **el modelo de computación**.

El tercer capítulo describe en detalle las infraestructuras y servicios Grid LCG desplegados en los centros españoles. Se desarrollan los procesos de instalación, configuración, integración y operación de los mismos. Los centros españoles se han incorporado oficialmente al conjunto de los centros de computación de CMS.

En el cuarto capítulo se describe el proceso de desarrollo de los sistemas de producción de datos Monte Carlo y de transferencias de datos del experimento. El sistema de producción Monte Carlo fue completamente adaptado para trabajar en un entorno altamente distribuido, y se usó de forma intensiva para llevar a cabo parte de la producción oficial del experimento. La experiencia adquirida durante estas operaciones sirvió para mejorar el software de análisis del experimento, y para crear un sistema de producción más robusto, eficaz y escalable. También se ha colaborado intensamente en la creación de un sistema de transferencia de datos que palíe las carencias de las herramientas Grid disponibles, robusto, eficiente y con las prestaciones adecuadas para distribuir entre los centros de computación varios petabytes de datos anualmente.

En el quinto capítulo se describe la participación en varios ejercicios de computación, de escala y complejidad crecientes, pensados para ejercitar los sistemas de computación Grid de LCG, y de CMS en particular. Se han puesto a prueba los flujos de datos y los flujos de trabajos de procesamiento de datos. Se han realizado medidas del rendimiento de gran utilidad y se han identificado problemas que han servido para mejorar el sistema de computación de CMS.

En el sexto y último capítulo se presenta el nivel alcanzado en CMS en las operaciones de transferencias de datos y de ejecución de trabajos de procesamiento y análisis durante el último año. El nivel alcanzado es resultado directo de todo el trabajo desarrollado, y muestra que este esfuerzo ha sido de importancia capital para permitir **el análisis de los datos** desde el primer momento de funcionamiento del detector, y que es el objetivo final que se persigue.

Capítulo 1

El Large Hadron Collider y el experimento CMS

Se ha realizado una gran cantidad de experimentos para validar una de las descripciones más completas de los constituyentes fundamentales de la materia y sus interacciones a altas energías, el Modelo Estándar de Partículas (SM) [1, 2, 3]. Varios grandes aceleradores de partículas han permitido, en las últimas tres décadas, confirmar la mayoría de las predicciones realizadas por el Modelo Estándar.

Este modelo describe la materia como una composición de dos tipos de partículas, los leptones y los quarks, con spin semientero (fermiones). El primer grupo incluye al electrón, el muon y el tau, sus neutrinos respectivos, y sus antipartículas. El segundo grupo lo forman los quarks y antiquarks. Los quarks y antiquarks siempre aparecen en combinación, nunca de forma aislada, y no han sido observados en estado libre hasta ahora. Así, combinaciones de tres quarks forman los bariones (como el protón o el neutrón) y de un quark y un antiquark forman los mesones. Este modelo también describe la integración del electromagnetismo y las interacciones débiles en una única teoría unificada electrodébil o la descripción de las interacciones fuertes entre quarks a través de la cromodinámica cuántica.

Sin embargo, este modelo sólo ofrece una descripción parcial de la materia y sus interacciones. Con el objetivo de resolver las cuestiones que aún permanecen abiertas, de validar o descartar definitivamente algunos de los modelos alternativos propuestos [4, 5, 6, 7], y, en particular, confirmar la existencia del bosón de Higgs [8], se ha diseñado un nuevo acelerador de partículas, el *Large Hadron Collider* (LHC) [9].

1.1. El Large Hadron Collider

El LHC es un colisionador protón-protón de alta energía, que se está instalando en el Laboratorio Europeo para la investigación Nuclear (CERN) [10] en Ginebra (Suiza). Gracias a sus parámetros de diseño, y al rango de energías en el que podrá llegar a operar, el LHC ofrece un potencial físico impresionante, no sólo en lo referente al descubrimiento del bosón de Higgs, sino también en muchos otros campos de la física de partículas como son las medidas de precisión de la interacción electrodébil (como la medida precisa de la masa del bosón W o del quark top), la física de los mesones y bariones b, el plasma de quarks y gluones a través de las colisiones de iones pesados o los estudios de búsqueda de supersimetría.

Por razones económicas, LHC reutilizará el antiguo túnel de LEP [11], de 27 km de circunferencia. Estas dimensiones, junto con el máximo valor del campo de los dipolos que curvan los haces en el acelerador, limitan la energía en el centro de masas a 14 TeV. La tabla 1.1 muestra algunos parámetros del diseño de LHC.

Son necesarios estos valores tan altos de energía en el centro de masas (14 TeV) porque la energía de los haces de protones ha de ser muy superior a la energía de los procesos que se quieren estudiar. Esto es

Parámetro	valor
Energía en el centro de masas	14 TeV
Energía de inyección en el LHC	450 GeV
Número de partículas por paquete	$1{,}1x10^{11}$
Número de paquetes por anillo	2808
Luminosidad nominal	10^{34} cm^{-2}s^{-1}
Tiempo de vida de la luminosidad	10 h
Longitud de los paquetes	53 mm
Radio del haz en el punto de interacción	15 μm
Tiempo entre colisiones	24.95 ns
Frecuencia de cruce de haces	40.08 MHz
Frecuencia de interacción	1 GHz
Circunferencia	26.659 km
Campo magnético	8.3 T
Temperatura del imán	<2 K

Tabla 1.1: algunos parámetros del diseño del acelerador LHC

así porque son sus constituyentes (quarks y gluones) los que participan en las interacciones y sólo una fracción de la energía portada por los protones es realmente puesta en juego en las colisiones. La figura 1.1 muestra la sección eficaz para diferentes procesos en función de la energía en el centro de masas en colisiones entre protones. Puede verse en la figura que la sección eficaz de producción del bosón de Higgs aumenta rápidamente con la energía en el centro de masas. A la energía de LHC esta sección eficaz es, aproximadamente, dos órdenes de magnitud superior que a la energía de Tevatron. Para aumentar la probabilidad de producción del bosón de Higgs será preciso trabajar a la mayor energía en el centro de masas posible. La sección eficaz, σ, es directamente proporcional a la frecuencia de ocurrencia de un determinado suceso: $R = \mathfrak{L}\sigma$, donde el factor $\mathfrak{L}$ es conocido como luminosidad instantánea[1].

Para compensar unos valores tan bajos en la sección eficaz de producción del bosón de Higgs (unos 10 órdenes de magnitud inferior a la sección eficaz total de interacción protón-protón), el LHC deberá operar a la mayor luminosidad posible, lo que se conseguirá con un frecuencia de cruce de haces de 40 MHz y una focalización tremenda de los haces, produciéndose varias interacciones en cada cruce. LHC operará a una luminosidad inicial de 2 x 10^{33} cm^{-2}s^{-1} (régimen de baja luminosidad) que se incrementará hasta alcanzar un valor de 10^{34} cm^{-2}s^{-1} (régimen de alta luminosidad).

Cuatro experimentos van a operar en el LHC. LHCb (*Large Hadron Collider beauty experiment*) [12] se centrará en el estudio de los mesones y bariones b, analizando la violación de la simetría CP en la desintegración de estas partículas. ALICE (*A Large Ion Collider Experiment*) [13] estudiará colisiones de iones pesados de plomo a energías de $\sqrt{s} = 1150$ TeV y luminosidades del orden de $10^{27} cm^{-2} s^{-1}$. A esos valores de energía se espera la aparición de un nuevo estado de la materia, el plama de quarks y gluones. CMS (*Compact Muon Solenoid*) [14] y ATLAS (*A Toroidal LHC ApparatuS*) [15] son ambos experimentos de propósito más general pensados para estudiar la física de partículas a los valores más altos de energía y luminosidad del LHC, aunque sus diseños difieren significativamente, especialmente en las soluciones escogidas para la configuración del campo magnético. Finalmente, un pequeño experimento, TOTEM [16], que operará junto a CMS, estará dedicado a la medida de la sección eficaz total, scattering elástico y procesos difractivos en el LHC. La sección eficaz total será medida usando un método independiente de la luminosidad basado en la detección simultánea de scattering elástico a bajo momento transferido e

[1] la lumosidad da el número de colisiones que se producen en función de los parámetros de los haces que colisionan. Viene dado por la expresión

$$\mathfrak{L} = f\frac{n_1 n_2}{A}$$

donde f es la frecuencia de cruce de los haces, n_1 y n_2 son los números de partículas en cada haz, y A es el área efectiva de solapamiento de los dos haces.

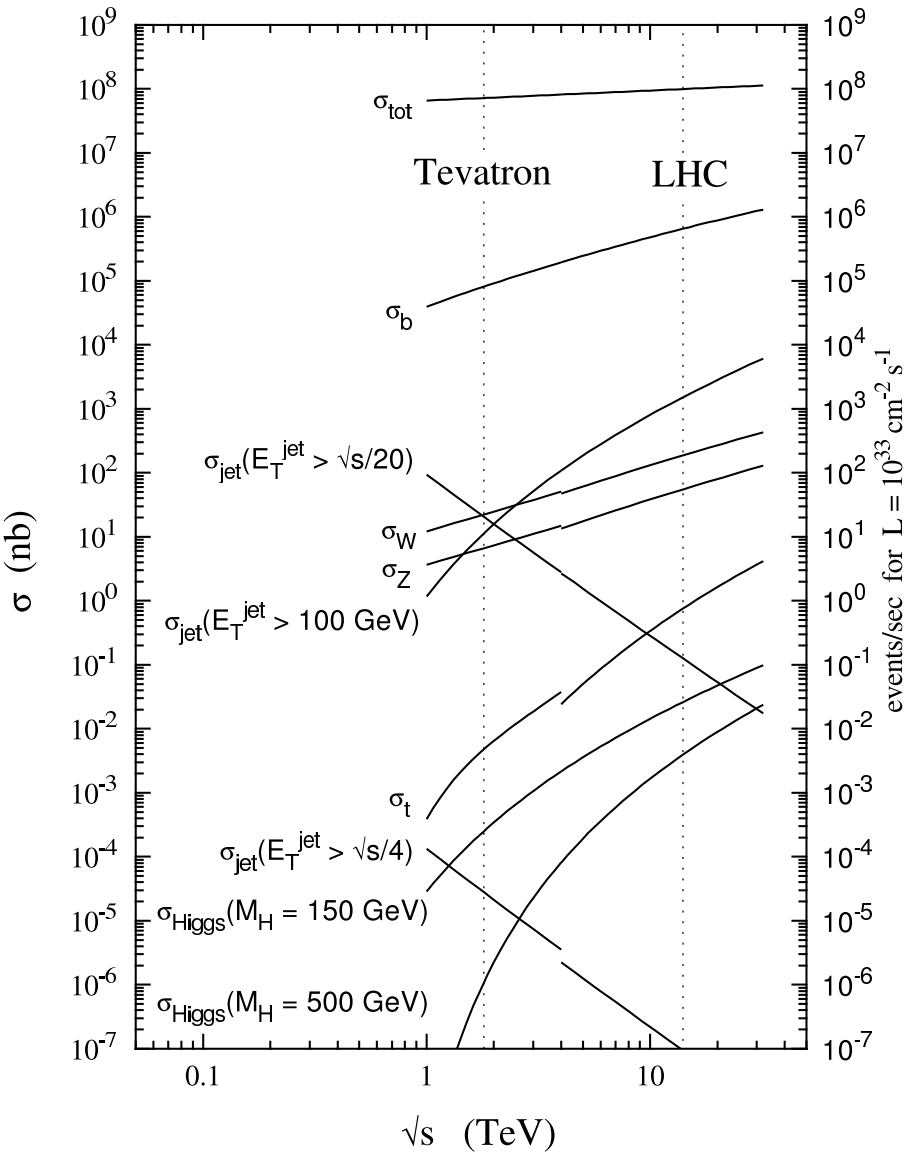


Figura 1.1: secciones eficaces y tasa de producción de varios procesos en función de la energía en el centro de masas en colisiones protón-protón.

interacciones inelásticas. Este método también proporciona, por tanto, una calibración absoluta de la luminosidad de la máquina.

1.2. El experimento CMS

CMS es un detector multipropósito que ha sido diseñado para aprovechar todo el potencial de LHC que podrá operar en el régimen de más alta energía y luminosidad del acelerador. La figura 1.3 muestra una vista del detector donde se puede apreciar su diseño final. Destacan sus dimensiones globales. Tendrá una longitud de 21.6 m, 14.6 m de diámetro y un peso de 12500 toneladas.

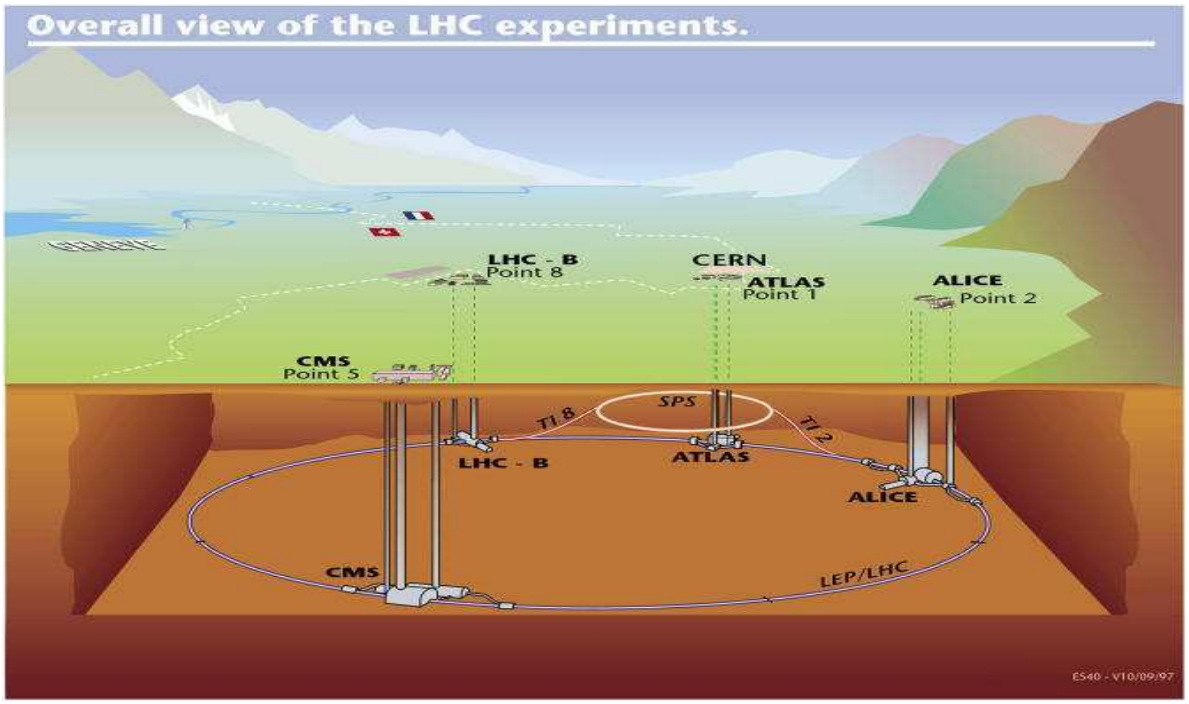


Figura 1.2: Esquema con la ubicación de los cuatro experimentos que operarán en LHC.

1.2.1. Objetivos de física y diseño

Los requisitos del diseño [17] de CMS para poder satisfacer los objetivos de física del LHC se pueden resumir de la siguiente forma:

- Buenas identificación de muones y resolución de momento sobre un amplio rango de momentos en la región $|\eta| < 2.5$ [2], buena resolución de la masa de los dimuones ($\approx 1\,\%$ a 100 GeV/c^2) y capacidad para determinar de forma inequívoca la carga de los muones con p<1 TeV/c.

- Buena resolución en el momento de las partículas cargadas y eficiencia de reconstrucción en la detección de trazas. Filtrado y clasificación de sucesos con τ y b-jets eficientes, lo que requiere de un detector de píxeles cercano a la región de interacción.

- Buenas resolución de la energía electromagnética y de la masa de difotones y dielectrones ($\approx 1\,\%$ a 100 GeV/c^2), cubriendo una amplia región espacial ($|\eta| < 2.5$). Correcta localización del vértice primario de interacción. Buen factor de rechazo de π^0 y aislamiento eficiente de fotones y leptones a altas luminosidades.

- Buena resolución de la masa de dijets 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 ($\Delta\eta$ x $\Delta\phi$ <0.1 x 0.1 [3]).

Para satisfacer estos requisitos CMS consta de los siguientes subdetectores (figura 1.4):

- un sistema de muones redundante y de excelentes prestaciones,

- el mejor calorímetro electromagnético consistente con este sistema de muones,

[2] la *pseudorapidez* η es una variable que se define a partir del ángulo polar Θ como $\eta = -ln(tg(\Theta/2))$, y tiene la ventaja de que se transforma de forma aditiva bajo transformaciones de Lorentz a lo largo del eje z.

[3] el ángulo ϕ es el ángulo en el plano transverso al haz.

- un calorímetro hadrónico hermético y de buenas prestaciones,
- un detector central de trazas de gran calidad,
- un imán superconductor solenoidal.

En la parte más interna se encuentra situado el detector central de trazas [18]. Esta formado por un detector de píxeles de silicio de 125 x 125 μm^2 que permiten una resolución espacial de entre 10 y 15 μm y un detector de tiras de silicio de 320 a 500 μm de anchura. Es capaz de reconstruir trazas de alto momento transverso con una eficiencia superior al 95 % en trazas aisladas y superior al 90 % en trazas dentro de jets en el rango $|\eta| < 2.6$, con una resolución en momento para leptones aislados de $\Delta p_t/p_t$=0.1 %.

El calorímetro electromagnético [19] está formado por cristales de $PbWO_4$, agrupados en un barril con un radio interno de 1.24 m y uno externo de 1.86 m. Para que la cascada electromagnética esté contenida en su totalidad en el calorímetro son necesarias al menos 26 longitudes de onda de radiación en los cristales para η=0. Para estos cristales esto corresponde a una longitud de 23 cm, aunque el tamaño final de cada cristal depende de su posición. La sección transversal de los cristales es de 22 x 22 mm^2. Con estas dimensiones de los cristales la granularidad del calorímetro es de 0.0175 x 0.0175 en $\Delta\eta$ x $\Delta\phi$, con una aceptancia que se extiende hasta $|\eta| = 3$. La resolución en energía en el rango entre 25 y 500 GeV viene dada por la expresión

$$\frac{\sigma_E}{E} = \frac{2{,}73}{\sqrt{E}} + \frac{142{,}2}{E} + 0{,}42$$

El calorímetro hadrónico [20] está formado por una sucesión de placas de cobre de 50 mm de espesor que actuarán como absorbentes, alternadas con plásticos centelleadores de 4 mm de espesor. Estas placas se agrupan en dos secciones cilíndricas de 4.3 m de longitud, cubriendo el rango $|\eta|$ <1.5. Con estas medidas, la granularidad que se consigue es de $\Delta\phi$ x $\Delta\eta$ = 0.87 x 0.87. La resolución en energía viene dada por la expresión

$$\frac{\sigma_E}{E} = \frac{(70-75)\,\%}{\sqrt{E}} + (7-9)\,\%$$

Finalmente, en la parte más externa se encuentra el espectrómetro de muones [21]. Consta de cuatro estaciones en el barril, distribuidas concéntricamente con respecto al haz, y cuatro estaciones en las tapas (*endcaps*), formadas por discos perpendiculares al haz. Estas dos componentes del sistema de muones constan de tecnologías diferentes debido a las diferencias en las condiciones del campo magnético y del flujo de partículas esperado. La parte del barril está formada por cámaras de tubos de deriva, mientras que el endcap lo componen *Cathode Strip Chambers* (CSC). Las CSC están formadas por planos de tiras de cobre detectoras en dirección radial que actúan como cátodos y permiten la medida en ϕ. Por cada plano de tiras hay un plano de hilos en posición transversal para la medida en θ. Las cámaras de tubos de deriva y las CSC se complementan con un sistema de detectores específicos llamados *Resistive Plate Chambers* (RPC), aptas para el *trigger* (sistema de filtrado en tiempo real) dada su buena resolución temporal, de 2 ns para intensidades del haz de partículas de hasta 6 kHz/cm^2. Este diseño es altamente redundante, y permite la detección de los muones con una eficiencias cercana al 100 % y medir sus propiedades con gran precisión.

En CMS se querrán medir estados resonantes de dimuones estrechos en el espectro de masa de hasta 1 TeV, para lo que se necesitará poder determinar, de forma inequívoca, la carga de las partículas a esa energía y conseguir una resolución en momentos del orden de $\Delta p/p \approx 10\,\%$ para p = 1 TeV. Para conseguirlo, un buen poder de curvatura de las trayectorias de las partículas se hace imprescindible. La elección tomada por CMS ha sido usar un imán solenoidal superconductor [22] que genera un campo magnético uniforme de 4 T, lo que ha hecho de CMS un diseño innovador. Este tipo de imanes proporciona un gran poder de curvatura a las trayectorias de las partículas cargadas sin necesidad de un solenoide de gran tamaño. Los calorímetros se encuentran situados en el interior del imán. El espectrómetro de muones se halla ubicado

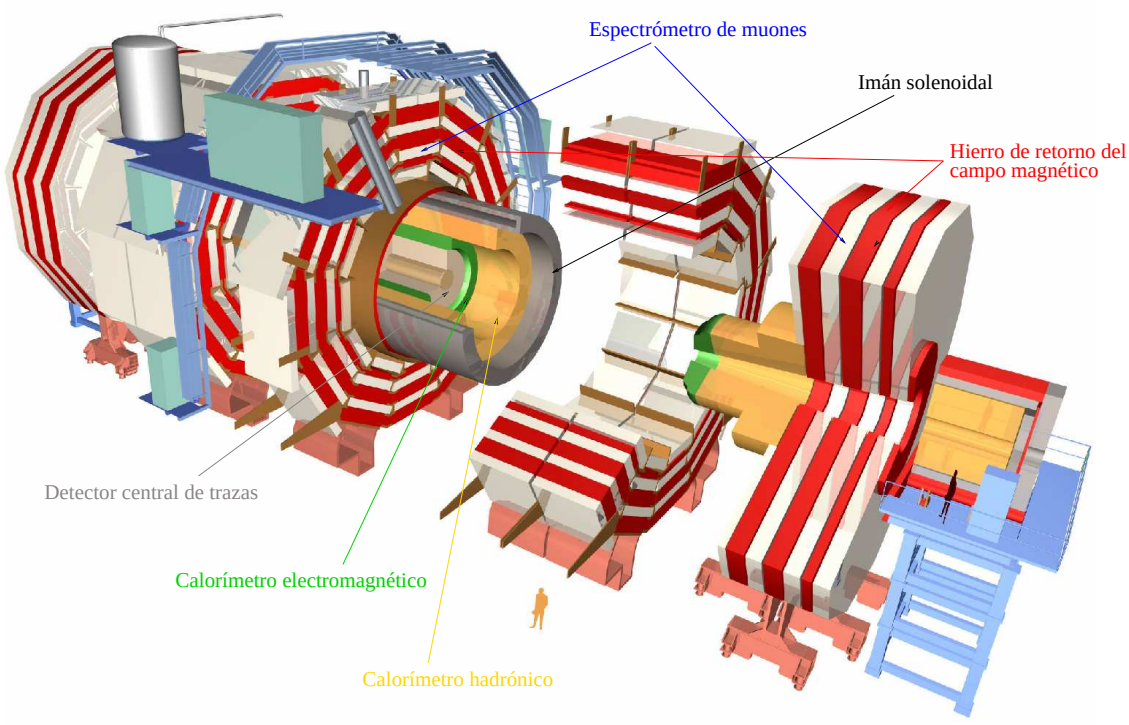


Figura 1.3: Esquema del experimento CMS.

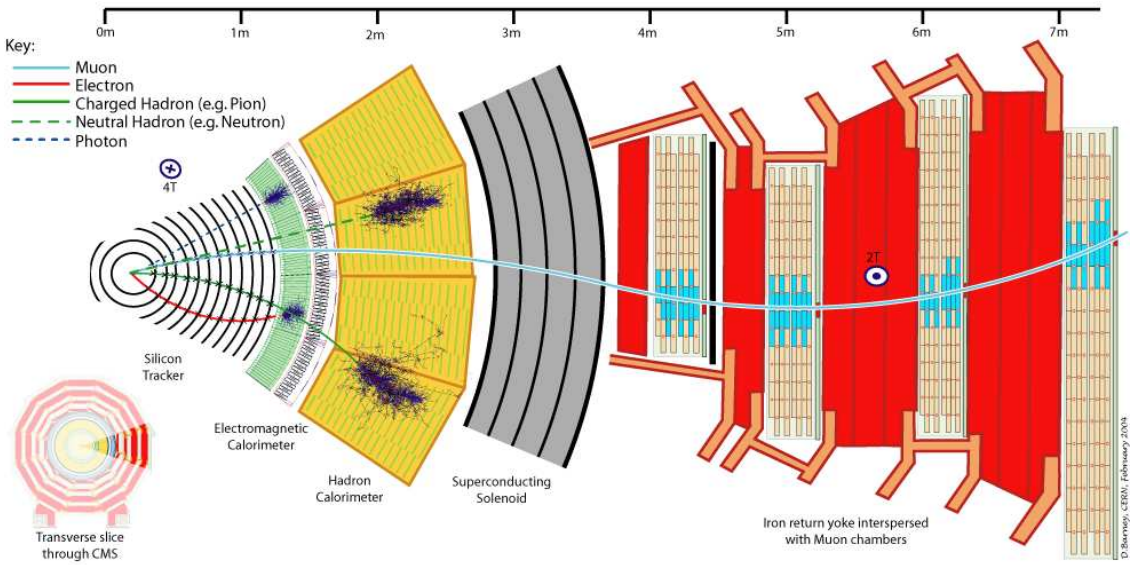


Figura 1.4: Subdetectores del experimento CMS.

entre el armazón de hierro que rodea a todo lo anterior (y que actuará como camino de retorno de las líneas del campo magnético).

Toda la información proporcionada por estos subdetectores será recogida por una cantidad de canales sin precedentes en los experimentos de Física de Altas Energías anteriores. La tabla 1.2 muestra la distribución de estos canales por subdetector. En total suman 54 millones y medio de canales, aunque aproximadamente un 98 % de esos canales se reparten entre el detector de píxeles de Silicio (44 millones) y el Tracker (9.3 millones).

Detector	**Número de canales**
Pixel	44 M
Tracker	9.3 M
Preshower	144 k
ECAL	83 k
HCAL	9 k
CSCs	500 k
RPCs	192 k
DTs	195 k
TOTAL	54.5 M

Tabla 1.2: Número de canales por subdetector.

Gracias a las buenas prestaciones del sistema de muones de CMS los procesos físicos con muones en su estado final adquieren gran relevancia en el experimento. Un ejemplo de este tipo de procesos es $H \rightarrow ZZ^{(*)} \rightarrow 4\mu$, uno de los canales más limpios para la búsqueda del bosón de Higgs del Modelo Estándar. Otros procesos con muones son $H \rightarrow WW^{(*)} \rightarrow 2\mu 2\nu$ y $Z' \rightarrow \mu\mu$. Pero no son los procesos con muones los únicos de interés que se estudiarán en CMS. Otros canales que también se podrán analizar son los que involucran electrones y fotones (como $H \rightarrow \gamma\gamma$ o $H \rightarrow ZZ^{(*)} \rightarrow 4e$), procesos con jets, mesones B y τ (como $B_s \rightarrow J/\psi\phi$, $H \rightarrow b\bar{b}$ o $H(A) \rightarrow \tau\tau$), etc.

1.2.2. Sistema de filtrado en tiempo real

A la luminosidad nominal del LHC, la frecuencia de colisiones es del orden de 10^9 Hz. Sin embargo, la tasa de colisiones en las que se producen procesos de interés es varios órdenes de magnitud inferior. Se hace necesario un sistema de filtrado que, en el menor tiempo posible, determine si las colisiones que se han producido en cada cruce de haces tienen interés físico o no. Puesto que el tamaño que ocupa la información correspondiente a cada suceso registrado por el detector es del orden del MB, y la frecuencia de cruce de haces es de 40 MHz, el ritmo total de producción de datos es de 40 x 10^6 MB/s. Almacenar y procesar toda esta información es inviable, por lo que esta selección de sucesos se ha de llevar a cabo de forma *on-line* (en tiempo real con el registro de los datos en el detector). El sistema encargado de esta selección online se conoce como *trigger*, cuyo esquema se puede ver en la figura 1.5.

El sistema de trigger consta de dos etapas, el *Level 1* (L1) [23] y el *High Level Trigger* (HLT) [24]. El L1 es capaz de reducir el flujo inicial de datos (de 40 MHz) hasta un ritmo inferior a 100 kHz. Debe tomar una decisión, aceptando o rechazando cada suceso, antes del siguiente cruce de haces; es decir, en menos de 25 ns. Sin embargo, gracias al hecho de que los datos que recoge el detector son temporalmente almacenados en varios canales en paralelo, cada uno de los cuales guarda información de 128 cruces, el trigger L1 puede tomar una decisión en un tiempo mayor, pero nunca superior a 3.2 μs (= 25 ns x 128). Para satisfacer estos requisitos tan fuertes L1 está implementado sobre un sistema hardware programable diseñado a medida.

La segunda etapa, el HLT, implementa una serie de algoritmos de software en una granja de varios miles de PCs, capaces de tomar una decisión en unos ms. Los datos procedentes de los distintos subdetectores se emsamblan en un suceso que se distribuye a los nodos de computación a través de una cascada de *switches* tipo Ethernet [25] de 1 Gigabit. Esta etapa reduce la frecuencia de sucesos aceptados a unos 100

- 150 Hz. Como el tamaño de los datos tras pasar por el HLT es de aproximadamente 1.5 MB, el flujo final de datos es de unos 225 MB/s (= 1.5 MB/suceso x 150 sucesos/segundo).

La tabla 1.3 recoge las estimaciones sobre el tiempo de toma de datos y luminosidad a los que operará el acelerador durante sus primeros años de funcionamiento. Multiplicando los 225 MB/s resultantes del trigger por los segundos de operación del LHC, obtenemos unos valores globales del orden de 225 MB/s x 10^7 s/año = $2{,}25 \cdot 10^6$ GB/año para guardar toda la información útil procedente del detector. También será necesario guardar una cantidad de datos similar producto de la simulación Monte Carlo. Serán necesarias varias decenas de miles de ordenadores actuales para procesar, simular y analizar tal ingente cantidad de datos. El volumen total de datos simulados y reconstruidos alcanzará un valor de varias decenas de Petabytes. En la misma tabla se muestran las necesidades en potencia de cálculo y capacidad de almacenamiento estimadas para los primeros años de operación del experimento CMS.

año	Tiempo de haz (segundos/año)	Luminosidad ($cm^{-2}s^{-1}$)	CPU (MSI2k)[4]	Disco (PB)	Cinta (PB)
2007	Primeras colisiones	-	21.9	4.1	5.4
2008	3 x 10^6	10^{32}	43.8	13.8	23.4
2009	10^7	2 x 10^{33}	67.2	23.3	41.5
2010	10^7	10^{34}	116.6	34.7	59.5

Tabla 1.3: Tiempo del haz, luminosidad y recursos de computación necesarios durante los primeros años de operación del LHC.

Dada la escala de los recursos de computación requeridos, el modelo de computación tradicional de los experimentos de física de altas energías, basado en la acumulación de la mayor parte de los recursos computacionales en el laboratorio donde está instalado el acelerador, resulta inadecuado. Por una variedad de razones es difícil concentrar la ingente cantidad de recursos que se van a poner en juego en una única localización. Recursos, no sólo de hardware sino también, lo que es más importante, de personal cualificado, infraestructuras y servicios de soporte. En LHC, los institutos que componen los experimentos aportan localmente los recursos de computación, y dichos experimentos han diseñado un modelo computacional donde todos estos recursos, distribuidos geográficamente, están interconectados mediante redes de Internet de gran ancho de banda. Una nuevo conjunto de tecnologías, las llamadas tecnologías Grid [26], se encargan de operar estos recursos de manera coherente y transparente.

[4]SpecInt2000 es una unidad de potencia de CPU definida por la *Standard Performance Evaluation Corporation* [27]. Un KSI2k equivale a un ordenador actual. Un MSI2k corresponde a 1000 KSI2k. Se ha comprobado empíricamente que esta unidad reproduce mejor que otros estimadores la eficiencia para los programas de simulación y procesado de datos secuenciales utilizados por los experimentos de LHC.

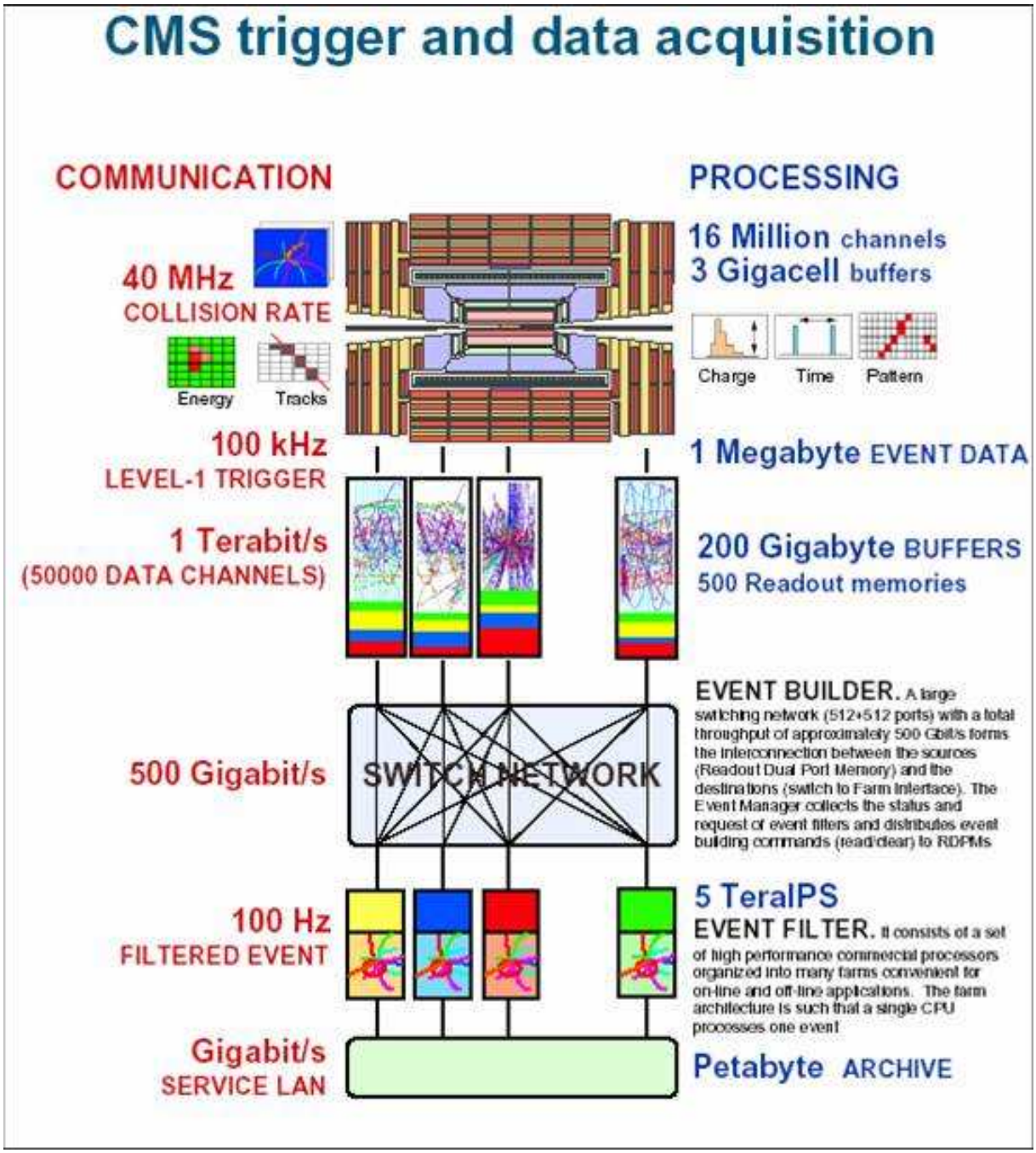


Figura 1.5: Esquema de los sistemas de adquisición de datos y de trigger de CMS. El sistema de trigger incluye las fases Level 1 y HLT.

Capítulo 2

Computación en el experimento CMS

2.1. Computación Grid

El experimento CMS deberá gestionar, procesar y analizar una ingente cantidad de datos, como se refleja en la tabla 1.3. Un complejo sistema de computación que integre todos los recursos disponibles será indispensable para llevar a cabo de manera eficiente estas tareas sobre este gran volumen de datos. No sólo se requiere una gran potencia de cálculo, sino que también son necesarios enormes recursos de almacenamiento masivo de datos, redes de Internet de gran ancho de banda y herramientas para gestionar su transferencia. También serán necesarios servicios y utilidades de software adecuados que, haciendo uso de los recursos computacionales disponibles, lleven a cabo todas las tareas de procesamiento y análisis sobre los datos. Los experimentos del LHC están realizando un gran esfuerzo para implementar estos nuevos modelos computacionales basándose en las tecnologías Grid.

El objetivo de las tecnologías Grid es unir de forma transparente y segura los recursos computacionales de diferentes dominios de administración respetando la autonomía de éstos (políticas internas de seguridad, herramientas de gestión propias, condiciones particulares de uso y administración, etc.) Estos recursos computacionales incluyen tanto la capacidad de cálculo y almacenamiento de los distintos centros involucrados como otro tipo de recursos como sensores, dispositivos de visualización, etc. El acceso a todos estos recursos debe poder hacerse de forma fiable, a través de una interfaz uniforme y desde múltiples localizaciones.

Para facilitar su gestión y administración, todos los recursos disponibles, los individuos y las instituciones se agrupan en colecciones dinámicas de forma flexible, segura y coordinada. Estas agrupaciones de recursos y usuarios reciben el nombre de Organizaciones Virtuales - *Virtual Organizations* (VO) -.

Las características que ha de satisfacer un sistema Grid son:

- Coordina recursos que no son objeto de un control centralizado. Un Grid integra y coordina recursos y usuarios procedentes de diferentes ámbitos.
- Usa protocolos[1] e interfaces[2] que son estándares, abiertos y de propósito general.
- Proporciona calidades de servicio[3] no triviales. Un Grid permite a sus recursos constituyentes ser usados de manera coordinada para proporcionar diferentes calidades de servicio, tales como el

[1] Un protocolo es un conjunto de reglas en un sistema de telecomunicaciones para el intercambio de información.
[2] Una interfaz es un conjunto de programas y métodos que permiten la intercomunicación entre sistemas.
[3] Un servicio es una entidad accesible via red que proporciona una capacidad específica; por ejemplo, la habilidad para mover ficheros, crear procesos o verificar permisos de acceso.

tiempo de respuesta, rendimiento, seguridad o disponibilidad o la asignación de múltiples recursos para satisfacer demandas complejas.

La solución computacional que han desarrollado e implementado los experimentos del LHC es conocida bajo la denominación de *LHC Computing Grid* (LCG) [28], y puede ser considerado como otro experimento más, dada su complejidad y magnitud.

2.2. Arquitectura del LCG

El LCG es una colección de recursos y servicios distribuidos geográficamente, agrupados en organizaciones virtuales, con el propósito de permitir a los distintos usuarios de los experimentos de LHC poder ejecutar de forma eficiente sus programas de generación, reconstrucción y análisis de sucesos, sin que exista la necesidad explícita de conocer el lugar concreto donde se encuentran almacenados los datos, dónde se ejecutarán dichos trabajos, o la ubicación donde se alojarán los resultados que produzcan. Estos usuarios pueden ser miembros individuales de los experimentos o grupos centrales de operaciones cuyos miembros disponen de ciertos privilegios y prioridades, o incluso la exclusividad en el uso de ciertos recursos. Son los experimentos los que otorgan, de forma dinámica, estas prioridades y privilegios con el objetivo de satisfacer determinados requisitos de carácter fundamental para toda la colaboración (como la producción masiva de datos simulados Monte Carlo, la distribución e instalación del software oficial del experimento, actividades de monitorización de los recursos disponibles, etc.)

Los componentes fundamentales del modelo de computación del LCG 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*-.
- Varios servicios de instalación y monitorización.

El WMS es el responsable de gestionar los trabajos enviados por los usuarios. Busca entre todos los recursos disponibles aquellos que satisfacen los requisitos concretos de un trabajo, y realiza las operaciones necesarias para enviar dicho trabajo al centro más apropiado según esos requisitos. Sabe en todo momento el estado del trabajo enviado, y permite al usuario recuperar el output que produce tras su finalización.

El DMS permite a los usuarios y al WMS realizar diversas operaciones con los ficheros de datos, tales como moverlos entre distintos sitios, replicarlos, eliminarlos, añadir nuevos o averiguar la ubicación de todas las copias disponibles. Estas operaciones se realizan haciendo uso de una serie de protocolos de transferencia de ficheros e interactuando con un conjunto de catálogos y bases de datos, tanto globales como locales.

El IS proporciona información sobre los recursos disponibles en todo momento a lo largo del Grid, y sobre su estado actual. Esta información es publicada por cada recurso individual y recopilada en un servicio de información global. El WMS hace uso de estos servicios para encontrar los recursos disponibles que satisfacen todas las necesidades de los trabajos de los usuarios. Asímismo, el DMS hace uso del IS para localizar los posibles recursos de almacenamiento de datos que haya disponibles.

El sistema de autenticación y autorización es el responsable de la seguridad en las operaciones del LCG. Mantiene una lista actualizada de los usuarios que han sido autorizados para hacer uso de los recursos y servicios del Grid, teniendo siempre en cuenta sus posibles privilegios. También verifica la autenticidad del usuario que ejecuta cada acción individual mediante un sistema de certificados de seguridad.

También hay disponibles una serie de recursos de monitorización que contabilizan el uso de los recursos disponibles, permiten a los usuarios obtener información sobre sus trabajos en ejecución o chequean el estado de los distintos servicios Grid.

El entorno de trabajo (o *framework*) sobre la que actualmente se desarrollan las componentes de computación Grid para el LHC recibe el nombre de gLite [29]. gLite nace de los proyectos EGEE [30] y WLCG (Worldwide LCG), aunque incluye componentes de otros proyectos, entre los que se encuentran DataGrid (EDG) [31], DataTag (EDT), Globus [32] o Condor-G [33], por ejemplo. El *middleware*[4] de gLite contiene componentes para implementar los distintos servicios de la computación Grid para el LHC, como son el middleware básico que gestiona las operaciones a nivel más fundamental, utilidades para la instalación de los servicios, los módulos para la autenticación y autorización, el WMS y el DMS, o las herramientas de monitorización e información.

2.2.1. Gestión del flujo de trabajo

Como se ha comentado en el apartado anterior, el WMS ha sido desarrollado por EDG, Condor-G y Globus y ha sido adaptado para satisfacer las necesidades específicas del LCG.

Para que un usuario pueda hacer uso de los recursos Grid disponibles debe poseer un certificado X.509[5] que ha de ser concedido por una Autoridad Certificadora -*Certification Authority* (CA)- aceptada por la VO. La misión de las CAs es garantizar la identidad de los usuarios y comprobar su derecho a poseer un certificado. Una vez que el usuario posee un certificado, debe crear a partir de él una credencial conocida generalmente como *proxy certificate*, o simplemente *proxy*. Esta credencial autentifica al usuario en todas las interacciones securizadas. Por razones de seguridad su tiempo de vida es limitado y el usuario debe volver a generarla cada vez que la necesite.

La figura 2.1 muestra las componentes que intervienen en la gestión de los trabajos. La componente que permite a los usuarios acceder a las funcionalidades del Grid del LCG se conoce como *User Interface* (UI), y usualmente se identifica con la máquina donde está instalada. Desde la UI un usuario puede ser autenticado y autorizado para usar los recursos del Grid y acceder a las funcionalidades básicas ofrecidas por el WMS, DMS y IS:

- listar los recursos disponibles para ejecutar un trabajo determinado,
- enviar trabajos para su ejecución,
- mostrar el estados de los trabajos en ejecución,
- cancelar trabajos,
- recuperar el output de los trabajos finalizados,
- recuperar cierta información de monitorización de los trabajos,
- copiar, mover y borrar ficheros en el Grid.

[4] El middleware es un software de conectividad que ofrece un conjunto de servicios que hacen posible el funcionamiento de aplicaciones distribuidas sobre plataformas heterogéneas. Funciona como una capa de abstracción de software distribuida, que se sitúa entre las capas de aplicaciones y las capas inferiores (sistema operativo y red). Permite abstraerse de la complejidad y heterogeneidad de las redes de comunicaciones subyacentes, así como de los sistemas operativos y lenguajes de programación.

[5] Estándar usado en sistemas de criptografía de claves públicas - *Public Key Infrastructure* (PKI) [34] -. Especifica, entre otras cosas, formatos estándar para certificados de claves públicas y un algoritmo de validación de la ruta de certificación. El objetivo de esta especificación es desarrollar un perfil para facilitar el uso de los certificados X.509 en aplicaciones de Internet para aquellas comunidades que deseen hacer uso de la tecnología X.509. Los usuarios de una llave pública desean tener confianza en que la llave privada asociada está en manos del sujeto (una persona o un sistema) adecuado, con quien se usará un mecanismo de firma digital o de encriptación. Esta confianza se obtiene a través del uso de los certificados de llave pública.

El conjunto de características y requisitos que describen un trabajo particular que se va a enviar al Grid se especifican en un fichero especial mediante una sintaxis muy definida, conocida como *Job Description Language* (JDL) [35]. El JDL está formado por un conjunto finito de instrucciones que asignan valores a una serie de atributos. Ejemplos de estos atributos son ciertas restricciones o condiciones sobre las características particulares del centro donde se ejecutará el trabajo o la necesidad de determinadas versiones del software.

Los ficheros que un trabajo pueda necesitar durante su ejecución también pueden enviarse desde la UI junto con dicho trabajo. El conjunto de todos los ficheros que son enviados recibe el nombre de *Input Sandbox*. De igual forma, los ficheros generados por el trabajo tras finalizar su ejecución y que son devueltos al usuario recibe el nombre de *Output Sandbox*. Existen, sin embargo, una limitación en el tamaño máximo del Input y del Output Sandbox. En el caso de que los datos de input o de output sean demasiado grandes, éstos se hacen disponibles a través de un sistema de almacenamiento de datos. La lista de ficheros incluidos en el Input y el Output Sandbox deben estar listados en el fichero JDL.

Los siguientes servicios se ejecutan en una máquina conocida como *Resource Broker* (RB). Es la máquina donde están instalados los servicios WMS. El *Network Server* (NS) acepta peticiones que le llegan desde una UI, auntentifica al usuario, copia el Input Sandbox y el Output Sandbox entre la UI y el RB, y redirige las peticiones al *Workload Manager* (WM). De forma opcional, el proxy del usuario puede estar registrado para su renovación de forma periódica por parte del *Proxy Renewal Service*.

Cuando un trabajo es enviado para su ejecución, el WM usa los servicios de *Matchmaking* para encontrar los recursos que mejor cumplen los requisitos especificados en el fichero JDL. Para tal fin, el WM interacciona con el IS y con un servicio que contiene un registro con la localización de los datos, conocido como *Replica Location Service* (RLS) [36].

Una vez que se han encontrado recursos que cumplen los requisitos especificados en el fichero JDL, el

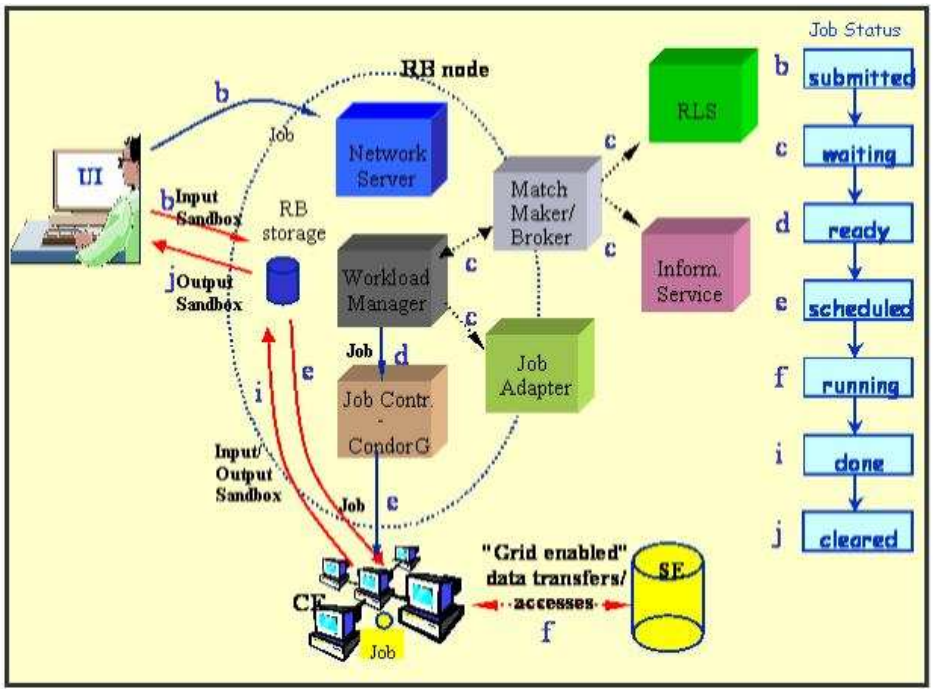


Figura 2.1: Componentes del flujo de trabajos en LCG.

trabajo es enviado a un *Computing Element* (CE) para su ejecución. El CE es la interfaz del Grid con un cluster de computación. Está implementado sobre un conjunto de nodos de computación, llamados *Worker Nodes* (WN), un Sistema de Gestión de Recursos Local -*Local Resource Management System* (LRMS)-, y un nodo que actúa como front-end[6] para el resto del Grid, conocido como *Grid Gate* (GG) o *Gatekeeper*. Al CE también se le envía un trabajo adicional por cada usuario, el *grid monitor*, para monitorizar el estado de ejecución de los trabajos de los usuarios. Mientras que los Worker Nodes sólo necesitan conectividad hacia fuera, el Grid Gate también debe ser accesible desde el exterior del centro. El Grid Gate es el responsable de aceptar trabajos y distribuirlos para su ejecución en los WNs, y proporciona una interfaz uniforme con los recursos de computación que maneja. En los Worker Nodes deberán estar disponibles todos los comandos y librerías que ejecutan acciones sobre los recursos y datos del Grid. En cada centro del LCG hay disponible al menos un Computing Element y una granja de Worker Nodes detrás suyo.

Se están incorporando nuevas funcionalidades a la gestión de los trabajos en gLite, entre las que destacan dos. La primera permite la ejecución de varios trabajos encadenados (cuando el output de uno es el input de otro, por ejemplo), especificando el orden de precedencia, de forma que un trabajo no comienza hasta que los anteriores no hayan finalizado. La otra funcionalidad permite el envío, seguimiento y recuperación del output de grupos de trabajos. El usuario manda un sólo trabajo al RB, con el consiguiente ahorro en tiempo, y es el RB el encargado de crear, a partir de los parámetros especificados por el usuario, el conjunto completo de trabajos que finalmente se ejecutarán. Esta opción es muy útil para aquellos trabajos que son idénticos salvo algún parámetro. Estas funcionalidades reciben el nombre de *bulk operations*.

2.2.2. Gestión de datos

El DMS se basa en dos componentes fundamentales: los servicios de catálogos y las unidades de almacenamiento, o *Storage Element* (SE). Los Storage Elements son los servicios que permiten a los usuarios y a las aplicaciones guardar datos para su futura recuperación. Por tanto, todos los datos guardados en un SE se deben considerar de sólo-lectura, y no pueden ser modificados (salvo que sean borrados y reemplazados por datos nuevos). El SE proporciona acceso uniforme a los recursos de almacenamiento. Puede gestionar simples servidores de disco, grandes conjuntos de discos (conocidos generalmente como *pool*[7] de disco), o bien sistemas de almacenamiento jerárquico (MSS) que ofrecen una copia de seguridad en cinta de los datos que guardan. Cada centro en LCG proporciona, al menos, un SE.

2.2.2.1. Catálogos de datos

Es la parte del DMS que permite conocer la existencia y ubicación de los datos. Existen catálogos de datos (RLS) y de metadatos[8] (*Replica Metadata Catalogue* -RMC-). Estos servicios permiten a los usuarios conocer qué ficheros hay disponibles, cuántas réplicas de cada uno de ellos existen y dónde están guardadas, y la información necesaria para saber cómo acceder a ellos. Existen distintos tipos de nombres para referirse a un fichero, dependiendo de la información que aporta cada una de ellas:

- *Grid Unique Identifier* (GUID), identifica a un fichero de forma unívoca.
- *Logical File Name* (LFN), es un nombre lógico o alias que puede usarse en lugar del GUID para referirse a un fichero.
- *Storage URL*[9] (SURL), también conocido como *Physical File Name* (PFN), identifica una réplica concreta en un Starage Element. Contiene toda la información necesaria para acceder a un fichero (como, por ejemplo, el protocolo de acceso). El SURL es, en principio, invariable con el tiempo, pues es una entrada en los catálogos de ficheros.

[6] Parte de un sistema que interacciona con el exterior, normalmente los usuarios.
[7] Un pool es un grupo de sistemas de ficheros localizados en uno o más servidores de disco
[8] Datos sobre los datos: definen la estructura y significado de los ficheros de datos.
[9] *Uniform Resource Locator*. Es una secuencia de caracteres, de acuerdo a un formato estándar, como documentos e imágenes en Internet, que se usa para nombrar recursos por su localización.

- *Transport URL* (TURL), un URI[10] válido con la información necesaria para acceder a un fichero en un Storage Element. El TURL se obtiene dinámicamente a partir del SURL a través del Sistema de Información, por lo que puede cambiar con el tiempo.

Los catálogos RLS guardan la equivalencia entre GUIDs y PFNs, mientras que en los catálogos RMC se puede encontrar, junto a otros atributos de los ficheros, la conversión entre GUIDs y LFNs.

2.2.2.2. Acceso a los datos

Los Storage Elements pueden dar soporte a diferentes protocolos para el acceso a los datos y mediante interfaces diferentes. Existen versiones de estos protocolos con y sin herramientas de seguridad GSI[11] Estas herramientas GSI permiten gestionar los certificados Grid (proxy) de los usuarios.

Los protocolos de acceso y de transferencia de datos más usuales, recopilados en la tabla 2.1, se detallan a continuación:

- **GSIFTP:**[12] ofrece las funcionalidades del protocolo FTP, pero añade las herramientas de GSI para la autenticación de los usuarios. Implementa las transferencias de ficheros desde/hacia los SE de forma segura, rápida y eficiente. Proporciona control de las transferencias de datos entre terceros, así como paralelización en dichas transferencias. Todos los SE operan al menos un servidor GridFTP.
- ***Remote File Input/Output protocol* (RFIO):** permite el acceso remoto de forma directa a los ficheros guardados en los SE. Fue diseñado para acceder a sistemas de archivado con cinta (como, por ejemplo, CASTOR). RFIO implementa una versión de las llamadas estándar POSIX[13]. Existen versiones con y sin herrramientas de seguridad. La versión que incluye las funcionalidades GSI se conoce como **gsirfio**.
- ***dCache Access Protocol* (dcap):** al igual de RFIO, también permite el acceso remoto directo al SE. Es un versión del protocolo nativo de los SE basados en dCache (dcap). Puede incluir herramientas de seguridad GSI, recibiendo entonces el nombre de **gsidcap**. Al igual que RFIO, dcap también permite acceso POSIX a los datos.

Protocolo	Descripción	Seguridad GSI	Opcional
GSIFTP	Transferencia Tipo FTP	Sí	No
dcap	Acceso POSIX remoto a ficheros	No	Sí
gsidcap	Acceso POSIX remoto a ficheros	Sí	Sí
RFIO no seguro	Acceso POSIX remoto a ficheros	No	Sí
RFIO seguro	Acceso POSIX remoto a ficheros	Sí	Sí

Tabla 2.1: Versiones de los protocolos de acceso y transferencia de datos usados más comúnmente en LCG.

La mayoría de los recursos de almacenamiento son gestionados por un servicio llamado *Storage Resource Manager* (SRM). El diagrama 2.2 muestra las distintas funcionalidades que ofrece el servicio SRM, y su acoplamiento con los demás componentes del DMS. SRM es un servicio middleware que ofrece funcionalidades como la migración transparente de ficheros de disco a cinta, retención de ficheros en disco

[10] *Uniform Resource Identifier*. Texto corto que identifica unívocamente cualquier recurso (servicio, página, documento, dirección de correo electrónico, etc.) accesible en una red.

[11] *Grid Security Infraestructure*.

[12] En la literatura, los términos GSIFTP y GridFTP se suelen usar indistintamente. Estrictamente hablando, GSIFTP es un subconjunto de GridFTP.

[13] *Portable Operating System Interface* para lectura y escritura de ficheros, familia de estándares de llamadas al sistema operativo con el objetivo de generalizar las interfaces de los sistemas operativos para que una misma aplicación pueda ejecutarse en distintas plataformas. Estos estándares surgieron de un proyecto de normalización de las API y describen un conjunto de interfaces de aplicación adaptables a una gran variedad de implementaciones de sistemas operativos.

(para impedir que el sistema automático de limpieza los borre al superar el tiempo máximo permitido de estancia en el disco) o la reserva anticipada de espacio de almacenamiento. También ofrece la posibilidad de transferencias entre diferentes puntos finales gestionadas por un tercero. Permite el crecimiento del espacio de almacenamiento simplemente añadiendo más discos. Existen varias implementaciones de los servicios SRM, que pueden depender de un SE a otro, y que ofrecen diferentes posibilidades. Sin embargo, proporciona una interfaz común para los SE, accesible en forma de servicio web[14].

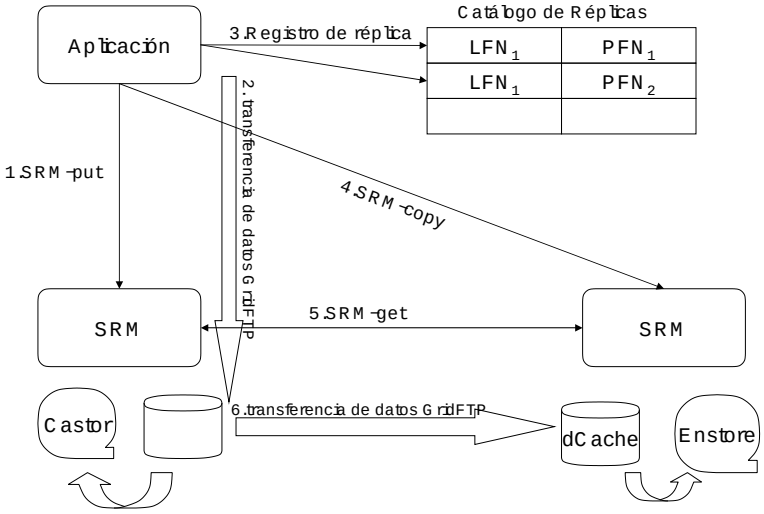


Figura 2.2: Funcionalidades del Storage Resource Manager.

El *File Transfer Service* (FTS) es el servicio de bajo nivel encargado de las transferencias de datos. Su objetivo es conseguir transferencias fiables de ficheros entre centros. Los usuarios pueden programar copias de ficheros entre origen y destino (punto a punto, sin enrutamiento entre nodos intermedios), mientras que los centros involucrados en la transferencia pueden controlar el uso del ancho de banda disponible. Estas transferencias se realizan de forma asíncrona.

FTS maneja internamente la negociación SRM entre los SE de origen y destino, así como la gestión de las transferencias GridFTP subyacentes. De esta forma, cuando un usuario (o una aplicación) solicitan una acción de transferencia, el servicio FTS se pone en contacto con los dos SRMs que gestionan los Storage Elements involucrados. Finalmente, los datos se copian haciendo uso del protocolo GSIFTP. El servicio FTS se puede configurar de forma que se delega en los dos servicios SRM la gestión de esta transferencia FTP.

Para gestionar las transferencias, el servicio FTS puede definir canales en los que se pueden configurar algunos parámetros tales como el número máximo de transferencias simultáneas, el paralelismo de cada una de ellas en la transferencia GridFTP, o el reparto del canal entre distintas VOs. El servicio FTS sólo acepta SURLs como valores de origen y destino, pues es insensible a la organización de datos particular de cada VO y no gestiona GUIDs, LFNs, ni conjuntos de datos agrupados por su contenido de Física (*Datasets*). La figura 2.3 muestra todas las componentes de un SE involucradas en las transferencias de datos, así como los flujos de datos y de información en dichas transferencias.

Con el objetivo de hacer transperente a los clientes el uso de las componentes de software y operaciones necesarias para interaccionar con los sistemas de almacenamiento (servicios SRM, catálogos de réplicas, conversiones entre los distintos tipos de nombres de ficheros y los diferentes protocolos POSIX de acceso a los SE) se ha desarrollado una herramienta, llamada *Grid File Acccess Library* (GFAL) [37], que

[14]Los servicios web son conjuntos de aplicaciones, tecnologías, protocolos y estándares, con capacidad para interoperar en la Web, que sirven para el intercambio de datos entre aplicaciones.

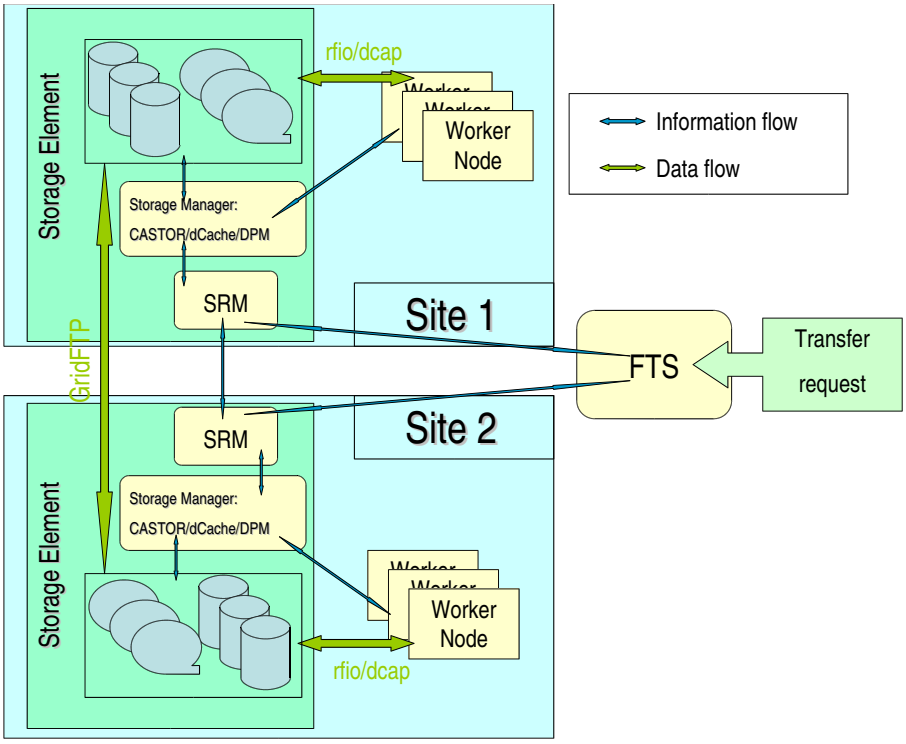


Figura 2.3: Esquema con los componentes de los SE, y los flujos de información y de datos en las transferencias entre dos SE distintos.

oculta estas interacciones y presenta una interfaz POSIX única para las operaciones de input/output. Esta interfaz única soporta los protocolos rfio, dcap y los de acceso local (tipo NFS [38]) y todos los tipos de nombres de ficheros (LFN, GUID, SURL, TURL). Permite, por tanto, el acceso a un fichero sin necesidad de conocer su PFN ni el tipo de protocolo concreto del sistema donde se aloja. Averigua el TURL apropiado de acceso a los datos en cada caso consultando al sistema de información y a los catálogos de ficheros.

2.2.2.3. Tipos de Storage Element

Se han desarrollado distintos tipos de SE, cada uno de los cuales ofrece distintas funcionalidades para satisfacer las necesidades de los distintos centros y experimentos. La calidad y tipo de los servicios que ofrecen los centros depende generalmente de su tamaño, y los distintos experimentos tienen sus propias políticas de gestión de datos, (como la ubicación de los datos en discos o en cintas, replicación de ficheros, etc.) Para satisfacer todas estas necesidades heterogéneas, se ha implementado varias modalidades de SE, que se resumen en la tabla 2.2.

Tipo	Recursos	Protocolo de transferencia	Protocolo de acceso	SRM
SE clásico	Servidor de disco	GSIFTP	RFIO no seguro	No
DPM	Pool de disco	GSIFTP	RFIO seguro	Sí
dCache	Pool de disco/MSS	GSIFTP	dcap/gsidcap	Sí
CASTOR	Pool de disco/MSS	GSIFTP	RFIO, seguro y no seguro	Sí

Tabla 2.2: Implementaciones de Storage Elements utilizadas en LCG.

***CERN Advanced STORage manager* (CASTOR)**

consiste en un *buffer*[15] de disco que actúa como front-end de un sistema de almacenamiento masivo en disco. Las complejidades inherentes a la configuración de los discos y las cintas es ocultada al usuario, que ve todo el sistema de almacenamiento bajo un espacio de nombres virtual. El protocolo de acceso nativo, RFIO, permite el acceso directo a los ficheros en el SE. Las versiones más recientes de CASTOR incluyen el protocolo RFIO con herramientas de GSI. El sistema puede ser configurado para que use sólo los discos, sin migración a cinta.

Uno de los componentes fundamentales de CASTOR es el llamado *stager*, encargado de gestionar el pool de discos. Entre sus funciones están encontrar espacio en disco para guardar cada fichero, mantener un catálogo con todos los ficheros almacenados en todos los discos, y borrar los ficheros que han sido accedidos menos recientemente cuando sea necesario crear más espacio libre del que hay disponible (este proceso se conoce con el nombre de *garbage collection*). Dado que el stager debe ser el único proceso que crea y borra ficheros en los discos, conoce en todo momento el espacio que hay libre en todos los discos del sistema. Para evitar una degradación en el rendimiento cuando el número de pools de discos aumenta y por motivos de redundancia, evitando una parada total del sistema cuando sea necesario ejecutar operaciones de mantenimiento, es conveniente la presencia de varios stagers simultáneamente. La figura 2.4 muestra un diagrama con los elementos con los que el stager se comunica para atender las peticiones cliente RFIO.

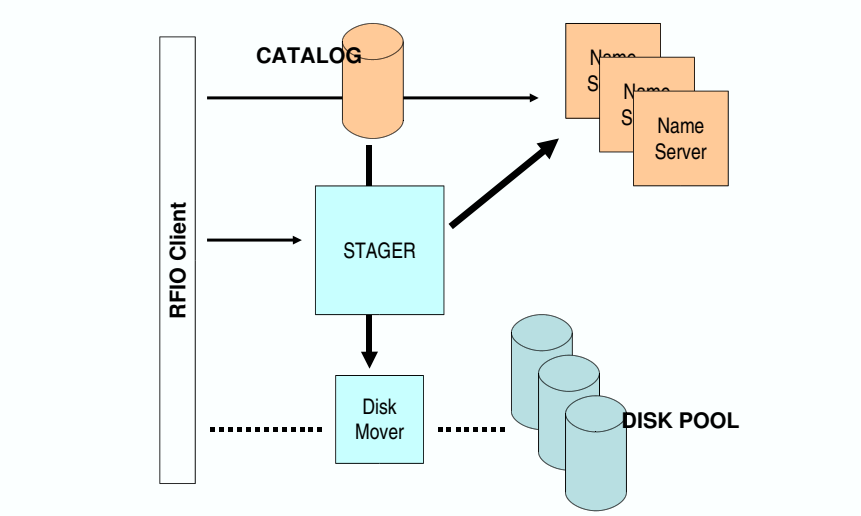


Figura 2.4: Esquema del funcionamiento del stager de CASTOR.

Para nombrar a los ficheros en CASTOR se usan, por conveniencia, nombres lógicos de la forma /castor/nombre_de_dominio/experimento/... (imitando la estructura de nombres de AFS [39]) La implementación de una visión jerárquica de este espacio de nombres de forma que los ficheros aparezcan organizados con esta estructura de directorios la lleva a cabo el *name server*. El name server también se encarga de recordar la localización de los ficheros que se encuentran en cinta tras haber sido migrados desde el pool de discos. El name server también mantiene la definición *fileclass* de los ficheros. El fileclass es un conjunto de atributos que describen a los directorios de ficheros, y es especialmente importante porque especifica si un fichero puede o no ser migrado y cómo, y si puede o no ser purgado durante el proceso de garbage collection. Algunos de estos atributos son el tiempo entre migraciones, el tiempo mínimo entre migraciones, el número de copias, el tiempo de retención en disco, el número de *streams*[16] en paralelo

[15] Ubicación de la memoria para el almacenamiento temporal de información.

[16] el término streaming hace referencia a la transmisión de datos (usualmente video o audio) a través de una red en tiempo real.

para las migraciones, o los pools de cintas (para repartir las copias en diferentes localizaciones físicas por motivos de seguridad y de rendimiento). Los ficheros pueden estar segmentados en varias partes (incluso en distintas cintas) permitiendo aprovechar mejor el espacio total disponible en la cinta o la existencia de ficheros de tamaño mayor que el de una cinta individual. De las transferencias de ficheros de disco a cinta, y viceversa, se encarga una componente llamada *Remote Tape COPY* (RTCOPY). RTCOPY es *multi-thread* [17], con un thread gestionando el input/output en la cinta mientras los demás threads se encargan de los discos. El servicio encargado de buscar las cintas disponibles para la migración, de conocer su estado y saber si están en uso en cada momento, se llama *Volume Drive Queue Manager* (VDQM).

Las políticas de migración se definen a través de pool de discos y de las definiciones fileclass de los directorios. Usualmente se tienen políticas de migración diferentes los pools de discos públicos (compartidos) y para los pools de discos del experimento. La configuración de cada pool de disco incluye una especificación que permite o no la migración de los ficheros. La migración se lleva a cabo cuando la cantidad de datos listos para ser migrados supera un cierto umbral, si el porcentaje de espacio libre en los discos cae por debajo de un cierto límite, si un determinado intervalo de tiempo ha sido especificado en el fileclass, o bajo demanda. Un fichero que es candidato para ser migrado no será borrado durante el proceso de garbage collection hasta que la migración haya finalizado.

Es posible tener múltiples copias de un fichero en varios pools de disco, pero sólo puede haber una copia por pool. En este caso, sólo una copia es de escritura/lectura y las demás serán de sólo-lectura. Si la copia de escritura/lectura cambia y es migrada a cinta las copias de sólo-lectura tardarán un cierto tiempo en ser actualizadas. Un caso típico es el de un registro central de datos, o *Central Data Recording* (CDR). La primera instancia de los ficheros procedentes del CDR irá a un pool determinado. Para conseguir un buen rendimiento de este proceso de escritura de datos, este pool no será entonces utilizado para tareas de análisis, sino que se hará uso de un segundo pool. El stager se configura entonces para que haga una copia de los ficheros desde el primer pool al segundo, donde esta copia de sólo-lectura es la que se usará para análisis. Si la copia en el primer pool cambia y una aplicación accede a la segunda, ésta última es automáticamente actualizada al no cumplirse algún criterio de control establecido (como que el tamaño de ambos ficheros sea diferente).

La figura 2.5 muestra un esquema con la arquitectura de CASTOR, donde se pueden ver todas sus componentes y las interrelaciones entre ellas.

En CASTOR, las peticiones se gestionan a través de un gestor de colas de trabajos llamado *Load Sharing Facility* (LSF) [40]. LSF es un producto para gestionar colas de trabajos para la ejecución controlada de grandes cantidades de trabajos en un cierto número de máquinas de acuerdo con las políticas del centro. Es un sistema para computación distribuida capaz de conectar un grupo de computadores en red formando un sistema único para conseguir un mejor uso de todos los recursos disponibles. Cuando una máquina queda libre se selecciona un trabajo para su ejecución en función de un cierto número de criterios como el estado de la carga de tareas en las distintas máquinas, el tiempo que el trabajo ha estado en espera, o los recursos solicitados por el trabajo (como el tiempo de CPU). CASTOR hace uso de este gestor de trabajos para organizar las peticiones, tanto de escritura como de lectura de ficheros. Al poder hacer uso de todas las facilidades ofrecidas por LSF es más fácil implementar políticas de prioridades, uso de buffers, etc.

dCache

fue diseñado como un front-end para un conjunto de *Hierarchical Storage Managers* (HSM). El propósito de un sistema HSM es organizar sistemas de almacenamiento de back-end o jerarquizados en niveles de

[17] El término inglés thread, que se puede traducir por hilo (o hilo de ejecución), es una unidad básica de ejecución en un sistema operativo. Un thread se puede ver como la unión de un trozo de programa junto con una serie de recursos (espacio de memoria, los archivos abiertos, situación de autenticación, etc.) La unión de varios threads que comparten los mismos recursos forman un proceso. El uso de threads permite a una aplicación realizar varias tareas concurrentemente.

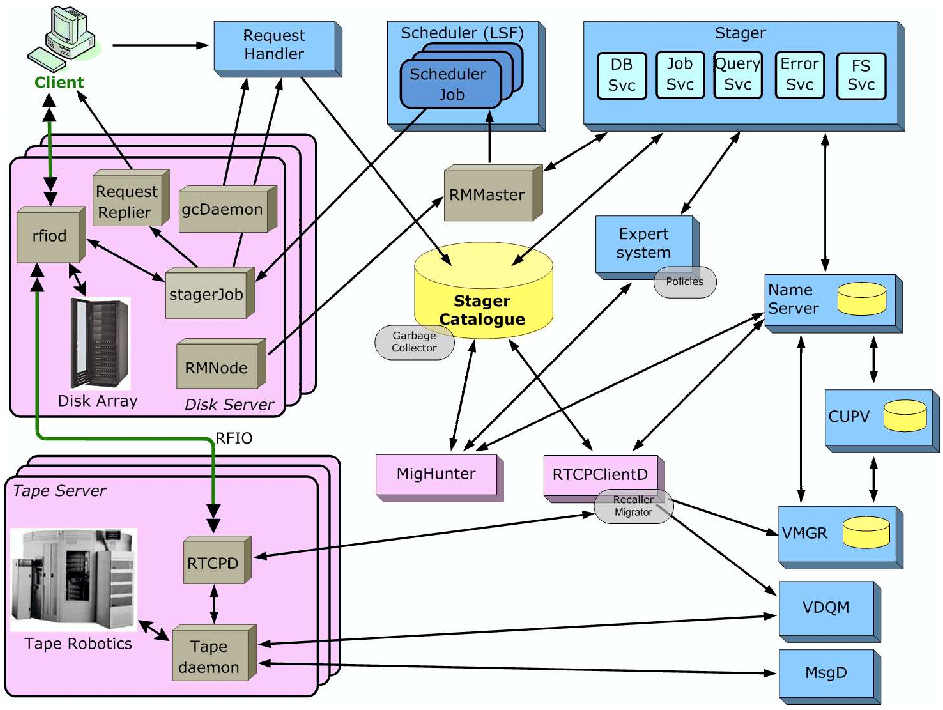


Figura 2.5: Arquitectura de CASTOR.

discos y cintas. El objetivo con el que se desarrolló dCache fue optimizar el uso de los dispositivos de cinta existentes y la integración de tecnologías de disco, sin que ello suponga un coste en el rendimiento, mediante la introducción de cachés en disco como front-end. El requisito básico es poder dar soporte a varias cachés descentralizadas y métodos de acceso a los datos independientemente de su localización. Este sistema es ampliamente usado como un buffer de disco que actúa como front-end para varios sistemas de almacenamiento masivo. Sin embargo, al igual que con CASTOR, puede configurarse para gestionar solamente almacenamiento en disco (sin usar cintas).

Las componentes principales son un servidor y uno o varios nodos de almacenamiento. El servidor aloja los servicios de administración, los puertos dcap, gsidcap y GridFTP, el servicio SRM, y la gestión del espacio de nombres. Así, el servidor representa un punto único de acceso al SE y presenta los ficheros guardados en el sistema de almacenamiento bajo un espacio de nombres virtual. Este espacio de nombres está gestionado por pnfs [41]. dCache usa pnfs como sistema de ficheros y para guardar los metadata, pues pnfs implementa un servidor de NFS de tal forma que se puede acceder a los ficheros y/o metadata mediante cualquier cliente NFS. En pnfs, los directorios tienen un cierto número de etiquetas (o *tags*). dCache utiliza estos tags de directorio para controlar qué pools son usados para el almacenamiento de los ficheros de ese directorio. La puerta de entrada a un servicio dCache se conoce como *door*. Ejemplos de doors son el SRM door, el gsi/dcap door, GridFTP door. Estos doors son los encargados de gestionar todos los procesos en los servicios de datos.

Aparte del servidor, la otra componente del sistema son los nodos de almacenamiento. Estos nodos se pueden añadir dinámicamente. Cada uno de estos nodos se puede dividir en varios pools, de forma totalmente configurable. Esta fragmentación permite, por ejemplo, asociar diferentes pools a distintas VOs, o a determinados grupos de usuarios. Los pools pueden ser de sólo escritura, sólo lectura, o para lectura/escritura en lo que se refiere a las operaciones permitidas; y de tipo volátil o permanente en función de si los ficheros pueden ser o no borrados por el módulo de limpieza. Los ficheros pueden ser

transferidos entre los distintos pools, de forma automática, en respuesta a una serie de condiciones:

- Si un fichero es solicitado por un cliente pero se encuentra alojado en un pool en el que, por configuración, el cliente no tiene permiso de lectura, el Datasets es copiado primero a otro pool donde el cliente sí tiene permitido el acceso para lectura.

- Si un pool se encuentra muy cargado, el sistema podría, si está configurado para ello, hacer réplicas de los ficheros a otros pools para lograr una mejor distribución de la carga de trabajo.

- Las operaciones de recuperación de los ficheros pueden dividirse en dos pasos. El primero lee los datos de cinta a un pool HSM conectado mientras que el segundo paso se encarga de replicar el fichero a un pool general de lectura.

- Si un Dataset se escribe en dCache, podría ser necesario tener estos ficheros replicados de forma instantánea. Los motivos pueden ser, o bien disponer de una copia de seguridad, o bien para asegurar que los clientes no intentarán acceder a él para su lectura a través de pools de sólo escritura.

El módulo fundamental de dCache es el *Pool Manager*. Cuando un usuario ejecuta una acción sobre un fichero (lectura o escritura) se envía al sistema una petición de transferencia (o *transfer request*). Entonces, el Pool Manager decide cómo gestionar esta petición. Por ejemplo, si un fichero solicitado para lectura está replicado en varios pools, se devolverá la copia de aquel que esté menos cargado. Pueden también ejecutarse algunas de las operaciones de copia entre pools comentadas, si el Pool Manager lo considera necesario. El comportamiento del Pool Manager es altamente configurable. Su componente principal es conocida como *Pool Selection Unit* (PSU), encargada de encontrar los pools que el Pool Manager tiene permitido usar para gestionar las peticiones de transferencia que le llegan. Mediante la configuración de qué pools pueden usarse para cada tipo de petición, el sistema puede configurarse para distintos escenarios: distintos pools para distintas organizaciones, pools especiales optimizados para escritura, etc. Pool Selection Unit genera la lista de pools disponibles para cada tipo de peticiones consultando una serie de reglas, también llamadas *links*, compuestas por condiciones y listas de pools. Aquellos pools que cumplen todas las condiciones se añaden a la lista de pools disponibles para ese tipo de peticiones.

Los procesos que se encargan de realizar la transferencia de datos en un servidor de disco se conocen como *movers*, y se crean a medida que llegan estas peticiones de transferencia. Se puede configurar el número máximo por pool y por protocolo (dcap, GridFTP, etc.) Estos números máximos son los que permiten determinar la carga de los pools y, en consecuencia, decidir si hacer o no réplicas de los ficheros, como se ha comentado anteriormente.

La figura 21 muestra un esquema de la arquitectura de dCache.

***Disk Pool Manager* (DPM)**

es un gestor de discos ligero, apropiado para sitios relativamente pequeños. Originalmente se pensó como una alternativa de almacenamiento sin cinta, apropiada para aquellos centros que no pueden permitirse gestionar un sistema más complejo como, por ejemplo, dCache. Los discos pueden añadirse dinámicamente en cualquier momento. Al igual que con CASTOR y dCache, la existencia de un espacio de nombre virtual oculta las complejidades de la arquitectura del pool de disco.

Storage Element clásico

no es más que un servidor GridFTP y un servicio de RFIO no seguro para acceder a ficheros guardados en uno o más discos. A diferencia de los casos anteriores, no se gestiona un espacio de nombres virtual, sino que los PFNs son conocidos directamente. Esto tiene el incoveniente de la falta de flexibilidad. El SE clásico ya no se utiliza.

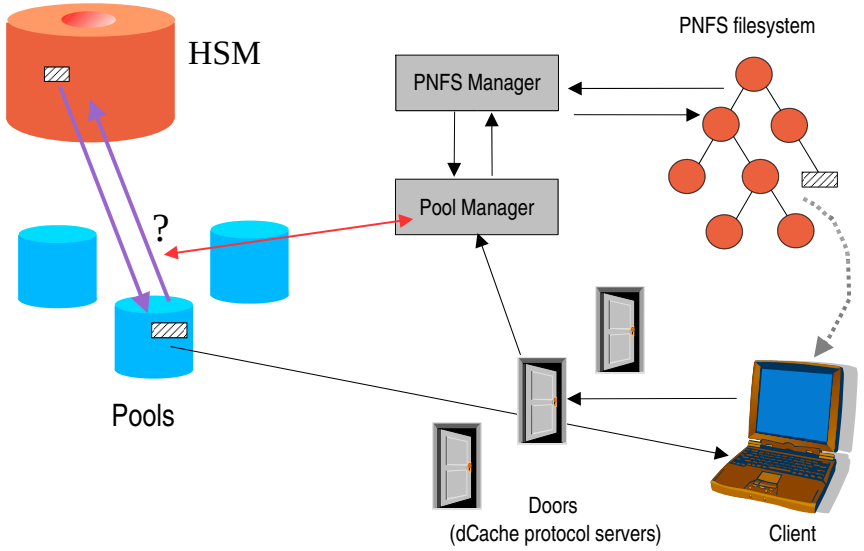


Figura 2.6: Arquitectura de dCache.

2.2.3. Sistemas de Información y Monitorización

Un sistema tan complejo y tan ampliamente distribuido como son los sistemas Grid necesitan de herramientas fiables que permitan monitorizar en todo momento la disponibilidad y calidad de los servicios, de los centros involucrados, y de las actividades que en ellos se están desarrollando. La principal características de los sistemas Grid, incluyendo LCG, es que no se tiene acceso directo a los recursos. Esto dificulta las tareas de monitorización y la pronta reacción antes cualquier eventualidad. Se hacen necesarios sistemas de monitorización que estén distribuidos de forma similar a los recursos y las actividades:

- **Recursos y servicios.** Son los propios centros los que comunican a un sistema central que recoge toda la información la existencia y características de los recursos y servicios que ofrecen. De esta forma se hacen públicos a los usuarios todos los elementos Grid de los que pueden hacer uso. Existen mecanismos que envían constantemente trabajos a los centros para confirmar la veracidad de la información publicada.
- **Utilización del Grid.** Se han desarrollado herramientas que permiten a los propios trabajos que se envían al Grid comunicar cierta información sobre su ejecución. Esta información recogida permite hacer públicas algunas estadísticas de utilidad, generalmente mediante interfaces web.

2.2.3.1. Sistema de Información

El Information Service proporciona información sobre los recursos que hay disponibles en LCG y sobre su estado actual. Esta información es esencial para el funcionamiento de todo el Grid pues el IS permite localizar los CE disponibles para ejecutar trabajos, los SE que guardan réplicas de los ficheros y los catálogos con información sobre estas réplicas.

La información que el IS publica se ajusta al esquema GLUE (*Grid Laboratory for a Uniform Enviroment*) [42], que trata de definir un modelo común para describir las propiedades de los CEs y los SEs. El esquema GLUE proporciona una descripción estandarizada de los sistemas computacionales Grid, con el objetivo de facilitar que los recursos y servicios disponibles se presenten a los usuarios (o servicios externos) de una manera uniforme. Para conseguirlo maneja un conjunto de atributos apropiados para la mayoría de los casos prácticos, incluyendo el descubrimiento (¿qué recursos hay?), selección (¿cuáles son sus propiedades?) y monitorización (¿en qué estado se encuentran?).

Actualmente existen en LCG dos sistemas de información, el *Monitoring and Discovery Service* (MDS), que se usa para el descubrimiento de recursos y para publicar el estado de los mismos, y la *Relational Grid Monitoring Architecture* (R-GMA), usada para accounting, monitorización y publicación de información a nivel de usuario.

MDS implementa el esquema GLUE haciendo uso de *Lightweigh Directory Access Protocol* (LDAP) [43], una base de datos especializada y optimizada para la lectura, inspección y búsqueda de información. El modelo de información LDAP está basado en entradas (servicios, usuarios, PCs, etc.) con uno o más atributos. Estas entradas vienen caracterizadas por un identificador único, el *Distinguished Name* (DN). Estos DNs se pueden agrupar en una estructura jerárquica tipo árbol, conocida como *Directory Information Tree* (DIT). Un ejemplo de esta organización jerarquizada de la información se muestra en la figura 2.7 El esquema LDAP, por tanto, describe la información que puede ser registrada en cada entrada del DIT. En los CEs y SEs se ejecuta un pequeño programa, el *Information Provider*, que genera la información relevante sobre el recurso y la publica a través de un servidor LDAP conocido como *Grid Resource Information Server* (GRIS). Simultáneamente, otro servidor LDAP, el *Site Grid Index Information Server* (GIIS), recopila toda la información de los GRISes locales y la vuelve a publicar haciendo uso del *Berkeley Database Information Index* (BDII) para guardar los datos. Por último, otro servidor BDII, situado en el nivel más alto de esta estructura jerárquica, recoge toda la información procedente de varios centros. Posee, por tanto, una visión global de todo el sistema Grid. La figura 2.8 muestra un ejemplo con esta organización jerárquica, donde dos servidores BDII recopilan la información de cuatro centros.

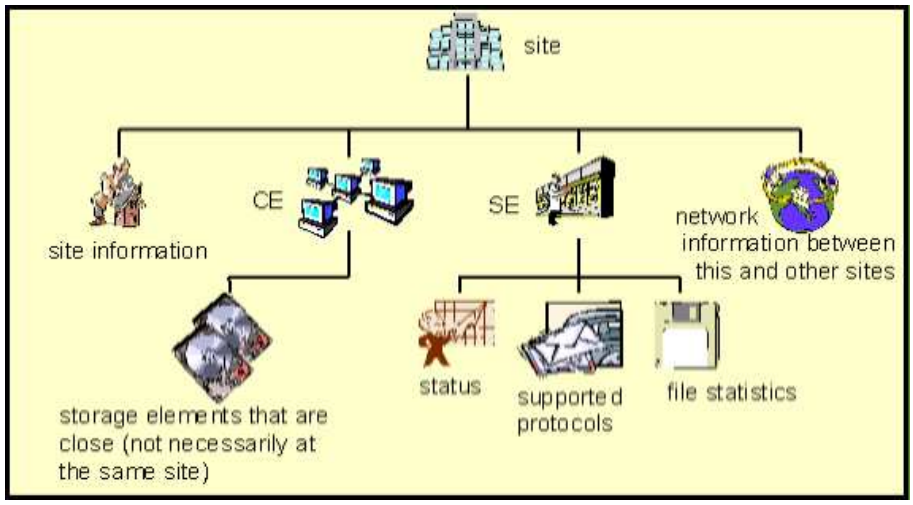


Figura 2.7: Información jerarquizada a través del Directory Information Tree

2.2.3.2. Sistema y herramientas de Monitorización

La capacidad para monitorizar todos los parámetros relacionados con los recursos disponibles es una necesidad funcional en cualquier sistema distribuido. Un sistema de monitorización apropiado implica la existencia de un repositorio central con información operacional. Las herramientas de este tipo recogen datos de todos los recursos que componen el sistema con el objetivo de poder analizar el uso, comportamiento y rendimiento del Grid, detectar y notificar rápidamente cualquier situación de fallo y responder ante posibles riesgos de seguridad.

Basándose en la información proporcionada por el IS, se han desarrollado distintos sistemas y herramientas

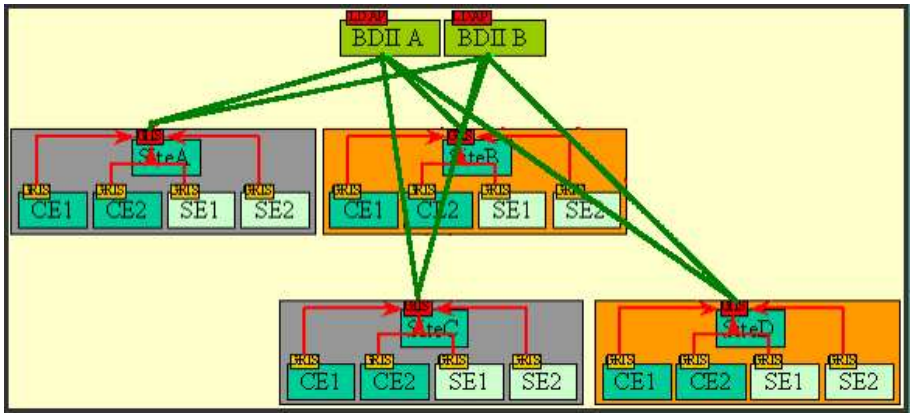


Figura 2.8: Organización jerarquizada del sistema de información

de monitorización y de *bookkeeping*[18]. La más extendida es R-GMA [44]. R-GMA es una implementación de la *Grid Monitoring Architecture* (GMA) propuesta por el *Global Grid Forum* (GGF) [45]. En R-GMA la información se presenta como si estuviera en una base de datos relacional distribuida globalmente, lo que permite operaciones de consulta más avanzadas. La arquitectura R-GMA se basa en tres componentes: los productores (que proporcionan y registran cierta información y describen el tipo y estructura de dicha información), los consumidores (que solicitan esta información) y el registro (que actúa como mediador). Esta relación entre los componentes de la arquitectura R-GMA se puede ver en la figura 2.9. R-GMA se suele usar para accounting y para monitorización, tanto a nivel de usuario como a nivel de sistema, sobre el mismo esquema GLUE que sigue MDS.

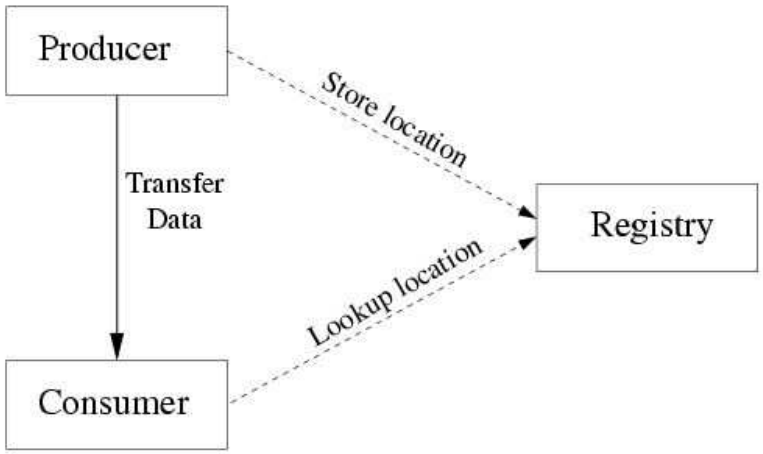


Figura 2.9: Arquitectura R-GMA.

Aparte de R-GMA, se han desarrollado diferentes herrramientas de monitorización, algunas de las cuales son de propósito más general.

GridICE [46] está construido sobre las componentes del Information System. Así, la fuente de información de GridICE es la componente MDS del IS, y el modelo de información para los datos recogidos es una

[18]Se ha generalizado el uso de los términos ingleses *logging and bookkeeping* y *accounting* para hacer referencia a los servicios de registro y de contabilidad. En general, hacen referencia a cualquier mecanismo que recoge, guarda y acumula cualquier tipo de información para su posterior consulta.

extensión del esquema GLUE. Además de recoger periódicamente la información publicada por MDS, GridICE también guarda algunos datos de monitorización históricos de forma persistente. Esto permite el análisis de la evolución temporal de los datos publicados.

SAM (*Service Availability Monitoring*) [47] es un entorno cuyo objetivo es proporcionar una herramienta de monitorización para todos los servicios Grid de forma uniforme, centralizada e independiente del centro. Es la principal fuente de información de monitorización para operaciones Grid alto nivel y actualmente está siendo utilizado para la validación de los centros y servicios a través del cálculo de ciertas métricas de disponibilidad. SAM incluye tests de funcionalidad para monitorizar cada servicio Grid, un módulo encargado de enviar periódicamente estos tests, y un cliente de servicios web para publicar la información recogida. Además, SAM también recoge información procedente del GOCDB (Grid Operations Centre Database) [48], y del BDII para el descubrimiento dinámico de recursos. GOCDB contiene información general sobre los centros que forman parte de LCG, sus recursos de computación, periodos de mantenimiento y forma de contacto con los administradores locales, por ejemplo. Cuando un centro no supera alguno de los tests más críticos puede ser excluido del sistema de información para evitar que se envíen trabajos a centros con problemas. Cada VO puede definir puede definir qué tests se consideran críticos.

Una herramienta que suele operar en conjunción con SAM es GridView [49]. GridView dispone de un módulo de generación de métricas, encargado de computar las métricas de disponibilidad de servicios en los centros basándose en los resultados de los tests de SAM, y de un módulo de visualización que muestra estos resultado en forma gráfica. La figura 2.10 muestra el acoplamiento de SAM y GridView con las distintas fuentes de información.

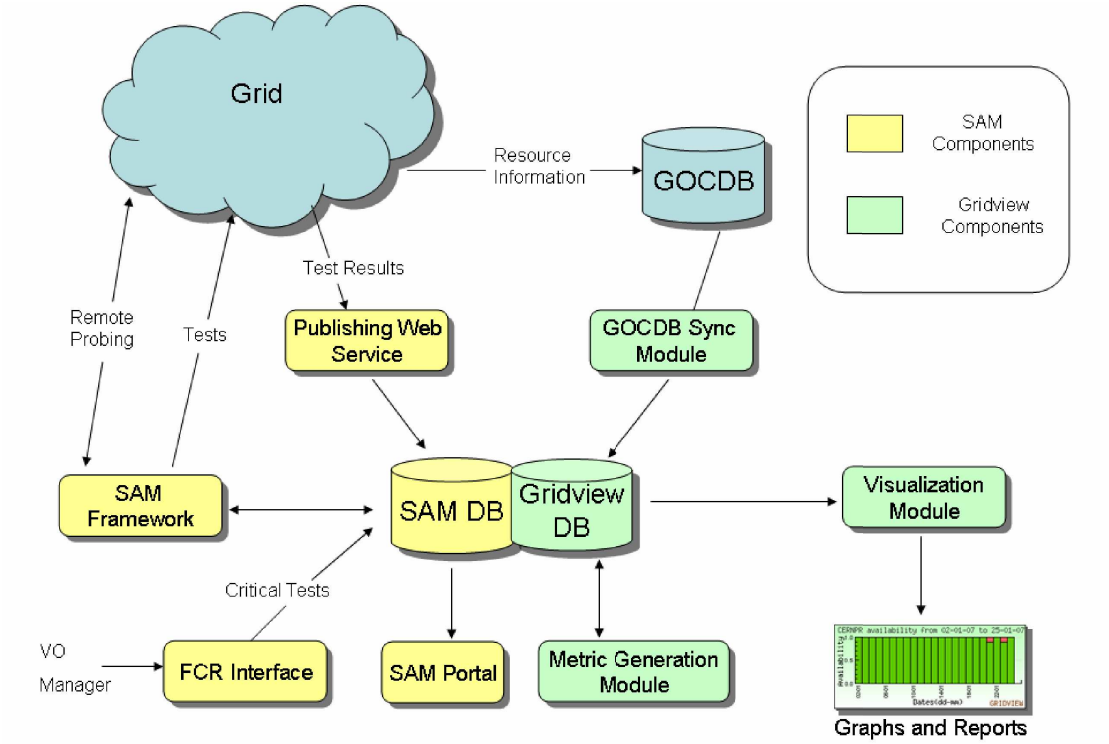


Figura 2.10: Arquitectura de SAM y GridView

Otra herramienta que se usa principalmente para la monitorización de los trabajos ejecutados en el Grid se conoce con el nombre de Dashborad [50]. El objetivo de Dashboard es proporcionar un punto de entrada

único para los datos de monitorización recogidos en los sistemas de computación distribuidos del LCG. Dashboard recoge información de múltiples fuentes, principalmente de R-GMA y MONALISA [51]. Cubre diferentes actividades de los experimentos del LHC, como son el procesamiento de trabajos, gestión de los datos, tests de transferencias, monitorización de la reconstrucción en el Tier-0, monitorización de la eficiencias de los centros, etc. Puede trabajar en los diferentes Grids actualmente implementados (OSG, LCG y gLite).

Toda la información que recoge Dashboard se hace disponible a través de una web, clasificada por temas. Se puede encontrar, por ejemplo, información histórica para distintos periodos de tiempo, estadísticas con las eficiencias de los centros, monitorización de los trabajos de producción Monte Carlo o información sobre las tasas de input/output.

La principal fuente de información de Dashboard es, junto con R-GMA, MONALISA (*MONitoring Agents using a Large Integrated Services Architecture*). MONALISA está basada en una arquitectura dinámica de servicios distribuidos, y es capaz de proporcionar una monitorización completa, control y optimización global de los servicios en sistemas complejos. Ha sido diseñado como un conjunto de subsistemas autónomos, basados en agentes, registrados como servicios dinámicos capaces de colaborar para desempeñar un amplio rango de tareas de procesamiento y de recogida de información. Estos agentes pueden analizar y procesar la información de forma distribuida para ofrecer decisiones de optimización en aplicaciones distribuidas a gran escala. Una arquitectura basada en agentes proporciona la habilidad para investigar el sistema con un grado creciente de inteligencia, reducir la complejidad y hacer sistemas globales manejables en tiempo real. El sistema está diseñado para integrar fácilmente otras herramientas y procedimientos de monitorización ya existentes y proporcionar esta información de forma dinámica a otros servicios o clientes.

El entorno MONALISA es un sistema para el descubrimiento, gestión y monitorización de servicios distribuidos. Entre otras, proporciona las siguientes funcionalidades:

- Descubrimiento, registro y administración remota de servicios y aplicaciones distribuidas.
- Monitorización (incluyendo interfaces gráficas) y bookkeeping de todos los aspectos de sistemas complejos:
  - Información de nodos de computación y *clusters*.
  - Información de red (tráfico, conectividad, topología, etc.)
  - Monitorización del rendimiento de aplicaciones, trabajos y servicios (punto a punto y en el punto final).
- Posibilidad de interaccionar con otros servicios para proporcionar información a medida en tiempo real basada en información monitorizada.
- Agentes interconectados implementados como una red distribuida que supervisan las aplicaciones, capaces de reiniciarlas o reconfigurarlas, y de notificar a otros servicios cuándo ciertas condiciones especificadas son detectadas, y de tomar decisiones de alto nivel.

2.3. Modelo de computación de CMS

La filosofía seguida en el diseño e implemención del modelo de computación de CMS ha sido la de hacer el mayor uso posible de los servicios Grid [52]. Sin embargo, en aquellos casos en los que se considera conveniente, cuando las herramientas disponibles en LCG no ofrecen aún la suficiente fiabilidad, robustez o escala requeridas, se hace uso de herramientas propias desarrolladas en CMS. Así, el modelo de computación hace un uso compartido de los servicios específicos desarrollados para CMS y de los servicios Grid disponibles. Para la gestión de los datos, el modelo de CMS no hace uso del catálogo global de réplicas RLS, sino que está basado en un sistema de catálogos propios. Para las transferencias de

los ficheros se ha desarrollado una aplicación propia que ofrece mejores prestaciones y fiabilidad que las herramientas Grid básicas. Sin embargo, para la gestión de los trabajos sí se hace uso de las facilidades de LCG, gracias a su fiabilidad y robustez, su buen rendimiento, y la existencia de varias herramientas de monitorización.

El primer objetivo es conseguir un sistema inicial que, en el momento en que comience la toma de datos, esté ya operativo y disponga de todas las funcionalidades para implementar los principios básicos del modelo de computación:

- Optimización para los casos comunes, como el acceso de lectura a los datos (la mayoría de los datos se escribirán sólo una vez, pero se leerán varias veces).
- Procesamiento organizado a gran escala (pero sin limitar a los usuarios individuales).
- Desacoplamiento de sus componentes, para permitir cierta independencia entre las partes del sistema.
- La información local de cada centro permanece local.
- Interoperabilidad entre los distintos Grids implementados (LCG y OSG).

La estrategia por la que se ha optado para diseñar el modelo de computación es que los datos sean ubicados en una localización específica. Así, no son los datos lo que se mueven entre centros en respuesta a las peticiones de los trabajos, sino que los trabajos se envían a los centros donde se encuentran los datos. Esta ubicación de los datos se escoge siempre siguiendo las políticas y prioridades de CMS.

2.3.1. Formatos de los datos en CMS

En CMS se usan distintos formatos para guardar la información de los sucesos, dependiendo de factores diversos como el grado de detalle, el tamaño o el refinamiento. Se consigue así un cierto nivel de abstracción, tanto sobre el contenido físico de los datos como sobre la forma en que éstos son agrupados y empaquetados para su manejo. Empezando por los datos tal cual son producidos por el sistema online de adquisición, sucesivos grados de procesamiento refinan estos datos, aplicando calibraciones y creando objetos físicos de más alto nivel. Los distintos formatos que se usan son:

- *Physics Stream*: es la unidad más grande en la organización de datos de CMS. Puede verse como una colección de datos de sucesos procedentes directamente del detector, agrupados por conjuntos de criterios de High Level Trigger que satisfacen, y que pueden ir juntos para satisfacer propósitos de análisis comunes.
- *Primary Dataset*: es la unidad de datos a la que se accede para analizar un sistema de trigger determinado. Contiene, por tanto, los datos correspondientes a todos los niveles de procesamiento pertenecientes a un trigger dado, o a criterios comunes de producción Monte Carlo (es decir, los datos han sido generados con los mismos parámetros). El descubrimiento y entrega de datos se lleva a cabo, principalmente, mediante Primary Datasets. Para los datos reales de CMS se esperan del orden de unos 50 Primary Datasets.
- *Processed Data*: representan una porción de los datos de un Primary Dataset definidos por la historia de procesamiento que se ha aplicado sobre ellos. También pueden corresponder con una gran cantidad de datos simulados producidos con la misma versión del software.
- *Data Tier*: es el resultado de un cierto grado de procesamiento aplicado sobre los datos.
    - *RAW Data*: es el output del sistema online de HLT. Contienen los datos del detector, el resultado del primer nivel de trigger (L1), el resultado de la selección por parte del HLT, y algunos de los objetos de más alto nivel creados durante el procesamiento del HLT. El tamaño típico de cada suceso será de 1.5 MB, aproximadamente.

  - *Reconstructed (RECO) Data*: es el nombre del Data Tier que contiene objetos creados por el software de reconstrucción de sucesos. Se obtienen a partir de los RAW Data y proporcionan acceso a los objetos físicos reconstruidos (trazas, vértices, identificación de partículas, jets...) en un formato adecuado para su análisis físico. Como las tareas de reconstrucción son costosas en términos de recursos computacionales, los Data Tier RECO proporcionan información compacta para su análisis evitando la necesidad de acceder a los RAW Data para la mayoría de los análisis. Cada suceso reconstruido ocupará, aproximadamente, 250 kB de memoria.
  - *Full Event* (FEVT): es la unión de los datos en formato RAW y RECO. Esta unión debe entenderse a nivel lógico y no físico, pues pueden almacenarse en ficheros separados.
  - *Analysis Object Data* (AOD): obtenidos a partir de los RECO, proporcionan suficientes datos sobre los sucesos recogidos, y en un formato adecuado y compacto, para que puedan ser usados directamente en cualquier análisis físico. Contienen una copia de todos los objetos físicos de alto nivel. Incluyen además suficiente información en formato RECO que posibilite algunas tareas de análisis típicas como, por ejemplo, el reajuste de trazas con constantes de alineamiento mejoradas. Este tipo de formato necesita tan sólo unos 50 kB para guardar la información de cada suceso.

- *Event Collection (o Data Collection)*: es el subconjunto de un Processed Dataset más pequeño al que se puede acceder a través del sistema de bookkeeping. Una Event Collection particular contendrá información referente a un Data Tier exclusivamente.
- *Analysis Dataset*: es una subconjunto de Event Collections a partir de un Processed Dataset determinado, como resultado de imponer ciertas restricciones sobre el mismo.
- *File* (fichero): es la unidad básica de almacenamiento. Son los contenedores físicos de los datos, y permiten su manejo por parte de cualquier componente computacional (de almacenamiento, transferencia, cálculo, etc.)
- *File Block*: conjunto de ficheros. Usualmente se agruparán aquellos ficheros a los se accederá simultáneamente para el procesamiento de los datos.

2.3.2. Arquitectura del modelo de computación de CMS

En el modelo de computación implementado por CMS las infraestructuras y servicios de computación están organizados en una estructura jerárquica de niveles (o *Tiers*), donde cada centro tiene asignada una serie de tareas concretas dependiendo del nivel al que pertenece. Ejemplos de estas tareas son la copia de seguridad de los datos tomados por el detector o de los datos simulados, preselección de datos, análisis, simulación de datos, etc. La figura 2.11 muestra un esquema del flujo de los datos, desde el detector hasta el último nivel de la jerarquía.

2.3.2.1. El centro Tier-0 en el CERN

El centro Tier-0 es, por definición, una infraestructura común para CMS, cuyas principales tareas son:

- Recogida de los datos procedentes del detector en formato RAW.
- Primer pase de reconstrucción sobre los RAW data (creación de los RECO data).
- Almacenamiento seguro para los datos, tanto en formato RAW como RECO.
- Distribución de estos datos a los centros Tier-1.

Durante los períodos de toma de datos, el centro Tier-0 aceptará y guardará temporalmente las muestras procedentes del sistema online de adquisición de datos. Esta copia debe realizarse lo antes posible, dada la escasa capacidad de almacenamiento del sistema online, pero garantizando siempre la integridad de los datos. Por otra parte, el enlace entre el sistema online de adquisión de datos y el centro Tier-0 deberá estar dimensionado para mantener el flujo de sucesos, con un margen adicional de seguridad que permita la

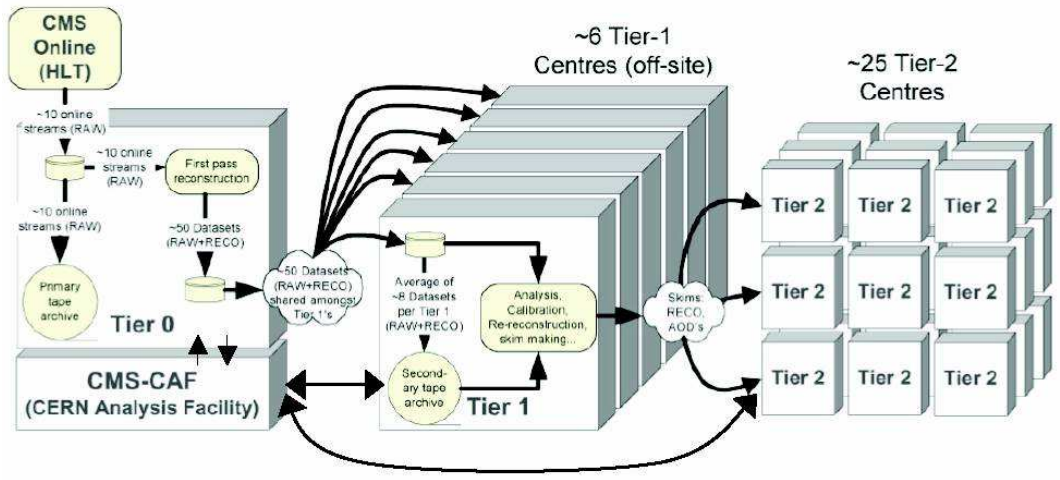


Figura 2.11: Estructura jerarquizada en niveles (Tiers) del modelo de computación de CMS.

rápida recuperación de interrupciones temporales en el flujo de datos. Tras la reconstrucción de los RAW data y la creación de los correspondientes RECO data, estas componentes RAW y RECO son guardadas y distribuidas conjuntamente facilitando así su posterior acceso. Estos datos en formato FEVT son copiados de forma permanente para su seguridad en sistemas de almacenamiento masivo, tanto en el Tier-0 como en centros Tier-1 externos.

CMS hará uso, de forma más o menos continuada, de los recursos del Tier-0 para completar la reconstrucción inicial de todos los datos tomados, prácticamente en tiempo real, incluso en los momentos de máxima luminosidad del LHC. Los resultados de estos pases de reconstrucción son divididos en Datasets físicos y almacenados de forma segura en formato FEVT. Deberá, además, disponer de capacidad de cálculo adicional que permita la pronta recuperación ante cualquier retraso y adecuarse convenientemente ante cualquier contingencia imprevista que pudiese afectar a los tiempos de reconstrucción o a los tamaños de las muestras de datos. Para poder satisfacer estas necesidades se estima que será necesaria una capacidad de cálculo total de, aproximadamente, 4.6 MSI2k (este valor, al igual que los correspondientes a las necesidades de disco y de cinta, tanto para el Tier-0 como para los Tier-1 y Tier-2, corresponden a un año promedio de toma de datos. En general, las necesidades de recursos de computación vienen dadas por la luminosidad acumulada en ese año promedio. Los valores ejemplos que se indican en este apartado corresponden a una luminosidad de $\mathfrak{L} = 2 \text{ x } 10^{33} \text{ cm}^{-2}\text{s}^{-1}$. Las tablas 2.3 y 2.4 muestran los valores requeridos para los primeros años de toma de datos del experimento).

Una de las responsabilidades principales del centro Tier-0 será la de mantener una copia segura de los datos en el CERN. El sistema de almacenamiento masivo aceptará los datos en formato FEVT y proporcionará capacidad de acceso suficiente en todo momento para guardar nuevos ficheros. Se realizarán principalmente operaciones de escritura en los períodos de toma de datos y de lectura de forma intensiva durante las fases en los que el acelerador no esté activo. Para la recogida inmediata de los datos procedentes del detector se estima que serán necesarios unos 0.4 PB de disco, mientras que para el almacenamiento permanente de todas las muestras recogidas se necesitarán aproximadamente unos 4.9 PB. Además, el acceso al sistema de almacenamiento masivo deberá permitir una velocidad promedio de acceso durante las fases de toma de datos de unos 300 MB/s.

El centro Tier-0 será responsable de la distribución de los datos a los centros Tier-1, tanto de los procedentes del sistema online de adquisión como de los resultantes de la primera reconstrucción. Una buena calidad en la conexión de red es, por tanto, fundamental. Los cálculos de los requisitos de conectividad tienen en cuenta la tasa promedio de transferencia de datos, corregida por dos factores adicionales de

seguridad que tienen en cuenta las posibles ineficiencias en los servicios de transferencia de datos o en la propia red. Las necesidades totales se estiman en, aproximadamente, 5 Gb/s.

Finalmente, el Tier-0 deberá ser capaz de implementar las políticas de prioridades de CMS sobre la reconstrucción y distribución de datos, y de aceptar nuevas decisiones de priorización, ofreciendo un corto tiempo de respuesta.

2.3.2.2. Los centros Tier-1

Habrá siete centros Tier-1 para CMS, más otro en el CERN, y sus funcionalidades están relacionadas con el almacenamiento seguro de los datos, tareas de re-reconstrucción, análisis de datos, y son también responsables de servir los datos a los Tier-2 para su análisis, almacenamiento de los datos simulados Monte Carlo, y de ofrecer cierto nivel de soporte a los usuarios, dependiendo de las distintas políticas de prioridades de la colaboración:

- Securizar, y hacer disponibles a los usuarios, una segunda copia de los RAW data y RECO data.
- Recibir y hacer disponible una copia de todos los Datasets en formato AOD completos.
- Participar en las tareas de calibración.
- Ejecutar tareas de filtrado a gran escala sobre los Physics Streams.
- Llevar a cabo pases de reprocesamiento seleccionados para los físicos y los grupos de CMS.
- Distribuir los Datasets a los centros Tier-2.
- Recoger los datos simulados Monte Carlo procedentes de los centros Tier-2, hacer una copia de los mismos, y distribuirlos a otros centros.
- Ejecutar pases de reprocesamiento sobre los Primary Datasets y las muestras Monte Carlo.

Los físicos de CMS podrán llevar a cabo tareas de selección, filtrado y reprocesamiento en los centros Tier-1 sobre los datos previamente guardados en ellos. Estas tareas de procesamiento y selección se llevarán a cabo de manera organizada, ejecutándose sobre las colecciones de datos completas y siguiendo ciertas políticas de prioridades. Esto facilita algunas operaciones básicas como, por ejemplo, los movimientos de los datos entre discos y cintas. Podrán, además, mover los resultados de estas tareas de procesado a otros centros (usualmente Tier-2) para completar su análisis, aunque se espera que determinados tipos de análisis muy específicos (como los relacionados con la calibración o los que requieran mucha estadística) se lleven a cabo enteramente en los Tier-1. Por tanto, será obligación de los Tier-1 proporcionar suficiente capacidad de cálculo para poder realizar todas las tareas de reprocesado y análisis sobre todos los datos que estén almacenados en ellos, en proporción 2 a 1. Se estiman unas necesidades de aproximadamente 2.5 MSI2k por centro, y un ancho de banda medio de un Gigabit/s para que las CPUs puedan acceder rápidamente a los datos.

Otra función crucial de los Tier-1 será proporcionar capacidad de almacenamiento para una fracción significativa de los datos procedentes del detector. CMS mantendrá una segunda copia de seguridad de todos los datos procedentes del detector (aparte de la del Tier-0) distribuida entre todos los centros Tier-1, que se harán responsables de la integridad y administración de estas muestras. Estos datos podrán ser requeridos en cualquier momento para llevar a cabo tareas de reprocesado sobre ellos. Por tanto, los centros Tier-1 deberán garantizar el acceso a todos los datos que tenga almacenados, y potencia de cálculo para poder ejecutar estas tareas de reprocesado. La capacidad de almacenamiento de los centros Tier-1 se distribuye en 1.2 PB de disco, incluyendo ciertos factores de eficiencia, para permitir las tareas de análisis sobre los datos; y 2.8 PB en cinta, para el almacenamiento seguro. El acceso a este almacenamiento masivo se llevará a cabo a una velocidad estimada de 800 MB/s.

Los centros Tier-1 también tendrán la responsabilidad de distribuir a los centros Tier-2 que los soliciten los datos necesarios para ejecutar ciertas tareas de análisis específicas. También los distribuirán a otros centros Tier-1 para su replicación. Aceptarán todos los datos procedentes de los Tier-2, ya sean producto de las simulaciones Monte Carlo o resultado de tareas de análisis específicas que se hayan llevado a cabo en ellos. Cada Tier-1 deberá garantizar, por tanto, suficiente conectividad con el Tier-0, con los demás Tier-1, y con los Tier-2 asociados, para poder dar soporte a todas estas transferencias. La conectividad de red de cada uno de los $N_{T1}-1$ centros Tier-1 (todos excepto el del CERN) deben tener dimensionadas sus conexiones a red para aceptar una parte igual a $\sim 1/(N_{T1}-1)$ de los datos procedentes del Tier-0 y los datos de la producción Monte Carlo provenientes de $\sim N_{T2}/N_{T1}$ de entre los N_{T2} centros Tier-2 (para lo que serán necesarios unos 7.2 Gb/s en total), y exportar las muestras que hayan sido solicitadas a los $\sim N_{T2}/N_{T1}$ centros Tier-2 (para lo que se estiman unas necesidades promedio de 3.5 Gb/s).

Finalmente deberán implementar las políticas de prioridades que la colaboración CMS determine en cada momento para llevar a cabo todas estas tareas.

2.3.2.3. Los centros Tier-2

Los centros Tier-2 serán los responsables de satisfacer los requisitos de análisis de aproximadamente 20-100 físicos cada uno, dependiendo de su tamaño. Proporcionarán, por tanto, recursos de computación para el análisis para una región geográfica o una región física de interés, y dedicarán una fracción significativa de sus recursos de procesamiento a sus comunidades de análisis asociadas, que podrán tener acceso directo a sus recursos. Las principales tareas de los centros Tier-2 son:

- Responsables de satisfacer los requisitos para las tareas de análisis de 20 a 100 físicos.

- Producción Monte Carlo completa.

- Desarrollos de calibraciones y estudios del detector específicos.

Los centros Tier-2 albergarán los pases del análisis sobre los datos filtrados y sobre al menos una fracción de los datos de otros Datasets de CMS que estén guardados en ellos. También se llevarán a cabo en los Tier-2 desarrollos específicos de calibración, alineamiento, u otras operaciones relacionadas con el funcionamiento del detector.

Otra gran responsabilidad de los centros Tier-2 será la producción completa de datos simulados Monte Carlo ($\sim 10^9$ sucesos/año sumando las contribuciones de todos los centros), para lo que deberán disponer de suficiente capacidad de cálculo. Para poder llevar a cabo la simulación del detector completo, el primer pase de reconstrucción y poder satisfacer las necesidades de análisis de su comunidad de físico local, se estiman, en total, unas necesidades de aproximadamente 0.9 MSI2k, y de 200 TB de disco, por centro. Para garantizar el acceso de todos los nodos de computación a esos 200 TB de disco, dichos nodos deberán contar con un capacidad de acceso de 1 Gb/s como mínimo.

Finalmente, los Tier-2 deberán proporcionar capacidad de almacenamiento suficiente para guardar todas las muestras que produzcan, importar los datos procedentes del Tier-1 y algunos conjuntos de datos replicados a otros Tier-2. Para lo primero se estiman unas necesidades de, aproximadamente, un TB diario (unos 10^8 sucesos/Tier-2/año multiplicado por el tamaño promedio de cada suceso, y dividido entre unos 200 días/año de producción, dan como resultado unos 10^6 MB/día). Este es el valor promedio, aunque se podrían alcanzar picos de 8 TB en un sólo día. Para lo segundo serán necesarios, aproximadamente, 5 TB/día (lo que equivale a unos cuantos miles de ficheros a almacenar por día).

En total, las necesidades de red de un Tier-2 promedio para transferir los datos simulados y copiar las muestras que su comunidad de físicos local soliciten se estiman en, aproximadamente, 1 Gb/s.

2.3.2.4. Los centros Tier-3

Los centros Tier-3 son infraestructuras de computación, generalmente de pequeño tamaño, para satisfacer las necesidades de comunidades de usuarios de instituciones locales. Proporcionan recursos y servicios a CMS de forma *oportunista*, y pueden suponer una contribución significativa a las necesidades del experimento. Los centros Tier-3 suponen una componente importante en la capacidad de análisis de CMS y permiten a determinados institutos poder llevar a cabo su trabajo con cierta libertad de acción.

Los centros Tier-3 participarán en las actividades de computación bajo la coordinación de un centro Tier-2 específico, y proporcionarán servicios como el desarrollo de software, análisis interactivos finales y producciones Monte Carlo.

2.3.2.5. La infraestructura de Análisis en el CERN (CMS-CAF)

CMS necesitará ciertos servicios de computación en el CERN aparte de los proporcionados por el Tier- 0. El *CMS CERN Analysis Facility* (CMS-CAF) ofrece una combinación de servicios similares a los proporcionados por los centros Tier-1 y Tier-2.

La función más importante de CMS-CAF será permitir el procesamiento rápido y con baja latencia de ciertos datos críticos, necesario para asegurar el funcionamiento estable y eficiente del detector. Un acceso rápido a los datos guardados en CAF será un factor importante para muchas de estas actividades. Las fundamentales son:

- Diagnóstico de problemas en el detector.
- Activar acciones relacionadas con el rendimiento como reconfiguración, optimización y test de nuevos algoritmos.
- Derivación de los datos de calibración y alineamiento para, por ejemplo, alimentar los algoritmos de HLT o la primera reconstrucción.

Otra función importante de CAF será proporcionar servicios de análisis similares a los que se podrán llevar a cabo en los centros Tier-2. En CAF también podrán llevarse a cabo reprocesamiento de datos y generación de muestras Monte Carlo si fuese necesario.

Los servicios ofrecidos por CAF estarán accesibles para todos los usuarios de CMS, en igualdad de condiciones, incluyendo el acceso interactivo para el desarrollo de código, capacidad para enviar de forma remota trabajos de análisis y disponibilidad para guardar los datos procesados en el espacio de almacenamiento de CAF. Sin embargo, si los usuarios tienen acceso a un centro Tier-2, éste debería ser usado antes que CAF.

2.3.3. Sistema de gestión de datos de CMS

El sistema de gestión de datos es una parte fundamental en el modelo de computación de CMS, que ha sido completamente diseñado e implementado en base a dichos datos. El modelo está concebido para garantizar la recogida, transferencia, almacenamiento, acceso, localización y análisis de los datos producidos por el detector. La arquitectura jerarquizada en Tiers, descrita anteriormente, está pensada para garantizar estas operaciones sobre los datos.

Dada la importancia crucial que la gestión de los datos tiene para el modelo de computación del experimento, CMS optó por desarrollar y utilizar sus propias herramientas cuando las que había disponibles en LCG no ofrecían aún el rendimiento necesario. Estas primeras herramientas de LCG eran ineficientes y poco fiables, no tolerantes a fallos, y la comprobación de que las transferencias se completan con éxito ha de hacerse manualmente. Por este motivo CMS decidió, con el objetivo de paliar estas deficiencias, desarrollar una componente propia para gestionar las transferencias de ficheros: PhEDEx [53]. Una vez que las utilidades de LCG han ganado en fiabilidad y rendimiento se han ido incorporando a la infraestructura de

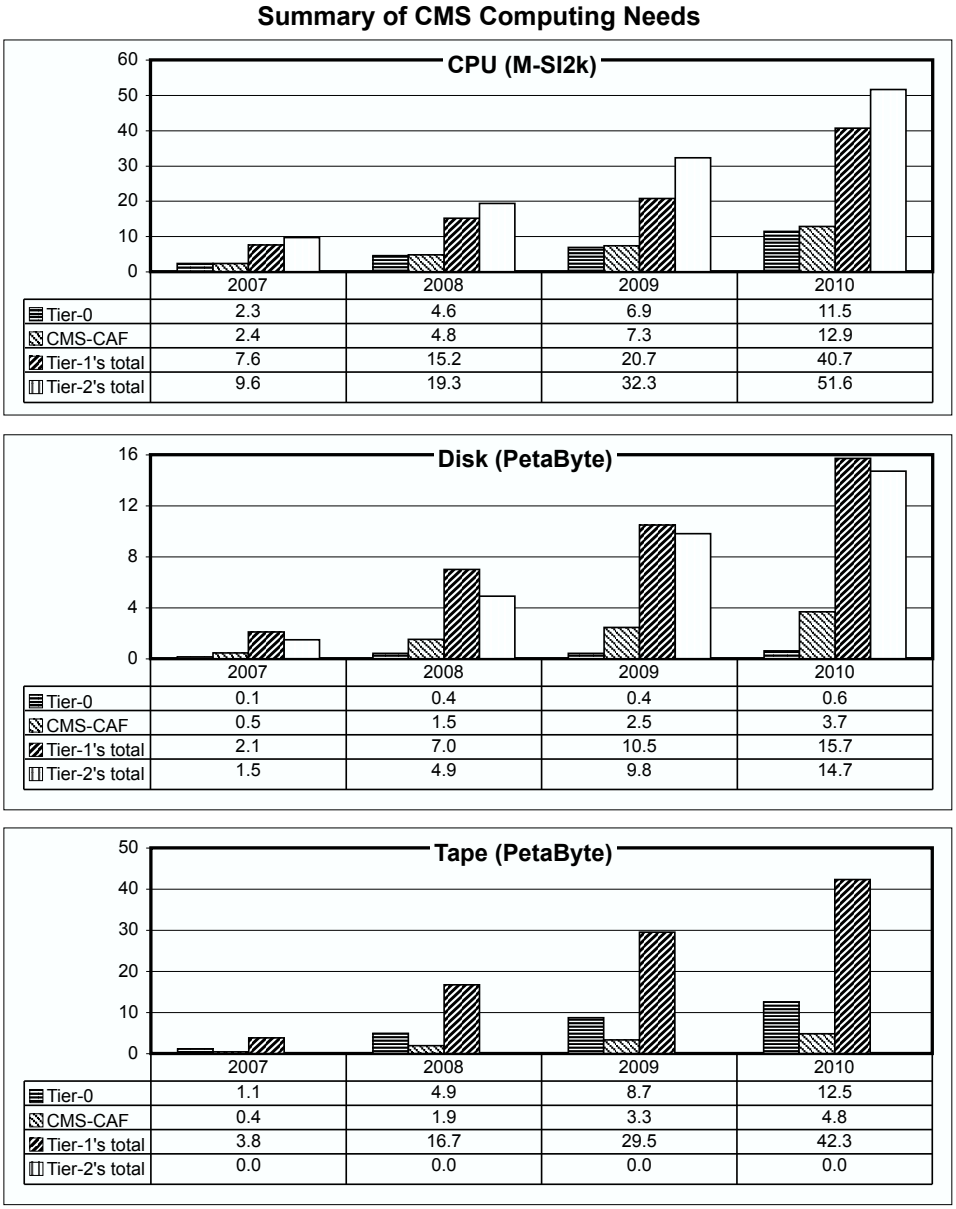
		Running Year				
		2007	2008	2009	2010	
Conditions		Pilot	2E33+HI	2E33+HI	E34+HI	
Tier-0	CPU	2.3	4.6	6.9	11.5	MSi2k
	Disk	0.1	0.4	0.4	0.6	PB
	Tape	1.1	4.9	9	12	PB
	WAN	3	5	8	12	Gb/s
A Tier-1	CPU	1.3	2.5	3.5	6.8	MSi2k
	Disk	0.3	1.2	1.7	2.6	PB
	Tape	0.6	2.8	4.9	7.0	PB
	WAN	3.6	7.2	10.7	16.1	Gb/s
Sum Tier-1	CPU	7.6	15.2	20.7	40.7	MSi2k
	Disk	2.1	7.0	10.5	15.7	PB
	Tape	3.8	16.7	29.5	42.3	PB
A Tier-2	CPU	0.4	0.9	1.4	2.3	MSi2k
	Disk	0.1	0.2	0.4	0.7	PB
	WAN	0.3	0.6	0.8	1.3	Gb/s
Sum Tier-2	CPU	9.6	19.3	32.3	51.6	MSi2k
	Disk	1.5	4.9	9.8	14.7	PB
CMS CERN	CPU	2.4	4.8	7.3	12.9	MSi2k
Analysis Facility	Disk	0.5	1.5	2.5	3.7	PB
(CMS-CAF)	Tape	0.4	1.9	3.3	4.8	PB
	WAN	0.3	5.7	8.5	12.7	Gb/s
Total	CPU	21.9	43.8	67.2	116.6	MSi2k
	Disk	4.1	13.8	23.2	34.7	PB
	Tape	5.4	23.4	41.5	59.5	PB

Tabla 2.3: Evolución temporal de las necesidades de computación de CMS.

computación de CMS, acoplándolas a PhEDEx, como ha sido el caso de FTS y de SRM. PhEDEx sigue teniendo la responsabilidad de implementar las prioridades y las políticas en las transferencias de datos. Además, PhEDEx incluye otras utilidades para tareas de monitorización, contabilidad, y agentes locales que pueden interaccionar directamente con los sistemas de almacenamiento locales, pues SRM aún no incorpora todas las funcionalidades necesarias.

Por otra parte, CMS no hace uso de un registro centralizado de réplicas (RLS) por cuestiones de rendimiento y la fiabilidad. En su lugar, se ha implementado un sistema en el que los ficheros se gestionan por bloques (File Blocks) y no individualmente (DBS/DLS) [54] [55], junto a la utilización de catálogos locales (TFC) [56]. El uso de un catálogo de bloques reduce el número de entradas registradas, y por tanto el número de consultas necesarias para localizar la ubicación de todos los ficheros, en un factor considerable (dependiendo del tamaño de los bloques, usualmente de entre 100 y 1000 ficheros) con la consiguiente mejora en el rendimiento. Además, en los experimentos de Física de Altas Energías es usual analizar y procesar los datos en bloques.

Con el uso combinado de las herramientas Grid y las utilidades propias, CMS persigue mejorar la fiabilidad, escalabilidad y rendimiento en las operaciones de consulta y de transferencia de datos. La figura 2.12 muestra el acoplamiento de estas componentes.



	2007	2008	2009	2010
Tier-0	2.3	4.6	6.9	11.5
CMS-CAF	2.4	4.8	7.3	12.9
Tier-1's total	7.6	15.2	20.7	40.7
Tier-2's total	9.6	19.3	32.3	51.6

	2007	2008	2009	2010
Tier-0	0.1	0.4	0.4	0.6
CMS-CAF	0.5	1.5	2.5	3.7
Tier-1's total	2.1	7.0	10.5	15.7
Tier-2's total	1.5	4.9	9.8	14.7

	2007	2008	2009	2010
Tier-0	1.1	4.9	8.7	12.5
CMS-CAF	0.4	1.9	3.3	4.8
Tier-1's total	3.8	16.7	29.5	42.3
Tier-2's total	0.0	0.0	0.0	0.0

Tabla 2.4: Evolución temporal de las necesidades de computación de CMS.

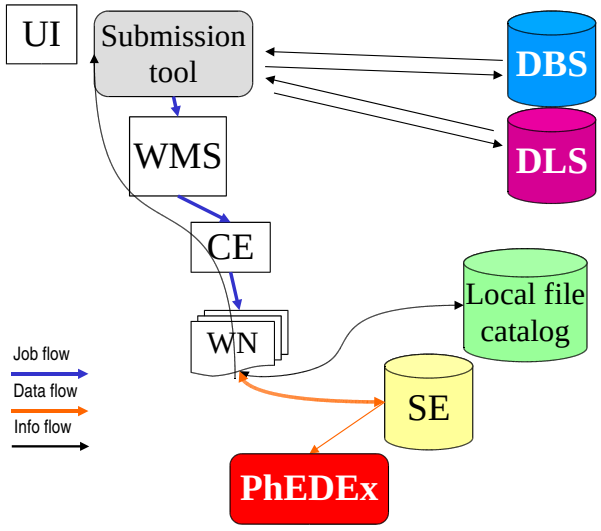


Figura 2.12: Componentes del sistema de gestión de datos de CMS.

2.3.3.1. Servicios de catálogos de datos de CMS

El sistema de registro de datos -*Dataset Bookkeeping System* (DBS)-, es la componente responsable de la descripción de los datos de los sucesos de una forma que sea independiente del centro donde se encuentren alojados. El DBS proporciona los medios necesarios para el descubrimiento, definición y uso de los datos de CMS. Los principales servicios que proporciona el DBS son:

- El descubrimiento de Datasets. El sistema debe ser capaz de dar la lista de Datasets o de colecciones de sucesos a partir de una lista de parámetros de interés que describan los datos.
- La creación de Datasets y su caracterización. Debe ser posible guardar la definición de un Dataset junto con la información que lo caracteriza.
- Facilitar la paralelización. Las herramientas para crear trabajos de producción Monte Carlo, de reprocesamiento de datos o de análisis, pueden usar el DBS para transformar la información disponible de forma que sea posible paralelizar la creación de dichos trabajos.

DBS proporciona una completa descripción de los File Bloks, de los ficheros de cada File Block, y de los sucesos guardados en cada fichero (incluyendo algunos atributos como la luminosidad asociada, la configuración con la que fueron procesados, etc.) Esto permite identificar los bloques de ficheros que agrupan datos un determinado contenido de Física. Para identificar convenientemente los bloques en DBS, éstos se representan como una ruta conocida como *DatasetPath*, compuesta por tres partes: /Primary Dataset/Processed Dataset/Tier.

Desde el punto de vista de la arquitectura el sistema está basado en una jerarquía de dos niveles. Una única instancia DBS, de ámbito global en el CERN (que podría ser replicada para consultas por motivos de rendimiento y accesibilidad), que contiene la descripción de todos los Datasets disponibles para toda la colaboración; y múltiples instancias DBS de ámbito local para propósitos particulares. Generalmente, todos los ficheros con datos serán registrados inicialmente en el DBS local, con la posibilidad de publicarlos también en el DBS global cuando el acceso a dichos datos sea de interés para toda la colaboración. Las instancias DBS locales y del CERN poseen la misma estructura interna y se accede a ellas de igual forma.

El servicio de localización de datos -*Data Location Service* (DLS)-, es la parte del sistema de gestión de datos de CMS encargado de proporcionar un medio para encontrar la localización de las réplicas de los

datos. DLS proporciona solamente los nombres de los SE que guardan los datos, pero no da información sobre la ubicación física de los ficheros dentro del SE.

La localización física de los ficheros dentro de cada centro es conocida únicamente de forma local a través del catálogo de ficheros locales -*Trivial File Catalogue* (TFC)- Se trata de un catálogo trivial con reglas de conversión de LFNs a PFNs (y viceversa), incluyendo el protocolo de acceso, que trabaja sobre un espacio de nombres estructurado dentro del sistema de almacenamiento de datos de cada centro. De esta forma se evita tener que acceder a un catálogo central de réplicas para obtener el PFN de cada fichero. Al usarse el TFC para obtener los PFNs de los ficheros en lugar de usar el catálogo central de réplicas, siguiendo con la filosofía evitar que los trabajos interaccionen demasiado con el sistema de información, no es necesario el uso de herramientas como GFAL.

2.3.3.2. Sistema de transferencia de datos de CMS

La componente que CMS ha desarrollado para la gestión de los datos (ubicación, transferencias de ficheros, etc.) recibe el nombre de *Physics Experiment Data Export* (PhEDEx). Este sistema se usa para definir, ejecutar y monitorizar decisiones administrativas sobre los movimientos de datos, qué copias se considerarán de seguridad, etc. Gestiona la localización de los diversos recursos de almacenamiento disponibles, así como las transferencias de datos entre ellos a nivel de Datasets y File Blocks.

PhEDEx simula los centros donde se guardan los datos como nodos que siguen una topología determinada. Esta topología refleja fielmente la política de CMS respecto a la distribución, almacenamiento y prioridades de los datos. Las transferencias de datos gestionadas por PhEDEx pueden producirse entre nodos no directamente conectados, en cuyo caso los nodos intermedios también participan en la transferencia.

Para gestionar las operaciones de transferencia de datos PhEDEx hace uso de las herramientas disponibles en el Grid, pero incorporando algunas funcionalidades extra para aumentar la fiabilidad y robustez. Algunas de estas funcionalidades son la búsqueda de rutas alternativas, el reintento en caso de fallo, algoritmos de tiempo de espera incremental entre reintentos para evitar la saturación de los sistemas, mecanismos de expiración de transferencias no completadas con éxito en un determinado periodo de tiempo teniendo en cuenta la historia reciente de ese canal, etc.

La arquitectura de PhEDEx está compuesta por una serie de procesos autónomos, conocidos como agentes, capaces de actuar de forma autónoma y flexible para cumplir con sus objetivos de diseño. Estos agentes comparten información sobre el estado de las réplicas y de las transferencias, sobre el enrutamiento a través de la red, las subscripciones a los Datasets y la infraestructura. Entre las componentes de PhEDEx hay bases de datos (que guardan esta información) y agentes que gestionan el movimiento de los ficheros entre centros, la migración de ficheros a sistemas de almacenamiento, asignan ficheros a los distintos destinos basándose en las subscripciones de los centros a los datos, manejan los ficheros localmente, etc.

Haciendo uso de estas componentes, los centros interesados en los datos hacen una petición de un fichero. Un agente exportador en un centro dado hace entonces el fichero disponible para su descarga, genera un TURL y lo inserta en la base de datos central del sistema. El agente remoto copia el fichero. Un agente opcional de limpieza puede borrar el fichero cuando éste ha llegado a su destino final.

2.3.3.3. Acceso a los datos de alineamiento y calibración

Determinados datos relacionados con el funcionamiento del detector y los detalles necesarios para la calibración y alineamiento de todos sus subsistemas han de estar siempre disponibles. Algunas aplicaciones de procesamiento de datos, como el HLT o los programas de reconstrucción y análisis, necesitan acceso a esta información. En el caso del HLT, además, este acceso debe poder hacerse en tiempo real. Principalmente se necesitan dos tipos de datos:

- **Calibración**. Necesarios para entender la respuesta a las señales de input de los canales del detector.

- **Alineamiento**. Es esencial un alineamiento preciso de las componentes del detector involucradas en la reconstrucción de las trayectorias de las partículas. Todos los subsistemas que componen el detector deben estar alienados unos con respecto a los otros.

Dado que el análisis off-line se lleva a cabo de forma altamente distribuida en el Grid y habrá decenas de miles de clientes distribuidos entre los centros donde se procesan los datos que accedan simultáneamente a estos datos, y que es muy probable que la misma información sea solicitada simultáneamente por varias aplicaciones que estén trabajando sobre los datos tomados en el mismo periodo de tiempo, es necesario que el sistema que guarda y distribuye esta información ofrezca un alto grado de accesibilidad.

Una buena solución es usar una base de datos central, y quizás una o dos réplicas de la misma por motivos de redundancia si fuese necesario. Para conseguir un acceso eficiente a esta base de datos central, CMS ha optado por una solución llamada FroNTier [57]. FroNTier es una combinación de sistemas de software y hardware que optimiza el acceso a grandes cantidades de datos registradas en bases de datos. A través de FroNTier, las aplicaciones de los usuarios consiguen acceder a grandes cantidades de información de forma transparente, uniforme, portable, rápida, segura, e independiente de la base de datos. Básicamente, FroNTier es un servicio web que proporciona acceso HTTP a bases de datos centrales.

Por otro lado, dado que la mayoría de estos datos generalmente no cambian, y se pueden considerar de *sólo lectura*, un sistema de *cache*[19] estático puede ayudar a aumentar la eficiencia. Para este fin, FroNTier hace uso de una herramienta llamada Squid [58]. Squid reduce el ancho de banda y los tiempos de repuesta mediante el uso de copias caché y reutilizando las páginas web que son accedidas con mayor frecuencia. El uso de esta herramienta permite que todo el sistema gane en fiabilidad y facilidad de configuración y mantenimiento. La figura 2.13 muestra un esquema con el flujo de datos desde la base de datos central hasta los clientes. En cada etapa, una instancia de Squid hace una copia caché de los datos, y son estas copias las que finalmente leen los clientes.

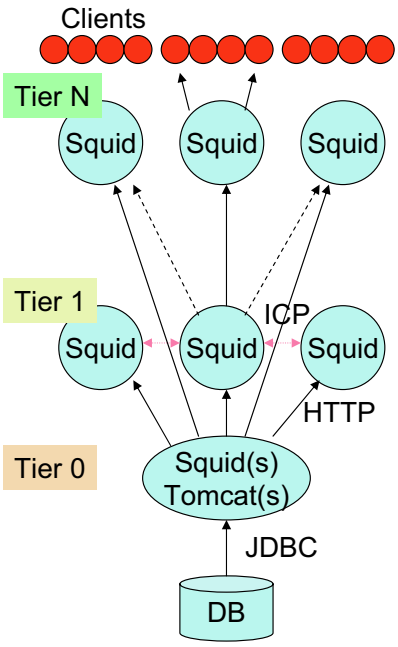


Figura 2.13: Esquema del flujo de información entre la base de datos y los clientes.

[19] conjunto de datos duplicados de otros originales (usualmente costosos de acceder, normalmente en tiempo, respecto a la copia en el caché). Cuando se accede por primera vez a un dato se hace una copia en el caché, y los accesos siguientes se realizan a dicha copia, con lo que el tiempo medio de acceso al dato es menor.

2.3.4. Sistema de gestión de trabajos de CMS

CMS ha desarrollado varias herramientas para automatizar y facilitar la gestión de trabajos de análisis, de producción MC y de reprocesamiento de los datos, haciendo transparente para el usuario el uso de los comandos Grid que ejecutan las operaciones más básicas.

2.3.4.1. Gestión de los trabajos de análisis

Para la gestión de trabajos de análisis, la herramienta que se ha desarrollado en CMS recibe el nombre de *CMS Remote Analysis Builder* (CRAB). Es una aplicación concebida para simplificar el proceso de creación y gestión de trabajos de análisis de CMS en un entorno distribuido Grid. Esta herramienta permite crear los trabajos, enviarlos al Grid, monitorizar su estado, recuperar el output, reenviarlos en caso de fallo o cancelarlos mientras se encuentran en ejecución. La figura 2.14 muestra la relación de CRAB con los demás componentes de LCG para llevar a cabo todas estas tareas.

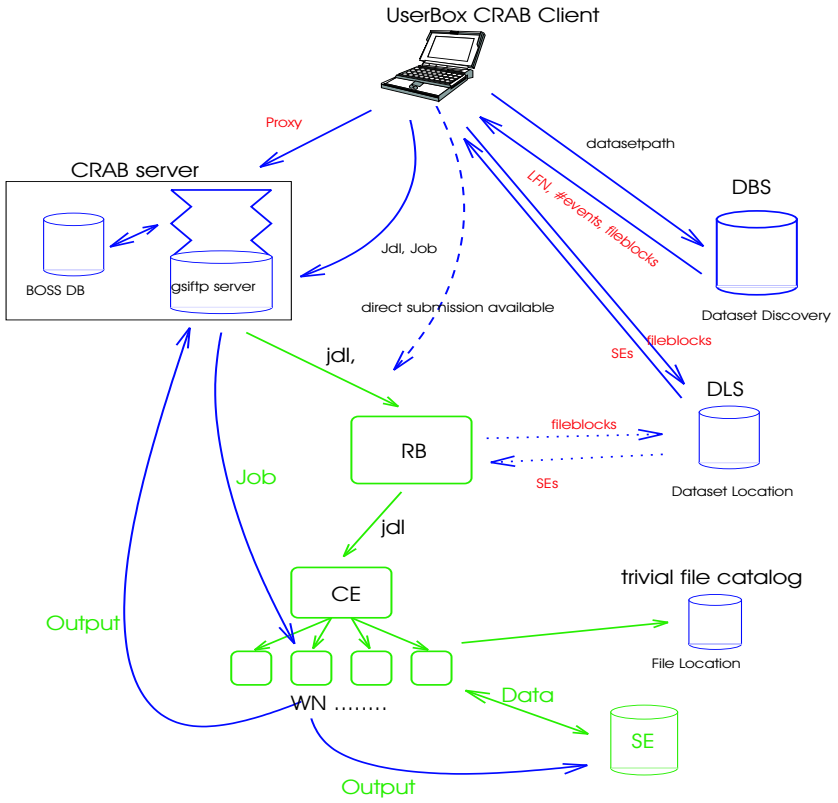


Figura 2.14: Gestión de trabajos en CMS mediante CRAB.

Para cada trabajo individual se crean scripts y algunos ficheros de datos adicionales. Tras su creación, estos scripts son enviados directamente al Grid haciendo uso de un herramienta llamada BOSS (*Batch Object Submission System*) [59]. Esta aplicación actúa como interfaz intermediaria con el Grid Scheduler para el envío de trabajos, como herramienta de monitorización en tiempo real y como sistema de logging y bookkeeping. Las tareas de análisis pueden ser divididas en pequeños trabajos cuando se pretende analizar un gran conjunto de datos siguiendo los requisitos especificados por el usuario.

CRAB da soporte a cualquier programa ejecutable basado en el software oficial del experimento (CMSSW), y con cualquier módulo o librería incluyendo las propias del usuario, y es capaz de manejar el output producido por el ejecutable. Proporciona, además, una interfaz con los servicios de gestión de datos de CMS de forma transparente para el usuario.

Los trabajos son creados de acuerdo a una serie de parámetros especificados por el usuario. Algunos ejemplos son: el tipo de trabajo, el scheduler, la muestra de datos a analizar, el número total de sucesos a procesar y el número de sucesos por cada trabajo, la semilla para la generación de números aleatorios, el nombre de los ficheros de output y los posibles parámetros para su almacenamiento, la configuración de las herramientas de gestión de datos, el RB, parámetros de seguridad, la configuración de las herramientas de monitorización, condiciones sobre el centro donde se ejecutará el trabajo, etc.

2.3.4.2. Gestión de los trabajos de producción Monte Carlo y reprocesamiento de datos

Para la gestión de los trabajos de producción de datos simulados MC y el reprocesado de los datos en el Grid, la herramienta desarrollada por CMS recibe la denominación de ProdAgent. Su objetivo es proporcionar un conjunto de gestores de trabajos automatizados que conformen un sistema de producción coherente a gran escala, capaz de satisfacer los requisitos de producción de datos simulados del experimento. Estas componentes autónomas, cada una de las cuales desempeña una tarea específica, se comunican entre sí a través de una base de datos. ProdAgent ha sido diseñado para conseguir el mayor grado de automatización posible, permitir su desarrollo y mejora de forma sencilla, que sea fácil de mantener, escalable, y que proporcione soporte para múltiples entornos Grid. En la figura 2.15 se pueden ver un esquema del flujo de los trabajos de producción MC gestionados con ProdAgent.

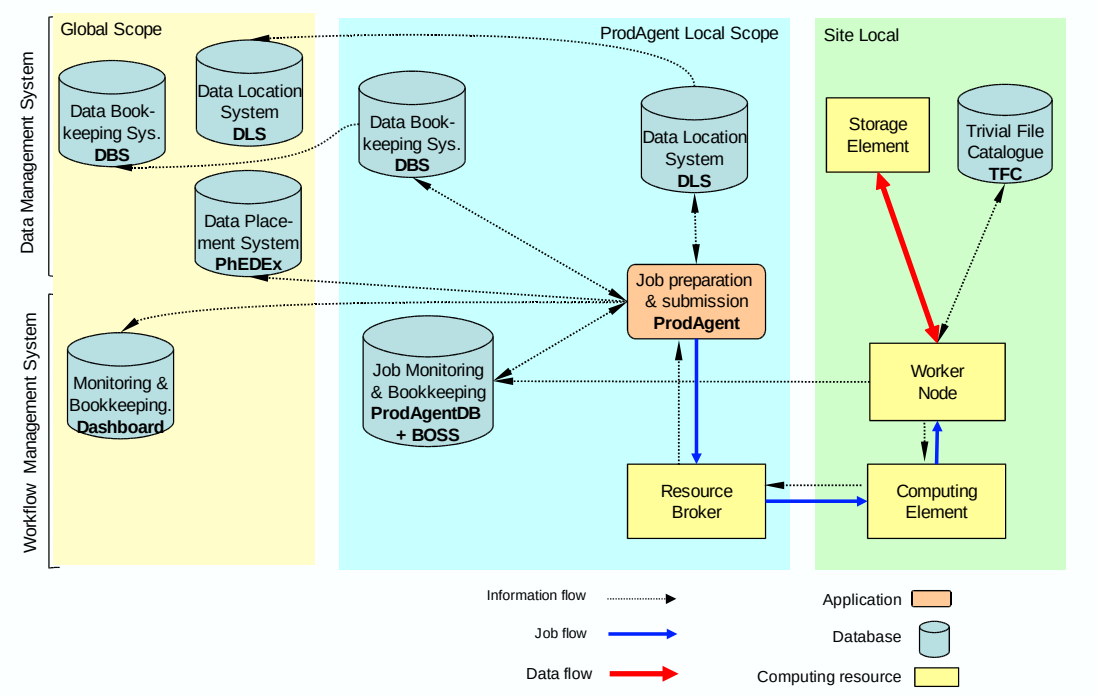


Figura 2.15: Flujo de los trabajos de producción MC gestionados por ProdAgent.

Actúa como un front-end común con diversos recursos, desde granjas de PCs hasta infraestructuras Grid. Divide los trabajos en pequeñas tareas atómicas. Dispone de una API simple para facilitar la comunicación entre sus diversas componentes. Puede funcionar de forma autónoma, aunque también es capaz de colaborar en grandes tareas. Escala mediante la adición de más instancias ProdAgents.
Sus funcionalidades básicas son:

- Envía trabajos para su procesamiento a los centros del Grid.
- Deposita (y junta) los datos producidos en los sitios correspondientes.

- Hace un informe de incidencias.
- Inserta entradas para los nuevos ficheros de datos en los diversos catálogos (incluyendo el del sistema de transferencia de datos de CMS).
- Automatiza el reenvío de trabajos en caso de fallo.

Dadas su versatilidad y escalabilidad, las funcionalidades de ProdAgent se están extendiendo a otras áreas, y no sólo a la producción Monte Carlo. Así, por ejemplo, la componente *CRAB Server* de la figura 2.14 está formada por componentes de ProdAgent. La reconstrucción que se llevará a cabo en el Tier-0 estará gestionada también por una instancia de ProdAgent.

2.3.5. Herramientas de monitorización y bookkeeping

Para las tareas de monitorización y bookkeeping y accounting, tanto de los servicios como de los trabajos y los datos, CMS hace uso de algunas de las herramientas disponibles en LCG y también ha desarrollado e implementado otras propias.

2.3.5.1. Monitorización de los servicios

CMS hace uso de SAM como sistema para comprobar la disponibilidad y calidad de los servicios Grid en los centros. Mediante el envío de trabajos que simulan tareas de análisis y de producción Monte Carlo se verifican algunas funcionalidades básicas (como la disponibilidad para escribir y leer ficheros en el SE, la accesibilidad a software del experimento, la calidad de la información obtenida a través de Trivial File Catalogue, etc.) CMS ha establecido su propia política en base a la cual los centros se mantienen o desaparecen del sistema de información a partir de los resultados de los tests ejecutados con SAM.

2.3.5.2. Monitorización de los trabajos

Los trabajos de análisis y de producción Monte Carlo se monitorizan en CMS haciendo uso de Dashboard, de BOSS, y de un servicio que recoge estadísticas y las hace disponibles a través de una interfaz web.

Los trabajos que se envían al Grid llevan incorporado varios paquetes que les permiten enviar desde el WN donde se ejecutan cierta información útil a Dashboard usando MONALISA como interfaz. Dashboard ha sido adaptado de forma que se puede acceder a algunas estadísticas específicas de los trabajos de análisis y producción de CMS.

BOSS fue desarrollado para proporcionar monitorización y bookkeeping en tiempo real de los trabajos enviados a cualquier granja de computación. En CMS se utiliza como herramienta de monitorización local de los trabajos, tanto de producción Monte Carlo como de análisis. La información que recoge es guardada de forma persistente en una base de datos para su posterior procesado. De esta forma la información disponible es estructurada con un formato bien definido que permite un más fácil acceso. BOSS puede recoger la información proporcionada por el sistema de gestión de colas de trabajos (nombre del trabajo, fecha del envío e inicio de la ejecución, código de retorno, etc.), y también información dinámica proporcionada por el propio trabajo en ejecución (Dataset que se está analizando, número de sucesos procesados, número de sucesos a procesar, etc.) Esta información específica es obtenida a partir del output que el trabajo va generando durante su ejecución. Para conseguirlo, el usuario configura la monitorización suministrando a BOSS el formato de las variables a monitorizar y los scripts necesarios que leen estas variables del output del trabajo. BOSS se encarga de enviar, junto al trabajo de análisis, estos scripts que analizan el output y otro programa que se pone en contacto con la base de datos para actualizar la información registrada en tiempo real. CMS hace un uso extensivo de esta herramienta para enviar y monitorizar los trabajos de producción Monte Carlo.

Finalmente, algunas estadísticas acumulados sobre los trabajos de producción Monte Carlo ejecutados por todos los operadores de CMS se pueden consultar a través de una interfaz web [60]. Incluye información

como el número de trabajos ejecutados (con y sin éxito), los tipos de fallos, tiempo de ejecución de los trabajos, sucesos procesados, etc.

2.3.5.3. Monitorización de los datos

Para la monitorización del sistema de gestión de datos, CMS hace uso principalmente del Data Bookkeeping System (DBS) y de las funcionalidades incorporadas en PhEDEx. DBS, al ser la componente responsable de la definición de los datos y, principalmente, del descubrimiento de los Datasets existentes, también puede considerarse como una herramienta de bookkeeping de los datos de CMS. Respecto a PhEDEx, una de sus características principales es la incorporación de algunas funcionalidades de monitorización y bookkeeping y accounting, y la posibilidad de obtener esta información a través de una interfaz web. Incluye información sobre la actividad reciente (con gráficos sobre el estado y la calidad de las transferencias, tipos de errores, etc.), información sobre los datos (qué réplicas hay, cuáles han de ser borradas, etc.), detalles sobre las peticiones pendientes, una interfaz para administrar las subscripciones a los datos, monitorización de los agentes y de la actividad reciente de las componentes activas, y detallados informes diarios (incluyendo algunas estadísticas como la distribución de los tamaños de los ficheros transferidos, etc.)

Capítulo 3

Incorporación de la infraestructura Grid en los centros españoles

Los centros españoles deben incorporar la infraestructura Grid, tanto de recursos de hardware como de servicios, necesaria para cumplir con estos objetivos, y poder llevar a cabo las tareas que el modelo de computación de CMS asigna a los centros Tier-1 y Tier-2. Las figuras 2.3 y 2.4 muestran los recursos requeridos durante los primeros años para los distintos tipos de centros de computación de CMS. Los centros españoles se distribuyen entre un Tier-1, ubicado en el Puerto de Información Científica (PIC) [61] en Barcelona, y un Tier-2 federado compuesto por el Centro de Investigaciones Energéticas, Medio Ambientales y Tecnológicas (CIEMAT) [62] en Madrid y el Instituto de Física de Cantabria (IFCA) [63]. Estos dos últimos aportan, cada uno, el 50 % de los recursos del Tier-2 completo. En todos ellos se ha hecho un gran esfuerzo para instalar, configurar, mantener y operar toda la infraestructura Grid incorporada, y para dotar a los centros de los recursos materiales y humanos necesarios. Este esfuerzo es continuo para poder cumplir con los objetivos adquiridos. Estos objetivos están marcados, tanto para el Tier-1 como para el Tier-2, en un 5 % respecto al total de recursos del conjunto de centros de cada tipo. En la tabla 1.3 se mostraron las previsiones para los próximos años de los recursos de computación necesarios para cumplir con los objetivos de física de LHC.

La secuenciación del trabajo en todas las actividades, aunque sean continuadas, viene modulada por los ciclos anuales de toma de datos del experimento. Según la planificación del LHC, los recursos de computación previstos para cada año deben estar disponibles a la colaboración en el mes de abril, y operativos durante la toma de datos y posterior análisis el resto del año. La tarea de producción Monte Carlo, aunque puede tener picos de actividad coincidiendo con el análisis, también es típicamente continua. En las tareas relacionadas con el análisis y calibración de datos se pretende que no sólo los recursos estén disponibles al comenzar la toma de datos, sino que también lo esté el entorno de análisis con todos los recursos integrados.

3.1. Infraestructura y servicios del centro Tier-1 español

Para satisfacer los requisitos y necesidades especificados en el modelo de computación, los centros Tier-1 han de incrementar progresivamente sus recursos de cálculo y almacenamiento. Las tablas 3.1 y 3.2 muestran las previsiones de crecimiento del Tier-1 del PIC para los próximos años, globales y para el caso particular de CMS, tanto en potencia de cálculo como en capacidad de almacenamiento. Estos mismos valores se muestran, de forma comparativa, en la figura 3.1. En la figura 3.2 se comparan las previsiones de crecimiento, en potencia de cálculo y capacidad de almacenamiento, de todos los centros Tier-1 de CMS y el Tier-1 del PIC.

	2007	2008	2009	2010
CPU (kSI2k)	501	1654	2647	5381
Disco (TB)	218	845	1578	2878
Cinta (TB)	243	1149	2425	4473
Red (GB/s)	1	10	10	10

Tabla 3.1: Planificación de los recursos de computación del Tier-1 del PIC para los próximos años.

	2007	2008	2009	2010
CPU (kSI2k)	294	692	1161	2868
Disco (TB)	81	313	584	1065
Cinta (TB)	142	731	1614	2845
Red (GB/s)	1	10	10	10

Tabla 3.2: Planificación de los recursos de computación del Tier-1 del PIC, dedicados a CMS, para los próximos años.

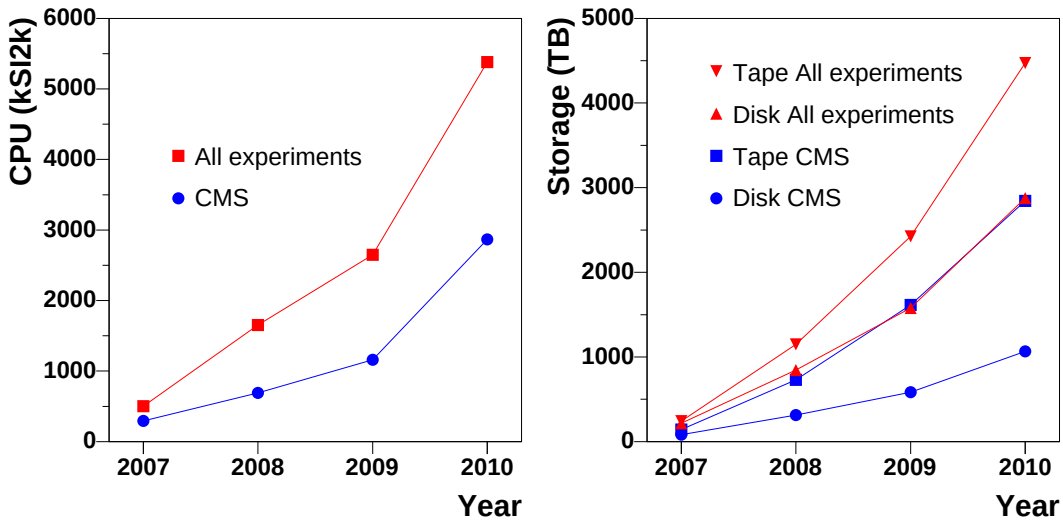


Figura 3.1: Requerimientos de potencia de cálculo y capacidad de almacenamiento para el Tier-1 español.

3.1.1. El Puerto de Información Científica

El Tier-1 de España se encuentra ubicado en el PIC (Barcelona). PIC es un centro fundado en junio del 2003 y financiado por un convenio de colaboración entre cuatro instituciones: Departamento de Educación y Universidades de la Generalidad de Cataluña (DEiU), CIEMAT, Universidad Autónoma de Barcelona (UAB) [64] y el Instituto de Física de Altas Energías (IFAE) [65].

El objetivo principal del PIC es la explotación de la metodología y la tecnología Grid para dotar a las comunidades científicas españolas de los recursos de procesamiento y almacenamiento de datos necesarios para una investigación de élite. La potencialidad de los centros con recursos Grid, como el PIC, radica en el establecimiento de colaboraciones entre instituciones distintas para el uso compartido de datos científicos de forma segura y que optimice el rendimiento de los recursos informáticos de todas las instituciones implicadas. El PIC permite adaptar los avances en el mundo de la investigación y tecnología informática sobre Grid al procesamiento, a precios asequibles, de gran cantidad de datos, con la intención de promover su uso inmediato en aquellas investigaciones de otras disciplinas que puedan beneficiarse de ellos.

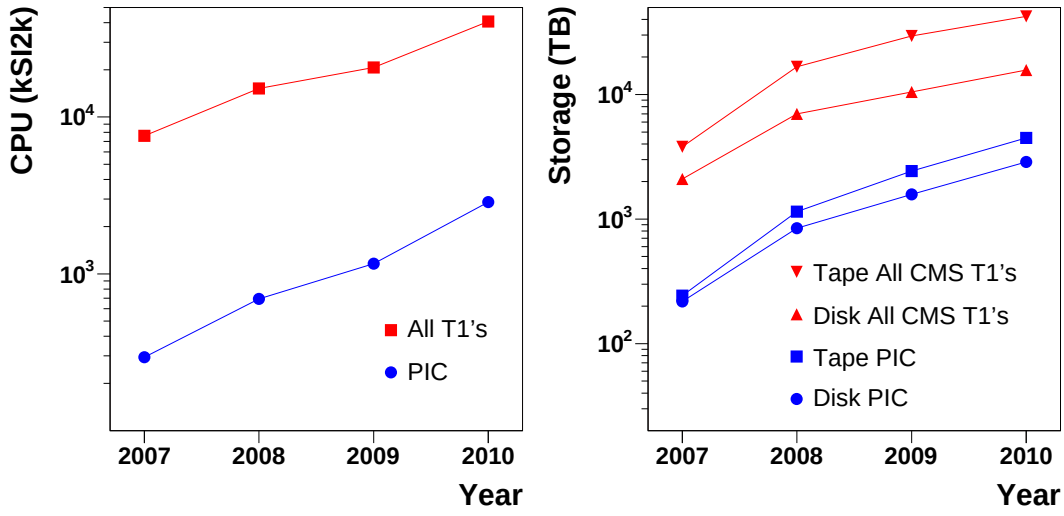


Figura 3.2: Previsiones de potencia de cálculo y capacidad de almacenamiento para todos los T1 de CMS y el PIC.

Los objetivos del PIC son:

- Servir como centro coordinador de computación científica en un entorno masivo de datos, inicialmente los de Física de Altas Energías (en especial los producidos por el LHC), para usuarios españoles, y abierto a otros usuarios potenciales de la Unión Europea y de terceros países.
- Servir como centro coordinador en la computación para el LHC de los grupos españoles de Física de Altas Energías, y de éstos con la comunidad internacional de computación para el LHC, particularmente con el CERN.
- Servir como centro de referencia en técnicas de computación científica en un entorno masivo de datos y extenderlas a otras disciplinas científicas que se puedan beneficiar de las mismas.
- Establecer colaboraciones con instituciones españolas relacionadas con la computación científica, en particular en supercomputación y computación masiva paralela.
- Desarrollar técnicas y servicios de interés general en el contexto de la futura Grid de información, sirviendo como centro de referencia.
- Desarrollar un centro de excelencia que permita a España participar en proyectos europeos dirigidos hacia el desarrollo de una futura infraestructura Grid internacional.

Los proyectos en los que el PIC colabora son EGEE, LCG, Magic [66], K2K [67], y un proyecto de almacenamiento de imágenes médicas. En el caso de LCG, el PIC está involucrado en tres experimentos (ATLAS, CMS y LHCb). Para distribuir la cantidad total de recursos disponibles entre estos tres experimentos, el criterio escogido ha sido dividirlos de forma proporcional al coste de la construcción de cada uno de ellos. La tabla 3.3 muestra la participación del Tier-1 del PIC en cada experimento y la proporción relativa de cada una de esas contribuciones. En el caso de CMS, el PIC aporta un 5 % de los recursos de todos los centros Tier-1. Los recursos de CPU, disco y cinta se derivan de los cocientes cpu/disco y disco/cinta para cada experimento.

3.1.2. Infraestructura hardware en el PIC

La granja de computación del PIC está formada por un conjunto de nodos, todos de iguales características, de Intel [68]. Esta homogeneidad en los recursos facilita la identificación y corrección de posibles

Experimento	Participación	Contribución relativa
ATLAS	5.0 %	53 %
CMS	5.0 %	37 %
LHCb	6.5 %	10 %

Tabla 3.3: Proporción de recursos de almacenamiento dedicados a cada uno de los experimentos del LHC en los que el PIC está involucrado.

problemas. La tabla 3.4 recoge las principales características técnicas de estos WNs. En total hay 50 nodos, con 4 CPUs cada uno de ellos, que ofrecen en total una potencia de cálculo de unos 600 kSI2k. La mitad de estas máquinas ($\sim$ 100 CPUs) están dedicadas a CMS.

Tipo de nodo	Frecuencia CPU (GHz)	RAM (GB)	Pontencia (kSI2k)	Nodos	CPU/Nodo	Potencia Total (kSI2k)
Intel(R) Xeon 5160	3.0	8	3.0	50	4	600

Tabla 3.4: Características técnicas más relevantes de la granja de computación del PIC.

La capacidad de almacenamiento en disco que posee actualmente el PIC es de aproximadamente 210 TB de disco. Este espacio está gestionado por 2 tipos de servidores, de Dell [69] y SUN [70], cuyas características se pueden ver en la tabla 3.5. Hay disponibles en total unos 200 TB de espacio. De esta capacidad de almacenamiento total están reservados para CMS unos 60 TB, de los que están ocupados 10 TB aproximadamente (ver figura 3.3). En el caso de los servidores SUN se consigue una conectividad de red de 4 Gbit/s mediante el uso simultáneo de 4 tarjetas de red de un 1 Gbit/s cada una. El *throughput*[1] de cada disco gestionado por estos servidores SUN es del orden de 600 MB/s.

Servidor	CPU	RAM (GB)	Conectividad (Gbit/s)	Espacio gestionado por servidor (TB)	Total (TB)
DELL	Intel(R) Xeon 3.2 GHz	4.0	1.0	2.4	70.0
SUN	AMD x86-64	16.0	4.0	24.0	130.0

Tabla 3.5: Características técnicas más relevantes de los servidores de disco del PIC.

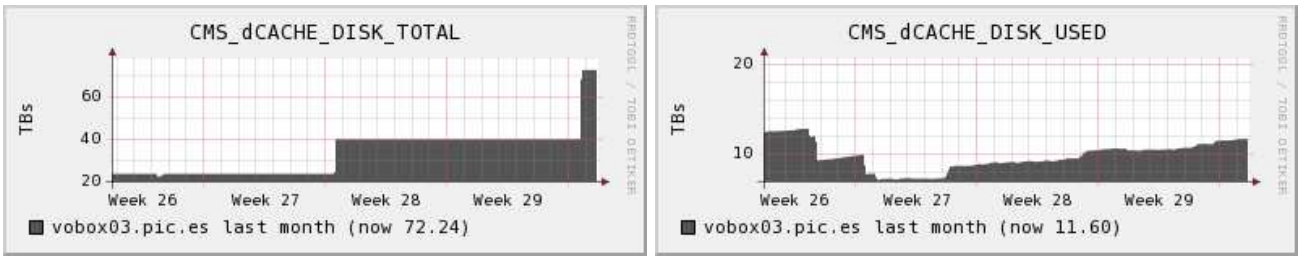


Figura 3.3: Espacio en disco dedicado (izquierda), y usado (derecha), para CMS en el PIC durante 2007.

Para la gestión de los datos en cinta hay desplegados dos sistemas. Uno de ellos es un servicio de SUN, llamado StorageTek [71], con capacidad para 3000 cintas, y la posibilidad de agregar otras 3000 cintas de tipo LTO [72], todas ellas de 200 GB. El segundo es un sistema de IBM configurado inicialmente para aceptar 1500 cintas LTO de 400 GB cada una. La tabla 3.6 muestra, para cada tipo de servidor, el

[1]término con el que se hace referencia a la cantidad de datos por unidad de tiempo entregados por un servicio, físico o lógico, y que pasan a través de algún elemento de red.

número de cintas que hay actualmente instaladas, el número de unidades de lectura y el throughput de cada de una de ellas. Las 207 cintas dedicadas a CMS ofrecen una capacidad de almacenamiento total de unos 41 TB.

Servidor	**Unidades de lectura**	**Throughput (MB/s)**	**Total (MB/s)**	**Cintas instaladas**	**Cintas para CMS**
SUN (StorageTek)	7	30	210	2500	207
IBM	4	80	320	520	0

Tabla 3.6: Características de los servidores de cinta del PIC.

3.1.3. Servicios en el PIC

Se han instalado en el PIC aquellos servicios que le permitan ofrecer la funcionalidad de un centro de sus características y dimensiones. En el modelo de computación de CMS las principales funcionalidades de los centros Tier-1 son la custodia de los datos y el procesamiento organizado (no *caótico*) de los datos, en contraposición al procesamiento caótico de los trabajos de análisis de los usuarios. Este procesamiento organizado facilita el DMS en los centros Tier-1 al ejecutarse los movimientos de datos de cinta a disco de forma organizada.

3.1.3.1. Gestión de trabajos

Para balancear la carga en la gestión de los trabajos que llegan al PIC se han instalado dos CEs, ambos con el software de LCG. Estas dos máquinas tienen un microprocesador Xeon Dual a 3.2 GHz con 4 GB de memoria RAM. Al no haber una comunidad de usuarios locales no ha sido necesario instalar más de una UI para CMS (cuyo único usuario es, por otra parte, el sistema de transferencia de datos PhEDEx).

Los trabajos que llegan a la granja de computación del PIC se organizan en colas gestionadas por TORQUE/MAUI [73]. MAUI es gestor de trabajos para *clusters* y supercomputadores. Es una herramienta configurable y optimizada, capaz de dar soporte a una lista extensa de políticas de gestión y de prioridades dinámicas. MAUI añade a los sistemas de gestión de recursos más básicos políticas y configuración de prioridades de los trabajos, administración de múltiples recursos, control de acceso y políticas configurables de cuotas, gestión del reparto y reserva anticipada de los recursos, diagnósticos del sistema, seguimiento y estadísticas del uso de los recursos, modos de test no intrusivos, etc.

TORQUE es un gestor de recursos que proporciona control sobre colas de trabajos y nodos de computación distribuidos. Es una evolución del gestor de colas PBS, incorporando avances significativos en escalabilidad, tolerancia a fallos, y posibilidades de uso.

Para facilitar la organización de los trabajos las colas suelen estar repartidas por VOs. Además, dentro de cada VO existen determinados usuarios con determinados niveles de prioridad. Estos privilegios se consiguen por dos mecanismos diferentes:

1. Asignar prioridad dentro de las colas a los trabajos del usuario. Este es el caso de los trabajos del *software manager* para la instalación del software del experimento o de los trabajos de producción Monte Carlo enviados por el equipo central de operaciones de CMS.

2. Dedicar recursos a determinados usuarios. De esta forma tienen siempre CPU disponibles que no son utilizadas por ningún otro usuario. En algunos centros (como CIEMAT) esto se hace así para los usuarios de producción MC y los trabajos de monitorización de SAM, por ejemplo.

En el PIC, algunas de las colas son específicas para determinadas VOs, pero también existen otras (*gshort*, *glong*) que son de uso general. Las colas que dan servicio a una única VO tienen las mismas

VO	Porcentaje
lhcb	8 %
atlas	34 %
cms	58 %

Tabla 3.7: Prioridades relativas de las colas asociadas a las VOs de los experimentos del LHC.

prioridades relativas que el centro otorga a los respectivos experimentos asociados. En el caso de LHC, estas prioridades relativas se reparten como muestra la tabla 3.7.
En el caso de la VO de *cms*, existen tres tipos de usuarios: *cms*, *cmsprd* y *cmsgm*. Los usuarios de la cola *cmsgm* tienen la máxima prioridad, mientras que los usuarios de las colas *cms* y *cmsprd* tienen prioridades relativas iguales. Existen pools de usuarios para todas las VOs.

Una vez que un usuario Grid ha accedido al CE a través de una de las colas disponibles, se hace un mapeo para asignarle una identidad UNIX local que será gestionada por TORQUE/MAUI. Este mapeo es distinto dependiendo de si el usuario obtuvo su certificado de autenticación proxy con o sin role. Si es con role, el mapeo lo hace un servicio de VOMS. En caso contrario, se hace mediante las reglas especificadas en el fichero gridmapfile tradicional. A los usuarios de CMS sin role especial se les asigna un usuario local que se escoge entre los que hay disponibles en un pool, con una identidad de la forma *cmsXYZ* (donde XYZ es un cierto valor numérico). Cuando el usuario tiene role de producción ocurre algo parecido, y se le mapea a una cuenta local dentro del pool de cuentas *cmsprodXYZ*.

La figura 3.4 muestra el número total de trabajos gestionados en el PIC durante el último año. Durante el último mes se han gestionado casi 75000 trabajos, y más de 360000 en el último año.

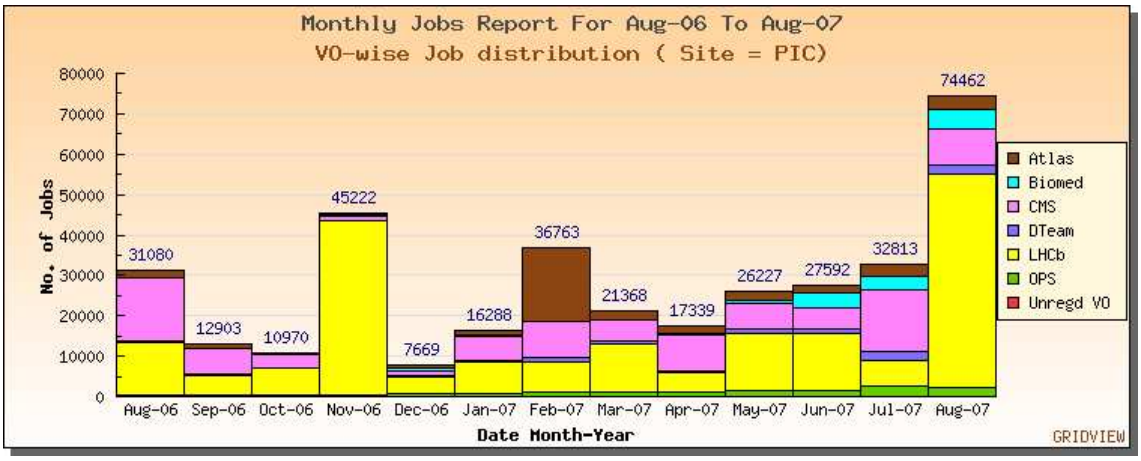


Figura 3.4: Número total de trabajos gestionados en el PIC durante el último año para las distintas VOs.

3.1.3.2. Gestión de datos

Para la gestión del espacio de almacenamiento PIC ha dispuesto hasta ahora de un SE de CASTOR y otro de dCache. En la actualidad se están migrando los servicios de acceso a cinta. Hasta ahora se realizaban a través de CASTOR, pues la instalación de dCache en el PIC no disponía de las funcionalidades necesarias para el almacenamiento en cinta. La solución escogida ha sido usar Enstore [74] en conjunción con dCache. Se trata de un servicio desarrollado en FNAL, ampliamente usado y testeado, y con un buen soporte técnico. Finalmente se procederá al desmantelamiento de CASTOR.

Para el almacenamiento de los datos hay instalado un único pool de disco para todas las operaciones: importación y exportación de datos, y lectura y escritura de datos por parte de los trabajos de procesamiento. Durante el SC3 (ver sección 5.2) el PIC desplegó dos pools de disco distintos, uno para entrada y salida de datos y otro para procesamiento. dados los diferentes patrones y requirimientos de los dos tipos de actividad. Por ejemplo, las transferencias desde el CERN deben estar garantizadas. Otra diferencia es que en las transferencias se leen y escriben ficheros completos a la máxima velocidad, mientras que en el procesamiento de datos la lectura es más lenta. Probablemente se tenderá en el futuro a este tipo de configuración, con dos tipos de pool de disco, o incluso con una segmentación mayor, según la experiencia y las necesidades lo vayan dictando. La migración a cinta se hace de forma automática para aquellos ficheros que son copiados a directorios del espacio de nombres marcados como "migrables a cinta".

Para adecuarse a las especificaciones del modelo de computación de CMS, existen canales específicos en FTS con el Tier-0, con todos los demás Tier-1, y con los centros del Tier-2 asociado. Existen, además, canales que agrupan las transferencias desde los demás Tier-1 al Tier-2 asociado. En el caso de las transferencias desde el CERN, el ancho de banda disponible se reparte entre las VOs de LHC como se muestra en la tabla 3.8.

VO	**Porcentaje**
lhcb	10 %
cms	40 %
atlas	50 %

Tabla 3.8: Distribución del ancho de banda entre el CERN y el PIC en el caso de transferencias simultáneas de varias VOs.

3.1.3.3. Instalación de los servicios

En el caso de los sistemas operativos, la instalación se lleva a cabo mediante un *kickstart* [75]. Kickstart es un método de instalación automático, desarrollado por Red Hat Linux [76], que permite a los administradores crear un fichero con las respuestas a las preguntas que usualmente se plantean durante el proceso de instalación. Este fichero de configuración depende del tipo de nodo que se instala. Los WNs, por ejemplo, necesitan menos software que las UIs.

La instalación del middleware de los servicios se hace con la herramienta oficial de instalación del LCG, llamada Yaim [77]. Es un conjunto de ficheros de configuración y scripts que proporcionan un método rápido para instalar y configurar el middleware de los servicios Grid. También se usa otra herramienta, llamada Quattor [**?**]. Quattor, desarrollado en el CERN, es una herramienta para gestionar la instalación de infraestructuras de computación a gran escala. Aunque estas estructuras no sean componentes Grid en sí mismas, una gestión eficaz de las mismas es imprescindible para conseguir un entorno de trabajo distribuido que sea robusto y eficiente. Quattor permite la instalación, configuración y gestión de clusters de computación de forma correcta, automática y flexible.

3.1.3.4. Herramientas de monitorización

Se han instalado varias herramientas para la monitorización de los recursos y servicios desplegados. Todas estas herramientas son utilidades estándar desarrolladas para los sistemas Linux.

Una de estas herramientas recibe el nombre de Ganglia [78]. Monitoriza una lista configurable de parámetros (como la carga de los procesadores, el números de procesos en ejecución, la ocupación de los discos, etc.) Esta monitorización se hace en tiempo cuasi-real (con una latencia de un 1 minuto, aproximadamente). Una instancia individual en cada nodo recoge información local y la envía a un servidor central que la almacena y la muestra a través de una interfaz web.

Para la monitorización de la red se usa una herramienta llamada Nagios [79]. Nagios permite la monitorización de los servicios de red (como HTTP, SMTP, etc.), de los recursos hardware en las máquinas (uso del disco y de la memoria, carga del procesador, etc.), de algunos factores ambientales (como la temperatura), etc. Permite definir tests a medida, ofrece cierto nivel de escalabilidad, redundancia y flexibilidad. Se pueden consultar los parámetros monitorizados a través de una interfaz web. Una de las características principales de Nagios es que dispone de un sistema de alarmas que avisa al operador cuando alguno de los parámetros supera los límites establecidos. Además, se puede configurar para que, de forma automática, paralice el funcionamiento de los recursos y servicios involucrados en una de estas alertas.

Finalmente, se han desplegado algunas aplicaciones que permiten, a través de respectivas interfaces web, consultar el estado de determinados recursos de hardware o gestionar un sistema de *tickets*. La primera es una aplicación que se ha desarrollado en el PIC. En el segundo caso se trata de un producto externo llamado Issue Tracker [80]. Issue Tracker está diseñado, principalmente, para permitir el seguimiento de errores, gestión de listas de tareas pendientes y para dar soporte a los usuarios.

3.2. Infraestructura y servicios del centro Tier-2 español

Para satisfacer los requisitos y necesidades especificados en el modelo de computación de CMS, los centros Tier-2 también han de incrementar progresivamente sus recursos de cálculo y almacenamiento. Estos requisitos de potencia de cálculo y de almacenamiento se han estimado teniendo en cuenta las necesidades durante el primer período de física del LHC en 2008. Es necesaria una implementación progresiva, como mucho doblando la complejidad cada año, para ir reconociendo y solucionando los problemas que se vayan presentando a medida que aumenta la escala de los centros. Como se recoge en la tabla 2.3, el total de los centros Tier-2 deben tener en 2008 totalmente operativa y a disposición del experimento una potencia de cálculo de casi 20 millones de SpecInt2000, y una capacidad de almacenamiento de aproximadamente 5 PB de disco.

En el caso particular del Tier-2 español, para poder satisfacer los requisitos fijados en un 5 % del total de los centros Tier-2 de CMS, se necesitarán para los próximos años unas capacidades de cálculo y almacenamiento en disco como las que se compilan en la tabla 3.9. Aunque las especificaciones no incluyen almacenamiento robotizado en cinta, se ha considerado importante disponer de suficiente capacidad de este tipo de almacenamiento para salvaguarda de los datos, al menos de los derivados de los programas propios de los grupos de análisis y usuarios que utilicen este Tier-2. La tabla tiene en cuenta este tipo de necesidades. La figura 3.5 compara las previsiones de crecimiento en potencia de cálculo y capacidad de disco para todos los centros Tier-2 de CMS y el centro Tier-2 español.

	2007	2008	2009	2010
CPU (kSI2k)	380	760	1280	2260
Disco (TB)	65	210	420	665
Cinta (TB)	65	210	420	665

Tabla 3.9: Planificación de los recursos de computación del Tier-2 de España para los próximos años.

3.2.1. Infraestructura hardware en el CIEMAT

En el momento actual, el centro del CIEMAT dispone de suficiente potencia de cálculo para satisfacer los requisitos exigibles a un centro de sus características. La tabla 3.10 recoge el número de nodos y de CPUs de cada nodo, y en la tabla 3.11 se detallan sus principales características técnicas: frecuencia, potencia de cálculo y memoria RAM. La granja de computación del CIEMAT ofrece, aproximadamente, unos 120 nodos que agrupan algo más de 400 CPUs, con una potencia de cálculo total de unos 700 kSI2k. Las CPUs escogidas son de Intel y de AMD [81]. En el caso de los nodos Intel(R) Xeon 5160 Dual, 9 de

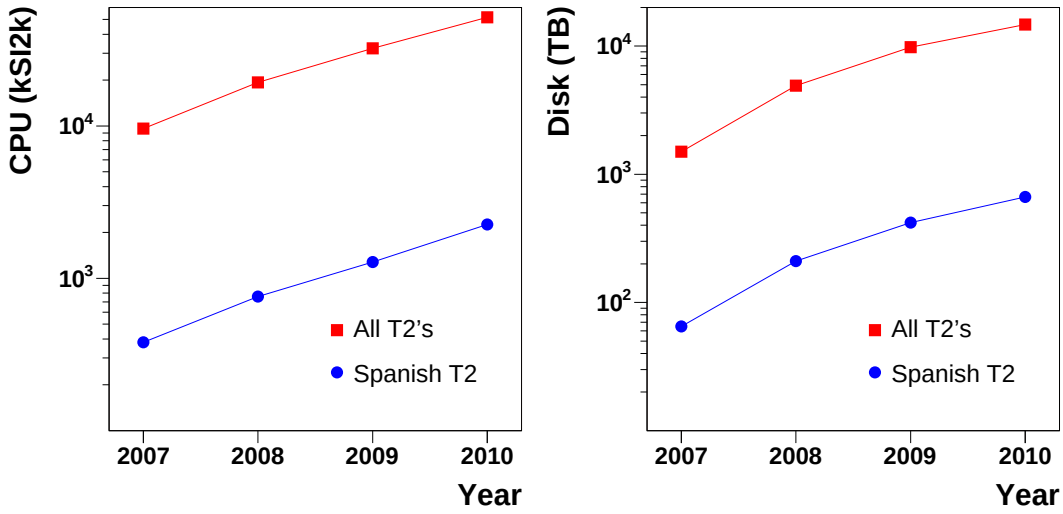


Figura 3.5: Requerimientos de potencia de cálculo y capacidad de disco para todos los centros Tier-2 de CMS y el centro Tier-2 español.

ellos están instalados sobre máquinas virtuales [82] [2]. Algunos de los nodos tienen activado el sistema de *Hyper-Threading* [83], que permite duplicar artificialmente el número de CPUs. Se ha visto, sin embargo, que el uso de esta técnica no se traduce en una ganancia real significativa.

Número de nodos	**CPUs/nodo**	**Total CPUs**
43	2	86
80	4	320
TOTAL		406

Tabla 3.10: Números de nodos de computación y de CPUs en la granja de computación del CIEMAT.

Tipo de nodo	**Frecuencia CPU (GHz)**	**RAM (GB)**	**Pontencia (kSI2k)**	**Nodos**	**Potencia Total (kSI2k)**
AMD Opteron Processor 270	2.0	4	1.3	15	78.0
Intel(R) Xeon (con HyperThreading)	3.2	4	1.3	39	202.8
Intel(R) Xeon	3.2	2	1.3	43	111.8
Intel(R) Xeon 5160 Dual Core	3.0	8	3.0	26	312.0
TOTAL					704.6

Tabla 3.11: Tipos de nodos, y sus características técnicas más relevantes, de la granja de computación del CIEMAT.

[2] La virtualización es la abstracción de los recursos de un computador con el objetivo de ocultar los detalles técnicos de una cierta tecnología a través del ocultamiento de los recursos complejos mediante la creación de una interfaz simple, conocida máquina virtual, o *virtual machine* (VM). Las ventajas que ofrece esta virtualización son:

1. **Duplicación**. Se pueden instalar varias VM en cada máquina física.
2. **Migración**. Si el hardware de una máquina falla se mueve la VM, de forma transparente para los procesos que hay en ejecución, a otra máquina.
3. **Replicación**. Para instalar nuevos nodos que vayan a prestar similares servicios a los ya existentes en el sistema (como pueden ser nodos de computación en una granja) basta con copiar la VM.

Los nodos de la granja tiene todos un disco local del orden de unos 50 GB, donde los trabajos pueden escribir el output que generan (el input lo suelen leer directamente de los servidores de disco), y están conectados mediante tarjetas de red de 1 Gigabit/s de ancho de banda máximo.

La capacidad de disco disponible en la actualidad es la mitad, aproximadamente, de la que corresponde a un Tier-2. El IFCA aporta la otra mitad del espacio en disco. El espacio total disponible está repartido en dos grupos. El primero da servicio a las VOs de *dteam*, *ops* y *cms*; mientras que el segundo está dedicado a las VOs de *dteam*, *ops*, *biomed*, *calice* y *fusion*. Para el primer caso hay disponibles 11 servidores de disco gestionados con CASTOR, con una capacidad total de unos 35 TB. Cada uno de los 11 servidores acumula unos 4 TB de espacio. Una configuración con un número relativamente alto de servidores y poco espacio en cada uno de ellos permite escalar el sistema y suministra un alto rendimiento en cada servidor (tanto de memoria RAM, CPU, y ancho de banda en acceso para lectura/escritura). En su mayor parte contiene datos procedentes de las simulaciones Monte Carlo. Para el segundo grupo se dispone de un SE de DPM con una capacidad algo superior a 2 TB.

En el CIEMAT se dispone de dos tipos servidores de discos, internos y externos, dependiendo de si el servidor está formado por una única caja de discos o por varias, respectivamente. Cada tipo de servidor está controlado por un tipo de tarjeta (o controladora) específico.

Los servidores Dell controlan cajas de discos externas (es decir, el servidor está formado por varias cajas). La capacidad total ofrecida por este servidor ha variado con el tiempo. En la configuración actual consta de 14 discos que suman unos 4.5 TB de espacio. Estos servidores están controlados por tarjetas PERC 4e/DC. Estas tarjetas ofrecen una buena relación calidad/precio.

En el caso de los servidores SUPERMICRO [84], los discos están en la propia caja y se controlan mediante una o dos tarjetas tipo 3-Ware 9xxx [85]. Estas tarjetas ofrecen una excelente relación calidad/precio. Los discos utilizados son del tipo SATA [3] internos, y están agrupados de tal forma que entran 14 discos por caja con una capacidad media de 400 GB por disco (aunque también ha habido discos desde 4 a 8 TB, uno por cada caja en este caso).

En ambos casos, los servidores están gestionados por procesadores Xeon a 3.2 GHz con 2 GB de memoria RAM, y están conectados con tarjetas de red de 1 Gigabit/s de ancho de banda máximo.

3.2.2. Servicios en el CIEMAT

A diferencia del PIC, en el CIEMAT existe una comunidad de físicos asociada al centro y, por tanto, el centro debe estar preparado para permitir la ejecución de los trabajos de análisis. Además, es un centro que colabora muy activamente en las tareas de producción MC. Por todo esto, los servicios relacionados con el WMS juegan un papel dominante. Por otro lado, el centro debe estar preparado para recibir los datos seleccionados a un ritmo de 60 MB/s, preparar el sistema para que funcione como una caché donde se van renovando los datos para el análisis y permitir la escritura de los datos simulados Monte Carlo a 10 MB/s.

3.2.2.1. Gestión de trabajos

Para permitir a los investigadores del CIEMAT el acceso a todos los recursos Grid se han instalado cuatro User Interfaces: una para *cms*, otra para la VO de *fusion*, una específica para los servicios de PhEDEx, y una de propósito general.

Para dar paso a la granja de procesamiento a los trabajos que llegan desde el exterior, hay instalados en el CIEMAT dos Computing Elements (ver tabla 3.12). En uno de ellos se ha instalado el software

[3] Serial ATA, o S-ATA, es una interfaz para discos. Las ventajas de S-ATA son: proporciona mayores velocidades, mejor aprovechamiento cuando hay varios discos, mayor longitud del cable de transmisión de datos, y la capacidad para conectar discos con la computadora encendida.

de LCG-2. Este CE es el que está en operativo y presta servicio a todas las VOs. En el otro CE se ha instalado el software de gLite y actualmente sólo se utiliza para pruebas (y da servicio únicamente, por tanto, a la VO de *dteam*).

Software	**CPUs**	**VOs**	**Función**
LCG-2	336	*dteam,ops,biomed,fusion,calice,cms*	Operaciones
gLite	2	*dteam*	Tests

Tabla 3.12: Computing Elements instalados en el CIEMAT.

Al igual que en el PIC, los trabajos que llegan a la granja de computación se organizan en colas gestionadas por TORQUE/MAUI. Existen otros gestores de colas, como SGE [86] o Condor [87]. Pero suelen ser productos más recientes, por lo que no están tan bien documentados, no hay tanta experiencia en su instalación y mantenimiento, o no ofrecen tantas posibilidades como los productos más antiguos. Además, suelen requerir licencia, y para un centro de las características de un Tier-2 no son necesarios.

No es usual que los trabajos que acceden a la granja de procesamiento lleven consigo todo el software que necesitan, pues existe un límite en el tamaño del Input Sandbox, y por motivos de rendimiento. El software del experimento se pre-instala en los centros y los trabajos acceden a esa copia ya instalada. Una opción es que este software esté ubicado en el disco de cada WN, pero no es una solución eficiente cuando ocupa mucho espacio. La otra opción, implementada en todos los centros, es instalar el software en una o varias máquinas dedicadas, a las que se accede desde los WN a través de NFS (o cualquier otro sistema de ficheros). En el caso de CIEMAT sólo hay un servidor, pues la utilización de varios servidores simultáneamente implica la necesidad de vigilar que todas las copias son iguales, y para un centro de tamaño mediano es suficiente con un sólo servidor de software. Pero sí fue necesario ajustar algunos parámetros para mejorar su rendimiento cuando el número de CPUs llegó a las 200, aproximadamente. Estos parámetros fueron el número de instancias NFS (para aumentar el paralelismo) y el tiempo de espera hasta dar un *time-out* cuando no hay respuesta a una petición. En cualquier caso, se ha comprobado que el acceso al servidor es especialmente lento para operaciones de escritura, como ocurre cuando hay que copiar una nueva versión del software.

Las colas que la VO de *cms* publica en el IS son *cms*, *cms_short*, *cms_long*, *cms_prod*, *dteam* y *ops*. Las tres primeras están accesibles para todos los miembros de la VO de *cms*. La cola de producción (*cms_prod*), sin embargo, tiene restringido el acceso a unos pocos usuarios a nivel de DN.

Al igual que en el PIC, los usuarios sin role de las colas *cms*, *cms_short* y *cms_long* se mapean a un usuario UNIX local cuya identidad es de la forma *cmsXYZ*. Sin embargo, tanto a los usuarios de la cola *cms_prod* como a los que tiene role de producción, se el asigna la cuenta local *cmsprod*, que en el caso del CIEMAT es única.

Los servicios desplegados en el CIEMAT se completan con un servidor BDII y una máquina con los servicios de FroNTier-cache.

La figura 3.6 muestra el número total de trabajos gestionados en el PIC durante el último año. Durante el último año se han gestionado más de 142000 trabajos, la mayoría de los cuales corresponden a CMS.

3.2.2.2. Gestión de los datos

En el CIEMAT se han desplegado los servicios para los tres tipos de SE más habituales: CASTOR, dCache y DPM. Cada uno de los tres SE responde a necesidades diferentes.

Los servicios de CASTOR, aparte de la gestión de los 11 servidores de disco, incluyen las funcionalidades del stager, el Name Server, y la interfaz con el robot de cintas. Estos servicios están instalados en uno

Figura 3.6: Número total de trabajos gestionados en el CIEMAT durante el último año para las distintas VOs.

de los servidores de disco. Cuando estos servicios se prestan a través del Grid están restringidos a la comunidad CMS. Sólo hay instalado un stager, a diferencia de centros de mayor envergadura, como el CERN, donde existe uno por experimento. El espacio disponible se divide en dos pools de discos, a los que se ha nombrado como POOL_USER y POOL_LCG. Esto es, sin embargo, relativamente transparente para los usuarios, que sólo ven la estructura de directorios virtual que les presenta el Name Server. Si el cliente es rfio la variable de entorno STAGER_POOL dice en qué pool escribir. Si el cliente es SRM o GridFTP lo determinan dos ficheros de configuración: un gridmapfile que mapea al cliente a un usuario local, y otro fichero que determina el pool que le corresponde.

La siguiente versión de CASTOR desarrollada en el CERN, conocida como CASTOR-2, es un producto que supera las necesidades de un centro Tier-2 típico, y requiere una serie de servicios adicionales (como una base de datos ORACLE, por ejemplo) y personal especializado para su mantenimiento. Por este motivo, para mejorar la gestión de los recursos de almacenamiento, en el CIEMAT se ha optado por migrar de CASTOR-1 a dCache. No existe experiencia local previa, pero muchos centros Tier-2 operan ya dCache y la escalabilidad y soporte técnico están garantizados. La instalación local en el CIEMAT está todavía, por tanto, en fase de pruebas. Actualmente gestiona 3 servidores de disco. La configuración del SE dCache en el CIEMAT posee algunas características propias. Se ha desplegado un servicio de GridFTP en cada uno de los servidores de disco para aumentar la paralelización. El servicio de gsidcap está instaldo en uno de los servidores de disco. El espacio reservado para cada VO está repartido entre servidores de disco, en lugar de agrupar todos los pools de la misma VO en un único servidor. Los servicios de GridFTP, gsidcap y srm son accesibles desde fuera de la LAN, mientras que los de administración, pnfs y dcap son privados (dcap se usa sólo para acceso desde dentro de la LAN). En el espacio de nombres virtual existen subdirectorios para *cms*, *biomed*, y *dteam*, respectivamente. En los dos últimos casos hay un pool asociado a cada uno de ellos. Sin embargo, este SE da servicio básicamente al experimento CMS, en particular para habilitar las transferencias de PhEDEx. Para CMS existen cuatro subdirectorios: dos para usuarios y dos para producción, donde en cada caso uno es permanente y otro temporal (o volátil). Cada uno de estos cuatro directorios está asociado a un pool distinto (llamados, respectivamente, *cmsuser_per*, *cmsuser_vol*, *cmsprod_per*, y *cmsprod_vol*). El servicio que gestiona el espacio de nombres de dCache, pnfs, está instalado en un nodo dedicado. No se ha instalado en los WN de la granja de computación para evitar la posible sobrecarga del sistema.

Finalmente, el SE de DPM, con un único servidor de disco, da servicio a la VOs de *biomed*, *fusion* y *calice*.

3.2.2.3. Instalación de los servicios

En el caso del CIEMAT, la instalación de los sistemas operativos también se hace con un kickstart, al igual que en el PIC. La instalación del middleware también se realiza con la herramienta oficial de instalación del LCG, Yaim. En el CIEMAT, por el contrario, no se hace uso de Quattor, pues está pensado para gestionar grandes infraestructuras de computación más que como una herramienta para la instalación de versiones de software.

El proceso de instalación consta de dos fases: la instalación de los paquetes de software usando un instalador de Red Hat Linux llamado Yum [88], y la configuración de los servicios dependiendo del tipo de nodo (UI, CE, WN...) Para facilitar el proceso de instalación de nuevos nodos se está experimentando con la virtualización del hardware de los WN.

3.2.2.4. Herramientas de monitorización

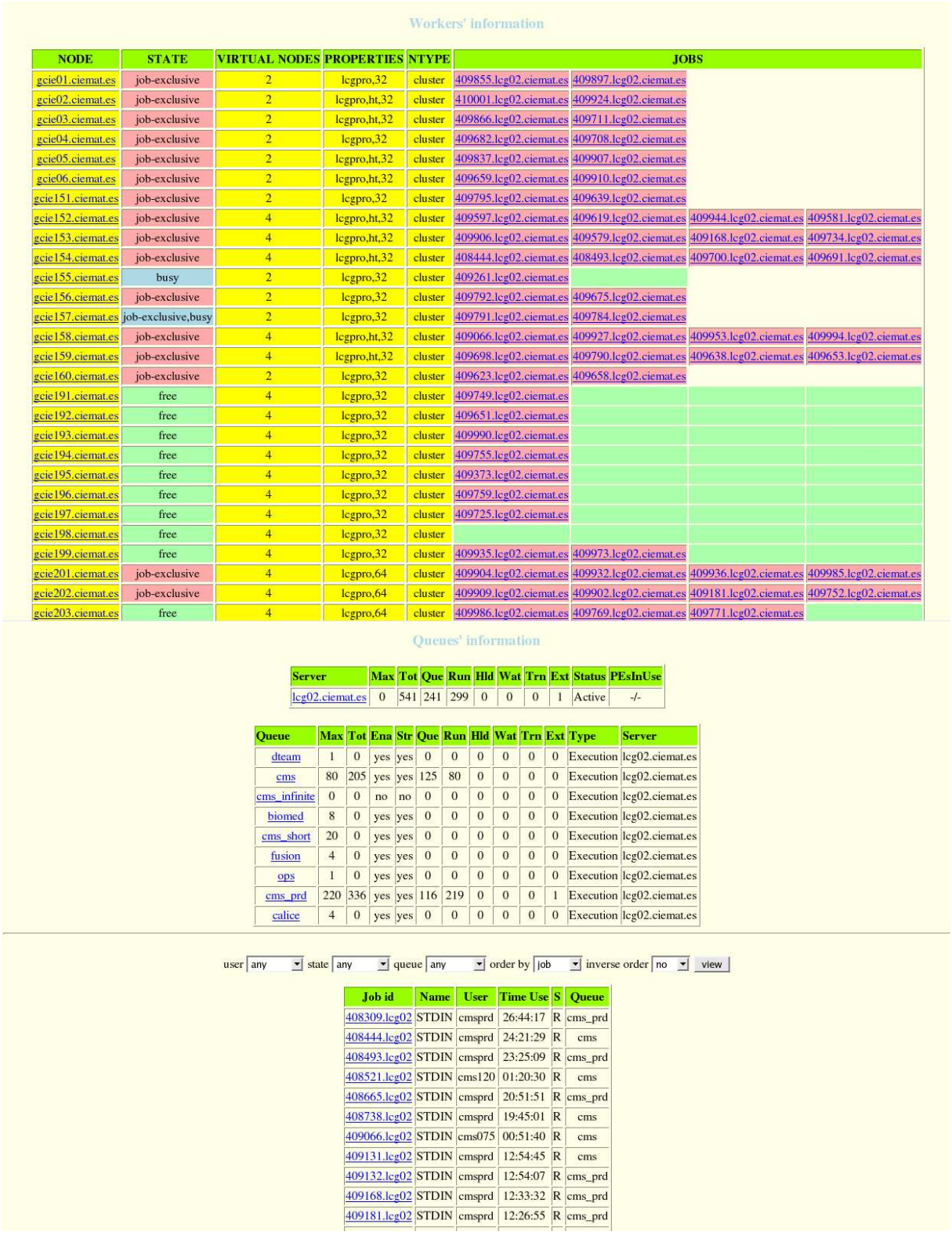
Se han instalado varias herramientas para la monitorización de los recursos y servicios desplegados. Algunas de ellas son utilidades estándar para servicios y sistemas Linux, otras son las propias de LCG, y un tercer grupo lo forman algunas utilidades específicas desarrolladas en el CIEMAT.

En el CIEMAT también se hace uso de ganglia. Existe una instancia en ejecución en cada máquina, y se usa también para la monitorización de la red local, y en particular del acceso a los servidores de discos. La unidad de comunicaciones del CIEMAT tiene una sonda en el *router* central que le permite extraer información sobre el tráfico diario en todo el CIEMAT. Como el uso de la red es mayor cuando PhEDEX está ejecutando operaciones de transferencia este sistema de monitorización resulta útil para examinar el ancho de banda máximo que se consigue en las operaciones de CMS. Carece, sin embargo, de un sistema de alertas (no avisa cuando alguno de los parámetros supera un cierto umbral). Toda la información de cada una de estas instancias de ganglia se recoge en un servidor central y se presentan los resultados agregados.

El paquete de GridICE viene incluido en la instalación del middleware, y puede activarse o no. En el CIEMAT este servicio está activado, y monitoriza algunos parámetros de funcionamiento de la granja de computación: número de trabajos en ejecución, estado de los servicios de cada máquina, estado global del sistema, etc. Esta herramienta ofrece cierto grado de redundancia, pero con una granularidad en la detección de fallos menor que la de ganglia.

Otras herramientas de monitorización estándar que también se usan en el CIEMAT son gstat [89] (que hace un examen de los centros mediante la información suministrada por los BDII locales) y GridView (que manda pequeños trabajos para examinar la disponibilidad y funcionamiento de los servicios en los centros). La granularidad temporal de gstat es mayor incluso que la de GridICE, mientras que la de GridView es del orden de una hora. gstat sí dispara una alerta si alguno de los servicios falla.

En el CIEMAT se han desarrollado varias herramientas para la monitorización de los recursos y los servicios. Una de estas herramientas vigila que el servicio de BDII global que corresponde por defecto al CIEMAT (ubicado en el Tier-1 del PIC) está activo. Éste es un componente fundamental del Sistema de Información y permite al centro ser visible para el resto del Grid, y detectar a tiempo cualquier anomalía permite la migración temporal a otro BDII diferente para evitar que el centro deje de ser accesible. Otra herramienta desarrollada inspecciona los ficheros de output de los trabajos ejecutados en el centro para ver si su tiempo de ejecución es siempre anormalmente bajo en algún nodo en concreto. Esto suele ser un factor indicativo de la existencia de problemas en ese nodo, que actúa como un *agujero negro* provocando la ejecución fallida de gran cantidad de trabajos. Desactivar estos nodos problemáticos a tiempo permite mejorar el rendimiento global del sistema y superar satisfactoriamente los tests de SAM. Para la monitorización de los trabajos en ejecución se desarrolló una aplicación que permite la visualización, via web, del uso en tiempo real de los Worker Nodes [90]. El aspecto de esta interfaz web se puede ver en la figura 3.7.



Workers' information

NODE	STATE	VIRTUAL NODES	PROPERTIES	NTYPE	JOBS			
gcie01.ciemat.es	job-exclusive	2	lcgpro,32	cluster	409855.lcg02.ciemat.es	409897.lcg02.ciemat.es		
gcie02.ciemat.es	job-exclusive	2	lcgpro,ht,32	cluster	410001.lcg02.ciemat.es	409924.lcg02.ciemat.es		
gcie03.ciemat.es	job-exclusive	2	lcgpro,ht,32	cluster	409866.lcg02.ciemat.es	409711.lcg02.ciemat.es		
gcie04.ciemat.es	job-exclusive	2	lcgpro,32	cluster	409682.lcg02.ciemat.es	409708.lcg02.ciemat.es		
gcie05.ciemat.es	job-exclusive	2	lcgpro,ht,32	cluster	409837.lcg02.ciemat.es	409907.lcg02.ciemat.es		
gcie06.ciemat.es	job-exclusive	2	lcgpro,ht,32	cluster	409659.lcg02.ciemat.es	409910.lcg02.ciemat.es		
gcie151.ciemat.es	job-exclusive	2	lcgpro,32	cluster	409795.lcg02.ciemat.es	409639.lcg02.ciemat.es		
gcie152.ciemat.es	job-exclusive	4	lcgpro,ht,32	cluster	409597.lcg02.ciemat.es	409619.lcg02.ciemat.es	409944.lcg02.ciemat.es	409581.lcg02.ciemat.es
gcie153.ciemat.es	job-exclusive	4	lcgpro,ht,32	cluster	409906.lcg02.ciemat.es	409579.lcg02.ciemat.es	409168.lcg02.ciemat.es	409734.lcg02.ciemat.es
gcie154.ciemat.es	job-exclusive	4	lcgpro,ht,32	cluster	408444.lcg02.ciemat.es	408493.lcg02.ciemat.es	409700.lcg02.ciemat.es	409691.lcg02.ciemat.es
gcie155.ciemat.es	busy	2	lcgpro,32	cluster	409261.lcg02.ciemat.es			
gcie156.ciemat.es	job-exclusive	2	lcgpro,32	cluster	409792.lcg02.ciemat.es	409675.lcg02.ciemat.es		
gcie157.ciemat.es	job-exclusive,busy	2	lcgpro,32	cluster	409791.lcg02.ciemat.es	409784.lcg02.ciemat.es		
gcie158.ciemat.es	job-exclusive	4	lcgpro,ht,32	cluster	409066.lcg02.ciemat.es	409927.lcg02.ciemat.es	409953.lcg02.ciemat.es	409994.lcg02.ciemat.es
gcie159.ciemat.es	job-exclusive	4	lcgpro,ht,32	cluster	409698.lcg02.ciemat.es	409790.lcg02.ciemat.es	409638.lcg02.ciemat.es	409653.lcg02.ciemat.es
gcie160.ciemat.es	job-exclusive	2	lcgpro,32	cluster	409623.lcg02.ciemat.es	409658.lcg02.ciemat.es		
gcie191.ciemat.es	free	4	lcgpro,32	cluster	409749.lcg02.ciemat.es			
gcie192.ciemat.es	free	4	lcgpro,32	cluster	409651.lcg02.ciemat.es			
gcie193.ciemat.es	free	4	lcgpro,32	cluster	409990.lcg02.ciemat.es			
gcie194.ciemat.es	free	4	lcgpro,32	cluster	409755.lcg02.ciemat.es			
gcie195.ciemat.es	free	4	lcgpro,32	cluster	409373.lcg02.ciemat.es			
gcie196.ciemat.es	free	4	lcgpro,32	cluster	409759.lcg02.ciemat.es			
gcie197.ciemat.es	free	4	lcgpro,32	cluster	409725.lcg02.ciemat.es			
gcie198.ciemat.es	free	4	lcgpro,32	cluster				
gcie199.ciemat.es	free	4	lcgpro,32	cluster	409935.lcg02.ciemat.es	409973.lcg02.ciemat.es		
gcie201.ciemat.es	job-exclusive	4	lcgpro,64	cluster	409904.lcg02.ciemat.es	409932.lcg02.ciemat.es	409936.lcg02.ciemat.es	409985.lcg02.ciemat.es
gcie202.ciemat.es	job-exclusive	4	lcgpro,64	cluster	409909.lcg02.ciemat.es	409902.lcg02.ciemat.es	409181.lcg02.ciemat.es	409752.lcg02.ciemat.es
gcie203.ciemat.es	free	4	lcgpro,64	cluster	409986.lcg02.ciemat.es	409769.lcg02.ciemat.es	409771.lcg02.ciemat.es	

Queues' information

Server	Max	Tot	Que	Run	Hld	Wat	Trn	Ext	Status	PEsInUse
lcg02.ciemat.es	0	541	241	299	0	0	0	1	Active	-/-

Queue	Max	Tot	Ena	Str	Que	Run	Hld	Wat	Trn	Ext	Type	Server
dteam	1	0	yes	yes	0	0	0	0	0	0	Execution	lcg02.ciemat.es
cms	80	205	yes	yes	125	80	0	0	0	0	Execution	lcg02.ciemat.es
cms_infinite	0	0	no	no	0	0	0	0	0	0	Execution	lcg02.ciemat.es
biomed	8	0	yes	yes	0	0	0	0	0	0	Execution	lcg02.ciemat.es
cms_short	20	0	yes	yes	0	0	0	0	0	0	Execution	lcg02.ciemat.es
fusion	4	0	yes	yes	0	0	0	0	0	0	Execution	lcg02.ciemat.es
ops	1	0	yes	yes	0	0	0	0	0	0	Execution	lcg02.ciemat.es
cms_prd	220	336	yes	yes	116	219	0	0	0	1	Execution	lcg02.ciemat.es
calice	4	0	yes	yes	0	0	0	0	0	0	Execution	lcg02.ciemat.es

user any state any queue any order by job inverse order no view

Job id	Name	User	Time Use	S	Queue
408309.lcg02	STDIN	cmsprd	26:44:17	R	cms_prd
408444.lcg02	STDIN	cmsprd	24:21:29	R	cms
408493.lcg02	STDIN	cmsprd	23:25:09	R	cms_prd
408521.lcg02	STDIN	cms120	01:20:30	R	cms
408665.lcg02	STDIN	cmsprd	20:51:51	R	cms_prd
408738.lcg02	STDIN	cmsprd	19:45:01	R	cms
409066.lcg02	STDIN	cms075	00:51:40	R	cms
409131.lcg02	STDIN	cmsprd	12:54:45	R	cms
409132.lcg02	STDIN	cmsprd	12:54:07	R	cms_prd
409168.lcg02	STDIN	cmsprd	12:33:32	R	cms_prd
409181.lcg02	STDIN	cmsprd	12:26:55	R	cms_prd

Figura 3.7: Herramienta de monitorización via web del cluster local del CIEMAT.

3.3. Infraestructura de red de los centros españoles

No sólo la potencia de cálculo y la capacidad de almacenamiento se están incrementando en los centros españoles, la infraestructura de red también se actualiza para poder cumplir con una de las operaciones básicas del modelo de computación: la recogida de los datos procedentes del CERN, su distribución a los centros Tier-2 asociados, y la recepción las muestras Monte Carlo producidas en los Tier-2. Una buena conexión de red, tanto con el Tier-0 como con sus Tier-2, es imprescindible para que un centro Tier-1 cumpla con sus objetivos. De igual manera son importantes las conexiones entre los centros Tier-1, ya que los resultados de las tareas de reprocesamiento de los datos, que se llevarán a cabo unas cuantas veces al año en los centros Tier-1, se deberán distribuir entre todos ellos para que cada uno disponga de una copia de los nuevos datos reconstruidos en formato AOD.

La conexión con el CERN se establece a través de la red óptica privada del LHC. Un esquema de esta red privada se puede ver en la figura 3.8 La arquitectura de esta red privada está basada en *light paths* de 10 Gb/s permanentes formando una Red Privada Óptica (OPN) para el LHC [91]. El principal objetivo es asegurar la calidad del servicio en el tráfico entre el Tier-0 y los Tier-1. También puede llevar tráfico entre los centros Tier-1 si es necesario, aunque prioriza el tráfico entre el Tier-0 y los Tier-1. No llevará tráfico a los Tier-2. La red privada del LHC está implementada sobre dos componentes:

- La red europea GEANT. Es una red híbrida IP, con núcleo de fibra oscura, con conmutación de circuitos de 10 Gigabits.
- La red LHCNet [92] con un enlace de 10 Gigabits con USA.

Existen principalmente tres rutas de acceso desde el CERN a España a través de la red GEANT:

- A través de París, con una conexión de 2 x 10 Gb/s.
- A través de Milán, con una capacidad de 10 Gb/s.
- Directamente desde Ginebra a Madrid, con una conexión de 10 Gb/s.

La figura 3.9 muestra un esquema con los tramos de red que conectan los centros españoles con el resto de Europa. Los datos se distribuyen a nivel europeo a través la red europea GEANT [93] (arriba) y la interconexión entre los centros españoles se realiza a través de la red nacional RedIRIS [94] (abajo). En el caso de Cataluña, los centros académicos y universidades están interconectados a través de una red específica llamada Anella Cientifica [95].

La interconexión entre los centros españoles y de éstos con el CERN se puede ver en la figura 3.10. En la figura se pueden ver el ancho de banda y las latencias aproximadas de las diferentes secciones de la red entre el CERN y los centros españoles. Estos valores están resumidos en la tabla 5.5.

El tráfico desde el CERN alcanza España a través de la red europea Geant-2, directamente desde Ginebra, con un ancho de banda de 10 Gb/s, y llega hasta un nodo en Madrid y otro en Barcelona. Desde estos nodos, la red académica RedIRIS transporta los datos dentro de España con un ancho de banda de 2.5 Gb/s. Existen 2 tramos para el acceso hasta IFCA a través de RedIRIS, uno desde Madrid y otro desde Barcelona, pasando por Santander. El ancho de banda hasta el PIC a través de Anella Cientifica estaba limitado a 1 Gb/s. La latencia total desde el CERN al PIC son del orden de 12 ms. Las latencias entre el PIC y el CIEMAT y entre el PIC y el IFCA son de 10 ms y 16 ms, respectivamente. En el año 2007 se ha desplegado en España un enlace de 10 Gb/s como parte de la infraestructura de Geant-2. Gracias a este enlace el PIC ya forma parte de la red óptica privada del LHC.

Una vez que el tráfico llega a los centros, la conexión de los mismos es de 2.5 Gigabits en el caso del CIEMAT. Existe una conexión de 10 Gibabits, pero no se hace uso de todo el ancho de banda por las

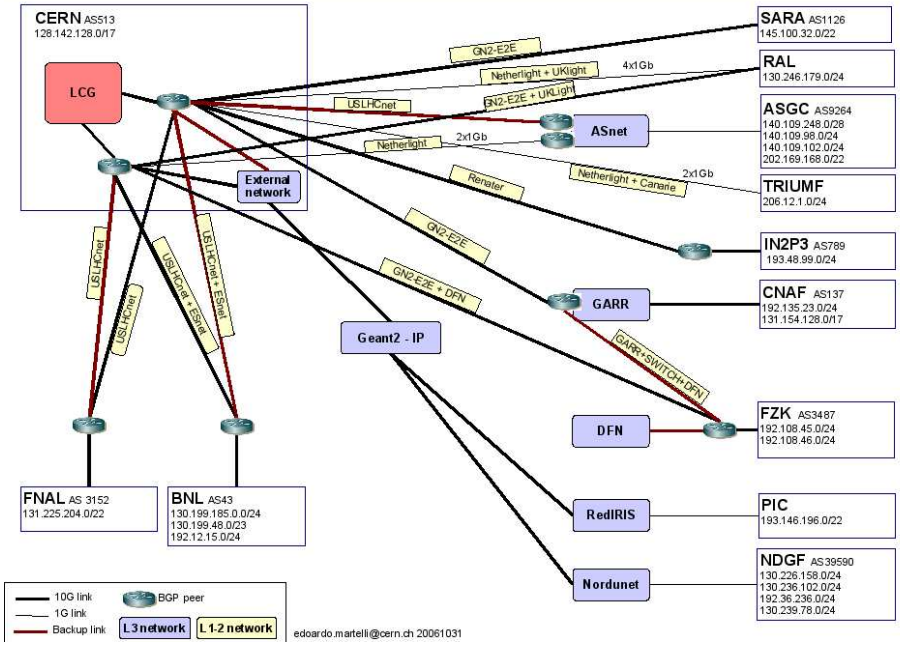


Figura 3.8: Infraestructuras de red del CERN. La red óptica privada permite una excelente conexión entre el CERN y los grandes centros de computación del LCG. Entre ellos se encuentran casi todos los centros Tier-1 de CMS: FNAL en USA, RAL en UK, IN2P3 en Francia, CNAF en Italia, FZK en Alemania y PIC en España.

Red	**Ámbito**	**Ancho de banda**	**Latencias máximas**
GEANT	Europeo	10 Gb/s	11 ms
RedIRIS	España	2.5 Gb/s	14 ms
Anella Cientifica	Cataluña	1 Gb/s	1 ms

Tabla 3.13: Infraestructura de red desde el CERN hasta los centros españoles para el DC04.

limitaciones de los servicios internos. La conexión del PIC con RedIRIS también es de 10 Gigabits, y se usa todo el ancho de banda disponible. En el interior de cada centro, tanto en el CIEMAT como el PIC, los nodos están interconectados entre sí mediante redes del tipo Ethernet de 1 Gigabit.

Con estas infraestructuras se ha conseguido un ritmo de transferencia de datos del PIC al CIEMAT a través de SRM de unos 10 MB/s en las transferencias de ficheros individuales con una tasa agregada de 50 MB/s, de manera que con unas pocas transferencias en paralelo se consiguen satisfacer las necesidades en las transferencias de datos del PIC al CIEMAT. El flujo de datos desde el CIEMAT hacia el PIC es mucho más modesto, pues las necesidades son del orden de un 1 TB/día, lo que equivale a unos 12 MB/s.

El rendimiento local de la LAN en la copia de un fichero desde el SE a un WN es mucho mayor, de unos 40 MB/s aproximadamente, aunque el modelo de computación de CMS no contempla las copias desde los SEs a los WNs, sino la lectura directa desde los WNs de los datos almacenados en el SE.

Para los trabajos de análisis CMS require del orden de 1MB/s/trabajo. Se ha probado, con éxito, la ejecución de unos 100 trabajos en paralelo con un throughput agregado de 100 MB/s, y no se han observado problemas de escala.

Finalmente, el throghput local de escritura desde los WN al SE es del orden de unos 20 MB/s.

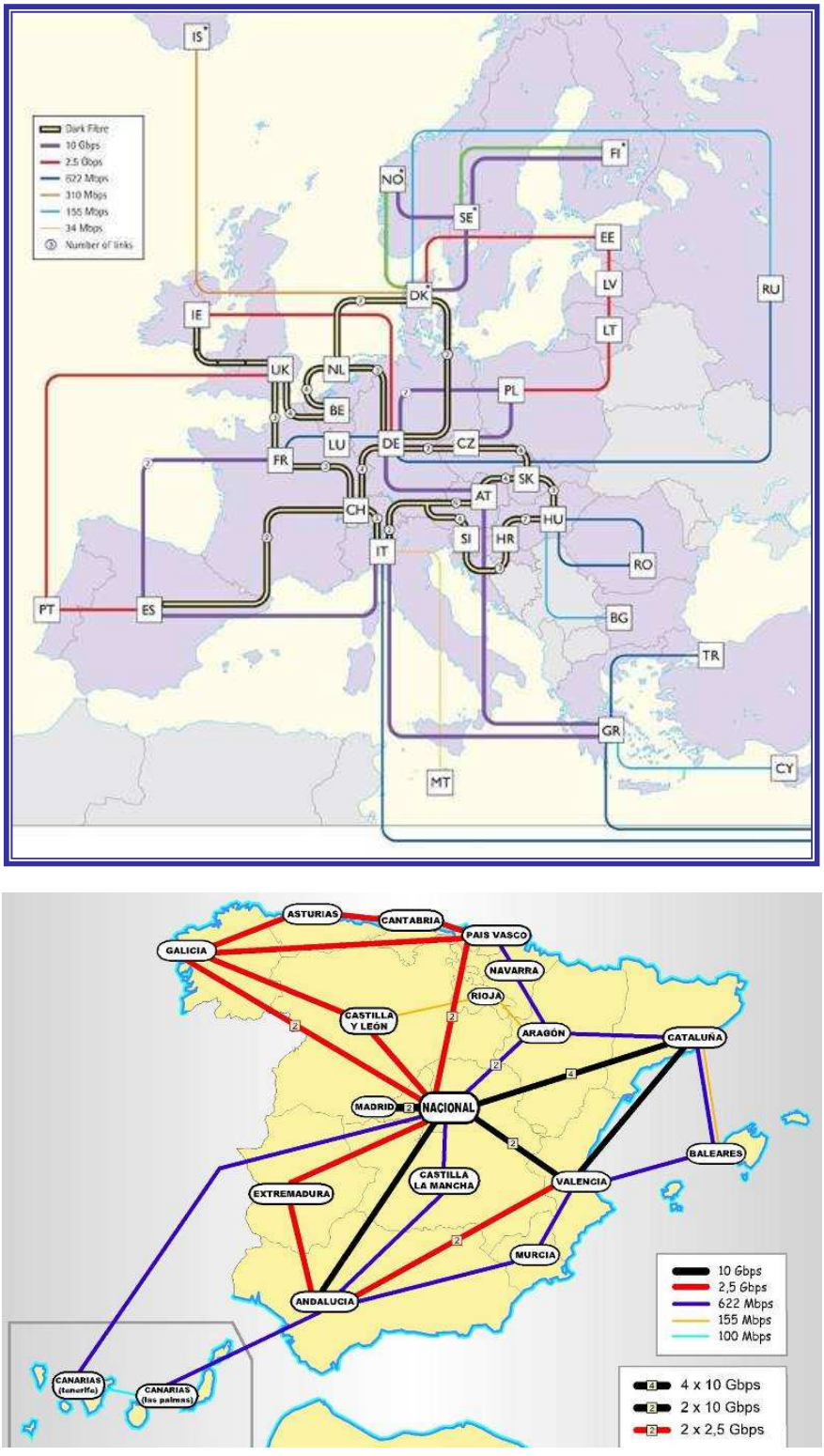


Figura 3.9: Infraestructuras de red del CERN, europea y española.

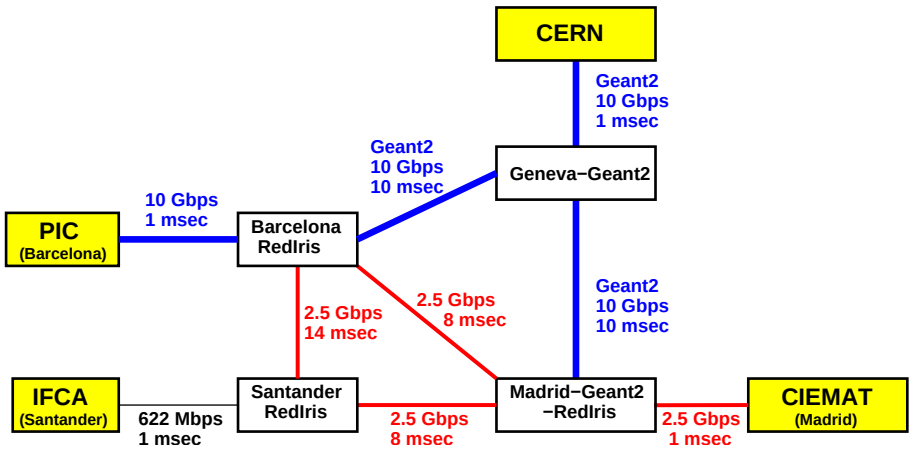


Figura 3.10: Diagrama de red con el ancho de banda y latencias aproximadas de las diferentes secciones de la red entre el CERN y los centros españoles.

Capítulo 4

Desarrollo del sistema de computación de CMS

Se ha llevado a cabo un gran esfuerzo para desarrollar todas las componentes que forman el sistema de computación de CMS. Este esfuerzo, que aún continúa, tiene como objetivo conseguir un uso fiable y eficiente de todos recursos que el Grid pone a disposición del experimento. El principal objetivo es conseguir un sistema que disponga de la funcionalidad básica en el momento en que comience la toma de datos del LHC. Estas funcionalidades que se persiguen son las que permiten la implementación de los sistemas de gestión de trabajos y de datos.

En el caso del WMS, la migración al LCG del sistema de producción de datos simulados Monte Carlo que se describe en este capítulo permitió hacer, por primera vez, un uso intensivo y más eficiente de la gran cantidad de recursos de computación disponibles en el Grid. Gracias a esto se ha conseguido un incremento muy significativo en el volumen de datos simulados en los últimos años. La simulación de datos Monte Carlo es importante tanto durante la toma de datos del LHC como antes de su comienzo. Será crucial para los estudios de física y para alcanzar una mejor comprensión del detector. Los experimentos de Física de Altas Energías dependen críticamente de la precisión de los generadores y de las simulaciones del detector. Los sucesos simulados son necesarios para la optimización del diseño del detector, su calibración, y los estudios de física. La magnitud de las incertidumbres asociadas al descubrimiento de partículas, o la medida de sus masas o secciones eficaces, está muy relacionada con la precisión con la que las simulaciones describen el funcionamiento del detector midiendo leptones, fotones y hadrones. Por tanto, es esencial para el éxito de un experimento de Física de Altas Energías una buena comprensión y ajuste de las herramientas de simulación que estén en buen acuerdo con las medidas reales y, por tanto, disponer de un sistema de producción de Monte Carlo eficiente capaz de suministrar la enorme cantidad de datos necesarios. En el apéndice A se describen las características de la simulación MC en los experimentos de Física de Altas Energías.

Llevar a cabo las tareas de producción de forma masiva en LCG también ha permitido encontrar algunas limitaciones de los paquetes de software de análisis del experimento. Estas limitaciones dificultaban la ejecución, de forma eficiente, de las tareas de análisis en un entorno tan distribuido como el Grid. Una vez corregidas estas limitaciones, gracias a la experiencia adquirida durante la migración y operación en el Grid del sistema de producción Monte Carlo, la mayor parte de los análisis físicos del experimento se llevan a cabo en LCG.

Por otra parte, dado que el modelo de computación de CMS está basado principalmente en la gestión de los datos, y que las primeras herramientas disponibles en LCG eran poco fiables y robustas, se hizo necesario el desarrollo de un sistema de gestión de datos que incorporase las funcionalidades requeridas. Se ha desarrollado un sistema de transferencia de datas eficiente, robusto, que distribuye los datos de acuerdo a las políticas y prioridades del experimento, que incorpora la organización de los datos propia

de CMS (organización en bloques, Datasets, etc.) Una vez que las herramientas Grid han ido ganando en fiabilidad incorporando nuevas prestaciones, el sistema de gestión de datos las ha ido integrando de forma paulatina. El objetivo final es construir un sistema que incorpore todas las funcionalidades requeridas y que sea escalable para poder gestionar varios PB de datos por año.

En este capítulo se presenta el desarrollo del sistema de producción Monte Carlo en LCG [96, 97, 98] y del sistema de distribución de datos [53, 99].

4.1. Migración del sistema de producción Monte Carlo a LCG

La producción de datos Monte Carlo tradicional en CMS se ejecutaba completamente en granjas locales de PCs y era gestionada por un operador local en cada centro. Estaba basada en el acceso directo a los recursos de computación y a los trabajos y no disponía, por tanto, de las herramientas necesarias para realizarse en un entorno distribuido.

Este sistema de producción Monte Carlo ha sido adaptado para poder usar, de forma intensiva y eficiente, los recursos Grid disponibles. Este proceso se ha llevado a cabo en dos fases. Durante la primera fase se adaptó el antiguo sistema de producción, dotándolo de las herramientas necesarias para trabajar de forma remota en un entorno distribuido. Este sistema adaptado ha sido utilizado durante un período de unos 500 días, aproximadamente. En una segunda fase, se ha desarrollado completamente un sistema nuevo basándose en la experiencia adquirida durante la primera etapa. Este apartado y el siguiente describen, respectivamente, estas dos fases en la evolución de la producción Monte Carlo para CMS desde el modo en granja local hasta su completa integración en Grid.

4.1.1. Sistema tradicional de producción Monte Carlo en CMS

La herramienta oficial de CMS para la simulación Monte Carlo ha sido, hasta mediados de 2006, un paquete de software llamado McRunjob [100]. Se usó McRunjob durante muchos años con gran éxito para realizar la producción Monte Carlo en granjas locales de PCs. En McRunjob, cada etapa de la simulación (generación, simulación, digitalización y reconstrucción) se ejecuta por separado, produciendo un Data Tier diferente (hits de la simulación, digis de la digitalización y objetos físicos en formato DST de la reconstrucción). Cada paso se procesa con una geometría y versión del software definidas. Los grupos de física y del detector solicitan un determinado número de sucesos de un Dataset y etapa de la simulación específicos. Por razones prácticas, el número total de sucesos se divide en grupos más pequeños (llamados *assignments*). Estos assignments se componen de trabajos (conocidos como *runs*), cada uno de los cuales procesa, generalmente, unos 1000 sucesos.

En una primera fase, McRunjob se pone en contacto con una base de datos específica (RefDB) [101], de donde obtiene toda la información necesaria para la preparación de los trabajos: lista de sucesos a simular, cartas de datos con información específica sobre dichos sucesos, especificaciones sobre los datos de entrada, una plantilla para generar el trabajo que se va a ejecutar, etc. Una vez que los trabajos son creados apropiadamente, McRunjob está instrumentado para enviarlos para su ejecución a granjas de PCs locales manejadas por diferentes sistemas de gestión de colas, así como para monitorizar su estado. Se adoptó una arquitectura modular, con interfaces configurables, para permitir una extensión fácil a nuevos entornos de trabajos y aplicaciones. Cada aplicación, base de datos o servicio externo diferente se modela internamente como una componente, conocida como *Configurator*, que guarda los metadatos relevantes con la descripción los parámetros de la aplicación, los resultados de las consultas a la base de datos o las llamadas a los servicios. Los Configurators también son responsables de la generación de los trabajos para la consecución de un determinado flujo de trabajos de procesamiento de datos. Se mantienen en un contenedor, llamado *Linker*, que coordinaba las acciones de los Configurators, facilita la comunicación entre ellos, y unifica todos los scripts específicos de las aplicaciones en un único trabajo. Una vez creados, otro Configurator, encargado de enmascarar los servicios de ejecución, los envía para su ejecución. Para

manejar el flujo de trabajo el usuario proporciona un script con macros que son interpretados por el Linker como llamadas a los Configurators y al propio Linker.

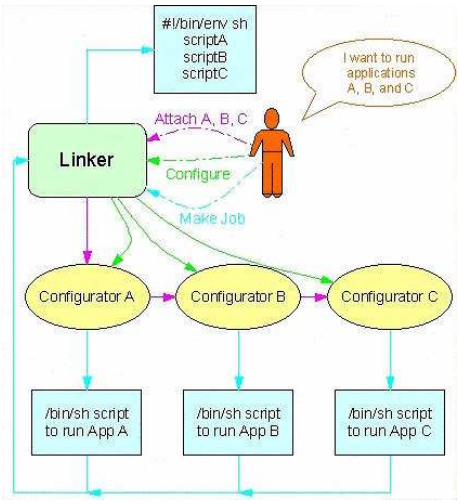


Figura 4.1: Relación entre las distintas componentes de McRunjob y su actuación para enviar un trabajo.

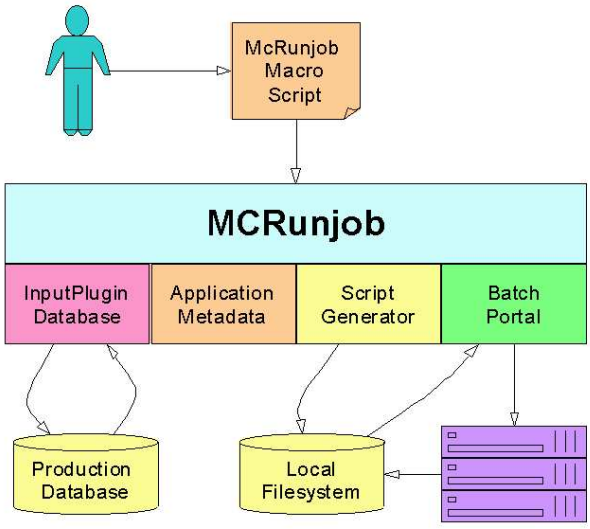


Figura 4.2: Acoplamiento entre McRunjob y los elementos externos involucrados en la simulación Monte Carlo.

En el caso del experimento CMS, los trabajos de generación no tienen input y generan un output pequeño (*ntuplas* de 10 a 50 MB). Son, por tanto, trabajos que sólo consumen tiempo de CPU y necesitan pocos minutos para su ejecución (aunque podrían necesitarse varias horas excepcionalmente si se imponen condiciones de filtrado muy exigentes).

Los trabajos de simulación necesitan como input los ficheros creados durante la generación. Son trabajos que hacen uso intensivo de la CPU y de la memoria, necesitan entre 24 y 48 horas para completar el procesamiento, y generan un output relativamente grande (del orden de 500 MB) en tres ficheros, conocidos como *EVent Data* (EVD). El fichero EVD0 contiene cierta información de cabecera. El menor de estos ficheros EVD es del orden de 100 kB.

Los trabajos de digitalización tienen menores requisitos de CPU y memoria. En general, de 5 a 10 horas son suficientes para finalizar el procesamiento. Sin embargo, estos trabajos ejecutan operaciones intensivas de lectura del SE debido a la necesidad de un acceso continuo a los datos de PU a través de la LAN. Generan un output grande, similar al de la simulación.

Por último, los trabajos de reconstrucción necesitan menos CPU (unas 5 horas) y generan un output menor (de unos 200 MB) en dos ficheros EVD.

En el anterior entorno de trabajo se usaba para la digitalización y reconstrucción un paquete de software conocido como ORCA (*Object oriented Reconstruccion for CMS Analysis*) [102]. La herramienta usada para la simulación era OSCAR (*Object oriented Simulation for CMS Analysis and Reconstruction*) [103]. Los servicios básicos necesarios para todo el procesamiento eran proporcionados por un entorno llamado COBRA (*Coherent Object-oriented Base for Reconstruccion, Analysis and Simulation*) [104]. Actualmente todo esta infraestructura de software ha sido sustituida por un único paquete llamado CMSSW [105].

Dado un Dataset, y para poder ejecutar la digitalización, ORCA necesita unos ficheros de metadatos de COBRA con la descripción de la colección completa de los sucesos simulados. Estos ficheros de metadatos se crean a partir de los ficheros EVD0 generados durante la etapa de simulación. El proceso por el cual se generan estos metadatos (conocido como *metadata attachment*) necesita, por tanto, acceso directo POSIX a los ficheros con los datos producidos.

4.1.2. Integración de las herramientas Grid en el sistema de producción

Aunque originalmente McRunjob fue utilizado exclusivamente en operaciones de producción en granjas de PC locales y no estaba adaptado para ser usado en un entorno Grid, gracias a su alto grado de modularidad y su adaptabilidad para trabajar en diferentes entornos, ha sido posible proporcionarle las componentes necesarias para operar en LCG. La creación de un nuevo Configurator, capaz de traducir las órdenes básicas (creación de los trabajos, envío, etc.) a operaciones del WMS de LCG, ha permitido hacer de McRunjob una interfaz válida de acceso a los recursos del Grid. Esta migración del sistema de producción a LCG no ha sido, sin embargo, una tarea sencilla. El sistema original estaba pensado para trabajar en modo local (necesitaba acceso directo a los ficheros de datos, la gestión y monitorización de los trabajos estaban basadas en la existencia de ciertos ficheros que cambiaban de directorio en un sistema de ficheros compartidos accesible desde los WNs, etc.) y ha sido completamente adaptado a un entorno distribuido de recursos remotos.

Tras ser preparados por McRunjob en base a las especificaciones guardadas en RefDB, los trabajos se envían al Grid usando las herramientas del WMS de LCG. McRunjob debe operarse en un UI con las aplicaciones cliente del WMS instaladas. De la forma usual, todos los requisitos del trabajo (CPU, memoria, versión de OSCAR/ORCA, etc.) se especifican en el fichero JDL correspondiente. Este fichero, junto con todos los demás necesarios para la ejecución del trabajo, se envía a un RB encargado de mandar el trabajo al centro más apropiado que cumpla todos los requisitos impuestos. Un centro es aceptado sólo si la versión del software de CMS requerida está disponible. En el anterior entorno de trabajo, el software de producción estaba empaquetado en archivos autónomos (conocidos como *DAR files*) que contienen todo lo necesario para ejecutar los trabajos de CMS, incluyendo las librerías del sistema. Evidentemente, tras la instalación de los paquetes DAR, el tag del software correspondiente debe publicarse en el IS para hacer el centro disponible para la producción.

Una vez comienza su ejecución en un WN, los trabajos preguntan al catálogo global de réplicas del LCG (RLS) para localizar el SE con los datos de input. Para un LFN dado, el catálogo devuelve el SURL que es usado por las herramientas de replicación de LCG para copiar los ficheros al WN. Los ficheros EVD de output se guardan en un SE de destino, fijado en las especificaciones del trabajo, y se registran en el catálogo RLS. Finalmente, cuando un trabajo finaliza se genera un fichero con la descripción de los datos generados. Este fichero se conoce comúnmente con el nombre de *summary file*. El summary file se devuelve a través del RB, en el Output Sandbox, y su información se registra en RefDB para llevar la contabilidad de las actividades de producción. Esta contabilidad permite, por ejemplo, conocer qué ficheros han sido creados o qué datos hay disponibles para el análisis.

Desde la primera implementación en LCG se usa BOSS como herramienta para el envío, monitorización y bookkeeping de los trabajos de producción. La figura 4.3 muestra un diagrama con los elementos involucrados en la gestión de los trabajos de producción en LCG con McRunjob.

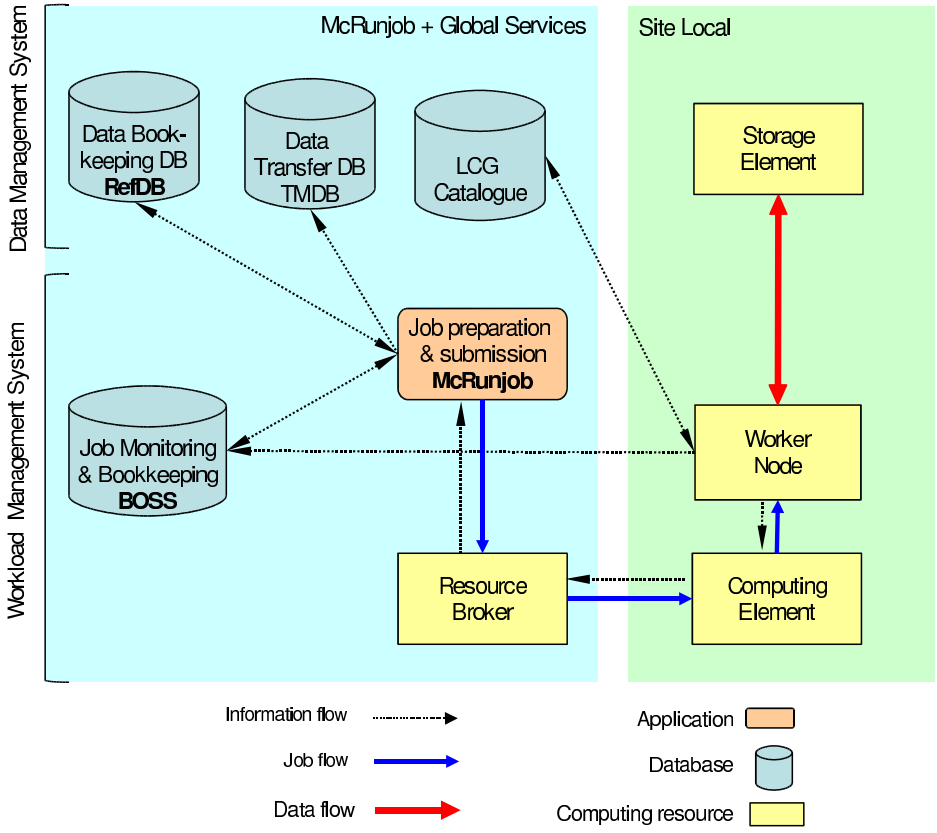


Figura 4.3: Flujo de los trabajos de producción Monte Carlo en LCG con McRunjob.

4.1.3. Optimización del sistema

Tras la migración de McRunjob al Grid, las primeras operaciones fueron bastante ineficientes, y dieron como resultado un ritmo de producción bajo. Estas dificultades iniciales se deben a la naturaleza intrínsecamente distribuida del sistema, lo que implica un aumento de las latencias, dificultad para encontrar y corregir problemas en trabajos ejecutados remotamente, falta de madurez del middleware, servicios y aplicaciones Grid, poca robustez en las operaciones de escritura y lectura de datos, etc. Se han identificado algunos de los problemas causantes de esta baja eficiencia: tiempos altos en las operaciones de envío

de los trabajos y de recuperación del output, alta tasa de fallo durante estas operaciones, problemas con la configuración local de los centros, inestabilidades en los servicios Grid, problemas al copiar los ficheros de input desde el SE al WN y los de output desde el WN al SE, etc. Además, la identificación de la causa de error en los trabajos fallidos era difícil, al carecer el sistema de un completo listado de códigos de error. La necesidad de una inspección visual de los ficheros de output para identificar la causa de fallo introduce considerables retrasos.

Se han ido solventando cada una de estas dificultades, lo que se traduce en un aumento considerable de la eficiencia, con el consiguiente incremento en el ritmo de producción.

Creación atómica de metadatos

El proceso de metadata attachment por el cual se crean los metadatos de COBRA necesita que los ficheros EVD0 estén accesibles via POSIX. Sin embargo, en el caso de la producción en LCG, el output de los trabajos de simulación y digitalización puede quedar muy distribuido entre varios SE remotos a los que no es posible acceso POSIX directo. Por otro lado, el tiempo que supone recoger todos los ficheros EVD en un único punto antes de comenzar el proceso de metadata attachment es irrealizable en la práctica, dados la cantidad y tamaño de estos ficheros EVD. Este fue el principal impedimento para conseguir un alto rendimiento en las operaciones de producción en Grid.

La solución que se ha implementado para solventar esta dificultad recibe el nombre de generación atómica de metadatos (o *atomic metadata attachment*). Las cabeceras de los ficheros de metadatos, necesarias para la creación de los mismos, se hacen disponibles para los trabajos de digitalización y reconstrucción descargándolos de un SE, y el metadata attachment se lleva a cabo en el momento de la ejecución del trabajo en el propio WN, sólo para el run que se va a procesar. Esta operación introduce un retraso de tan sólo unos pocos segundos.

McRunjob guarda las cabeceras de los ficheros de metadatos en uno o más SE en el momento de la creación de los trabajos. Esta operación se lleva a cabo en la UI donde se ejecuta McRunjob, una sóla vez por cada Dataset y etapa de producción, independientemente del número de trabajos que se estan preparando para enviarlos a LCG.

Finalmente, para poder analizar los datos, es necesario producir otros ficheros de metadatos correspondientes a la colección completa de sucesos simulados, digitalizados y reconstruidos. El sistema de transferencia de datos recoge los ficheros EVD para los diferentes pasos de la producción y los transfiere a los centros de análisis, donde finalmente se lleva a cabo la operación de metadata attachment.

Reducción del Input Sandbox y Output Sandbox

Otro de los motivos de ineficiencia durante las primeras operaciones del sistema en LCG fue el hecho de que todos los ficheros necesarios para la ejecución de los trabajos se enviaban desde la UI al CE, a través del RB, mediante el Input Sandbox. Este Input Sandbox incluía inicialmente los scripts de inicialización y los ficheros de metadatos de COBRA. Se ha comprobado que un Input Sandbox demasiado grande introduce un retraso no despreciable en la operación de envío de los trabajos y puede causar problemas en el RB. La solución que se ha implementado para evitar estos incovenientes ha sido guardar directamente algunos ficheros (los ficheros de metadatos de COBRA y los ficheros XML de POOL[1] [106]), empaquetados en un único fichero comprimido, en varios SEs, y registrar todas las copias en el catálogo de réplicas. Esta operación se realiza una única vez para cada etapa de la producción y para cada Dataset. Al no estar incluidos en el Input Sandbox se reduce considerablemente su tamaño, el tiempo necesario para su envío, y la probabilidad de fallo. Los trabajos pueden entonces encontrar las copias preguntando al catálogo y descargarse una de ellas antes de iniciar la ejecución de la tarea de simulación.

[1] Primer sistema que se implantó para guardar datos y metadatos para el LCG.

Tratamiento del Pile-Up

Para poder incluir en la digitalización los sucesos de pile-up (ver el apéndice A para una descripción de este tipo de sucesos) se preparan con antelación grandes muestras (de unos 100 GB) con un gran número de sucesos de este tipo. Estas muestras (tanto los ficheros EVD como los de metadatos) se preparan en el CERN. Luego se transfieren a los centros Tier-1 y Tier-2 donde se tiene prevista la ejecución de los trabajos de digitalización, evitando así su transferencia por la red cada vez que eran requeridos. Para saber cómo acceder a estas muestras en cada uno de los centros donde han sido guardadas se prepara con anterioridad un catálogo XML de POOL específico con los PFNs de los ficheros en el sistema local de almacenamiento. Estos centros publican entonces un *tag* de software especial en el sistema de información de LCG que permite averiguar su localización. Se ha adaptado convenientemente a McRunjob para que encuentre este catálogo XML específico del pile-up en una ubicación estándar, y para que sepa cómo acceder a las muestras usando la información contenida en él.

Sin embargo, la necesidad de acceso a las grandes muestras de ficheros con sucesos de pile-up dificulta la ejecución de la digitalización en el Grid de forma totalmente eficiente. Sólo puede llevarse a cabo la digitalización en aquellos sitios donde estas muestras han sido previamente copiadas, y el número máximo de trabajos que se pueden ejecutar en paralelo viene determinado por el número máximo de accesos simultáneos que el sistema de almacenamiento local puede soportar sin dar problemas de lectura. Claramente, éste es el cuello de botella en la ejecución de la cadena Monte Carlo en LCG. Para aliviar esta situación se ha implementado la posibilidad de que los trabajos en ejecución descarguen un subconjunto aleatorio de las grandes muestras de sucesos PU previamente guardadas en varios SEs. Con esta opción, el metadata attachment para el PU se lleva a cabo sólo sobre los subconjuntos de sucesos seleccionados y descargados. El número de trabajos debe conseguir un compromiso entre el tamaño de la muestra seleccionada, y que debe ser transferida por la red, y el número mínimo de sucesos PU necesario para garantizar fidelidad en la física simulada. Esta opción permite el uso de un mayor número de recursos computacionales para la ejecución de la digitalización con PU, aunque no ha llegado a usarse a gran escala.

Almacenamiento de los datos

Otra de las principales causas de la baja eficiencia del sistema durante las primeras operaciones de producción con McRunjob en LCG fue la alta tasa de fallos en las operaciones de input y output. Estos fallos suelen deberse a interrupciones temporales en los servicios de los Storage Elements o del catálogo RLS. El riesgo de fallo aumenta, por tanto, con el número de operaciones de escritura/lectura. Inicialmente se llevaba a cabo una operación de escritura, y el correspondiente registro en el catálogo, para cada uno de los ficheros EVD producidos.

Para evitar la multiplicidad de operaciones de input/output se ha empaquetado todo el output que generan los programas (los ficheros EVD, el catálogo de POOL para estos ficheros EVD, el summary file y los ficheros de output y error de la aplicación), en un paquete ZIP[2] sin comprimir. Esto ha sido posible porque los paquetes de software de CMS pueden acceder directamente a los datos guardados en ficheros ZIP sin necesidad de desempaquetarlos. De esta forma se reduce el número total de ficheros a guardar en el SE, disminuyendo la complejidad en las operaciones de transferencia y gestión de ficheros en las operaciones de producción. Por otra parte, algunos de los ficheros individuales suelen ser de pequeño tamaño, lo que dificulta su gestión por parte de los sistemas de almacenamiento y de transferencia, optimizados para el manejo de ficheros de gran tamaño. Este problema queda automáticamente resuelto con el uso de un único fichero de gran tamaño con todo el output. El uso de ficheros en formato ZIP impone algunas modificaciones en la herramienta de publicación de datos de CMS (CMSGLIDE) [107], para que se puedan crear los catálogos XML[3] de POOL y los ficheros de metadatos a partir de los

[2] zip es un formato de almacenamiento muy utilizado para la compresión de datos, en el que cada archivo es almacenado de forma independiente, bien sin comprimir, bien utilizando una amplia variedad de algoritmos de compresión.

[3] *eXtensible Markup Language*, es un metalenguaje extensible de etiquetas que permite definir la gramática de lenguajes específicos, muy útil para el intercambio de información estructurada entre diferentes plataformas, bases de datos, editores

ficheros empaquetados, y la instrumentación de McRunjob para que los desempaquete antes de comenzar la ejecución de las siguientes etapas de la simulación. Este fichero XML de POOL creado por CMSGLIDE contiene los nombres lógicos y físicos de los ficheros.

Las ineficiencias en las operaciones de escritura del output eran una causa habitual de fallos. Los trabajos fallaban, tras finalizar la simulación de los sucesos, por no poder guardar los resultados generados debido a problemas temporales en el SE. Para reducir la tasa de fallos por esta causa se ha conferido robustez al proceso de almacenamiento de los datos de output. Se ha implementado el uso de una lista con varios SEs. Si el SE de referencia está temporalmente fuera de servicio se busca otro alternativo en la lista para copiar en él los ficheros. Además, este ciclo de búsqueda de un SE alternativo se puede repetir varias veces, con un cierto retraso entre intentos. De esta forma, si el problema se debe a alguna interrupción temporal del servicio de catálogos RLS, se le concede un cierto margen de tiempo que permite su recuperación antes de declarar la operación como fallida definitivamente. El tiempo entre intentos y el número de ciclos son parámetros configurables, y se han ido ajustando a medida que se ganaba en experiencia.

Acoplamiento con el sistema de tranferencia de datos

Los ficheros de datos simulados que han sido guardados en los SEs han de hacerse visibles para el sistema de transferencia de datos de CMS (PhEDEx) para que éste se encargue de su transferencia de forma fiable y eficiente. Durante la primera fase, el sistema de producción de Monte Carlo y PhEDEx estaban desacoplados, y las transferencias se hacían manualmente usando las herramientas básicas proporcionadas por el Grid. Como se comentó en el capítulo 2, estas herramientas eran aún bastante rudimentarias, poco fiables y no tolerantes a fallos. Los movimientos de datos se hacían mediante ficheros individuales, y la comprobación de que las transferencias se completaban correctamente era una operación manual. En el caso de la producción en LCG la fiabilidad y rapidez de las transferencias es aún más importante que para la producción local tradicional, pues los datos suelen quedar dispersos en gran cantidad de sitios. El acoplamiento de los sistemas de producción y de transferencia de datos ha sido una de las grandes mejoras introducidas en el sistema de producción de Monte Carlo.

El proceso por el cual se hacen conocidos los ficheros por parte de PhEDEx recibe el nombre de *data injection*. En este proceso, los atributos de los ficheros se insertan en la base de datos central de PhEDEx (TMDB). Los PFNs están disponibles únicamente en el catálogo local de PhEDEx accesible en cada sitio, y sólo son accesibles en el momento de ejecutarse las transferencias. En el caso de LCG, este catálogo es el catálogo central de LCG.

PhEDEx modela los centros como nodos, interconectados de acuerdo a una cierta topología de transferencia. En la anterior producción en granjas locales todos los ficheros generados quedaban guardados en el sitio donde se ejecutaban los trabajos de simulación. Sin embargo, en la producción en LCG estos ficheros de output pueden quedar ampliamente distribuidos entre una gran cantidad de sitios. Para manejar esta situación con eficacia, PhEDEx modela todos los centros del LCG con datos como un único *nodo virtual*. De esta forma, los ficheros de producción se inyectan en este nodo virtual y están disponibles para su manejo, en particular para su recolección y consiguiente transferencia a otros nodos reales de PhEDEx.

Los ficheros se inyectan en PhEDEx de forma atómica, a medida que se van generando, sin necesidad de esperar a que se haya procesado y publicado la colección completa, de modo que quedan inmediatamente disponibles para su transferencia.

El acoplamiento entre los sistemas de producción y PhEDEx se consigue a través del summary file, que contiene toda la información necesaria para la inyección de los ficheros en PhEDEx. El operador de producción recupera este summary file a través del Input Sandbox, permitiendo a los agentes de inyección extraer de él la información necesaria y guardarla en la base de datos de PhEDEx.

de texto, hojas de cálculo, etc.

Tras la implementación del empaquetamiento del output de los trabajos de simulación, son estos ficheros ZIP los que se registran en el catálogo y se inyectan en PhEDEx.

Instalación local del software

Otra mejora introducida en el sistema de producción ha sido posibilitar la ejecución de los trabajos en todos los nodos del Grid, aunque no tengan instalado el software del experimento. Si no está disponible, los trabajos descargan e instalan el software (previamente guardado en un SE) antes de comenzar su ejecución. El tiempo adicional que introduce esta operación es relativamente corto en comparación con el tiempo total de ejecución del trabajo, pero permite el uso de gran cantidad de recursos, incluyendo aquellos que tradicionalmente dan poco o ningún soporte al experimento CMS.

Catálogo de ficheros dedicado

Se ha implementado un catálogo de ficheros dedicado para las operaciones de producción, el *LCG File Catalogue* (LFC) [108], que ha sustituido a RLS. Esto ha ayudado a reducir la tasa de fallos en las consultas para averiguar la localización de los ficheros de input o registrar los de output. Las primeras producciones Monte Carlo en LCG pueden dividirse en dos etapas claramente diferenciadas, en las que el catálogo es RLS o LFC, respectivamente, y se puede constatar un aumento considerable en la eficiencia del sistema de producción durante la segunda fase.

Código de errores

Inicialmente, cuando un trabajo fallaba, era difícil averiguar la causa de error, haciendo imprescindible la inspección visual de los enormes ficheros de salida producidos, introduciendo grandes retrasos en el reenvío de los trabajos. Se ha conseguido crear un código de errores, bastante amplio y preciso, que facilita la identificación rápida y correcta de las causas de fallo y permite reaccionar de forma más eficaz.

4.1.4. Experiencia

Tras su adaptación para trabajar en un entorno distribuido y heterogéneo como es el Grid, se ha usado el sistema de producción para llevar a cabo una producción masiva de datos Monte Carlo [109, 110, 111]. La eficiencia de la producción depende significativamente de la estabilidad y fiabilidad de los servicios Grid (conectividades, servicios centrales de bases de datos, etc.), y de todos los centros donde se ejecutan los procesos. A medida que se ganaba en experiencia se han ido descartando los centros con mayor tasa de fallos, y se ha establecido una clara estrategia de listas blancas: sólo un número limitado de centros, pero fiables, se han usado de forma masiva.

La figura 4.4 muestra el número de procesos diarios ejecutados en el Grid durante un período de 500 días, y el número acumulado de sucesos simulados, digitalizados y reconstruidos en el mismo período. En total, en un año de operaciones, se han procesado en LCG más de 16 millones de sucesos simulados, 11.5 millones de sucesos digitalizados y 6.5 millones de sucesos reconstruidos. Se han generado unos 15 TB de datos. En media, unos pocos cientos de trabajos han sido procesados cada día, con algunos picos de más de 1000 trabajos. Se puede ver cómo el rendimiento fue bajo en los comienzos, pero ha mejorado rápidamente gracias a todas las mejoras introducidas en el sistema de producción, descritas antes. Se ha consesguido así un ritmo medio de unos 110 trabajos en ejecución por día. Varias razones explican la falta de continuidad en el ritmo de producción, con periodos de baja productividad: ausencia de peticiones de simulación de sucesos por parte de los grupos de física, demoras por el cambio en la versión del software del experimento, falta de automatismo en algunas operaciones del sistema de producción, y operaciones de mantenimiento en los servicios Grid.

En la última etapa se alcanzó un valor cercano a los 1500 trabajos simultáneos en LCG, lo que se puede considerar, sin ningún tipo de dudas, como una gran éxito, dados los escasos recursos aún disponibles. En esta fase final de producción se logró completar con éxito la simulación de 2.4 millones de sucesos en

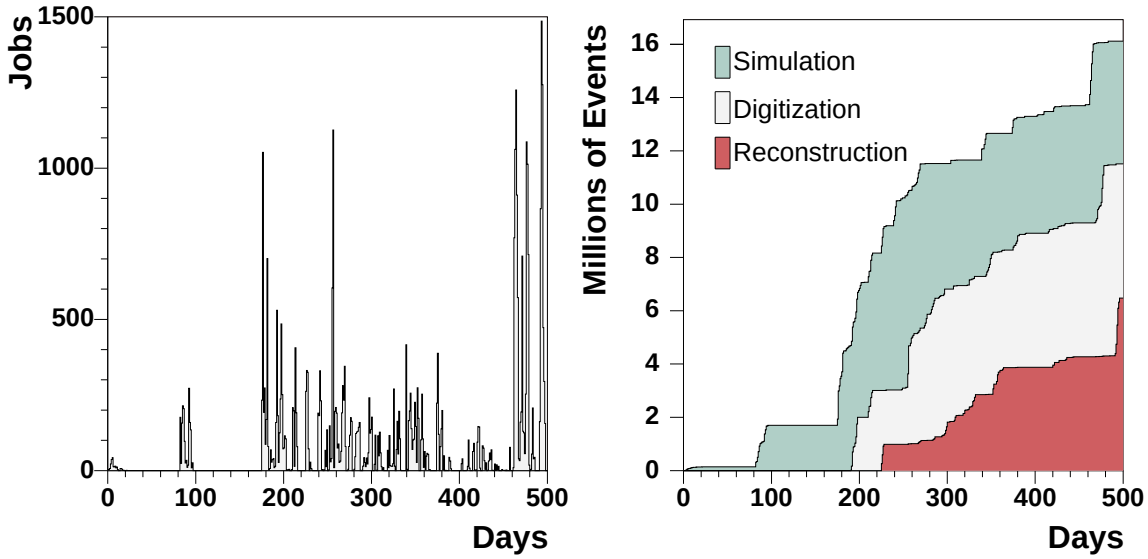


Figura 4.4: Número de trabajos ejecutados cada día (superior izquierda) y número de sucesos simulados, digitalizados y reconstruidos (superior derecha) durante un período de 500 días.

menos de una semana. La digitalización y reconstrucción de estos sucesos también se llevó a cabo en un plazo muy corto de tiempo.

En la figura 4.5 se muestra la distribución de laboratorios donde se han ejecutado los procesos. Se puede ver cómo se ha seguido una estrategia clara basada en la idea de usar pocos sitios, pero muy fiables y robustos. Un porcentaje bastante elevado de los procesos han sido ejecutados en los dos centros españoles (CIEMAT y PIC). Al tener acceso a los recursos de ambos centros (bien directamente o a través de los administradores locales), han sido los candidatos ideales para la ejecución de la producción masiva. En general, aparte de la cantidad de recursos que aporta y su estabilidad, uno de los factores clave para decidir si se usa o no un centro Grid es la buena predisposición para colaborar y la rapidez de reacción de los administradores locales de ese centro. Éste es el caso de los centros alemanes y británicos, principalmente, cuyo personal se ha mostrado dispuesto a colaborar en todo momento. El número de centros que puede usarse es aún menor en el caso de la digitalización, pues sólo se puede ejecutar en aquellos sitios donde se han instalado previamente las muestras de pile-up. Debe disponer, además, de un sistema potente de acceso local a datos que permita la lectura simultánea de estas muestras por parte de decenas de trabajos de forma eficiente. Pocos centros han sido capaces de satisfacer estos requisitos. 42 laboratorios repartidos en 14 países han contribuido a la producción, aunque casi el 90 % se ha realizado en sólo 13 laboratorios de 5 países diferentes. Sólo se han usado de forma masiva aquellos centros con un alto rendimiento continuo y una baja tasa de fallos.

Gracias a las herramientas de monitorización integradas en el sistema de producción, y al completo y preciso código de errores implementado, se puede hacer un estudio detallado de las causas de fallo de los trabajos. La figura 4.6 muestra, diferenciando por etapas, el número de intentos necesarios para conseguir que cada trabajo finalice con éxito, lo que se puede expresar en términos de eficiencia.

Durante la fase de implementación inicial el sistema aún no era lo bastante robusto. Esto, junto al hecho de que el sistema de catálogos que usaba CMS no era aún lo bastante fiable, introdujo una gran cantidad de fallos. En la segunda etapa, tras las mejoras añadidas, el sistema ha mejorado sustancialmente. Se ve claramente cómo se ha reducido, de forma muy significativa, el número de intentos por trabajo, lo que se traduce en un aumento considerable de la eficiencia. En todos los pasos (simulación, digitalización y

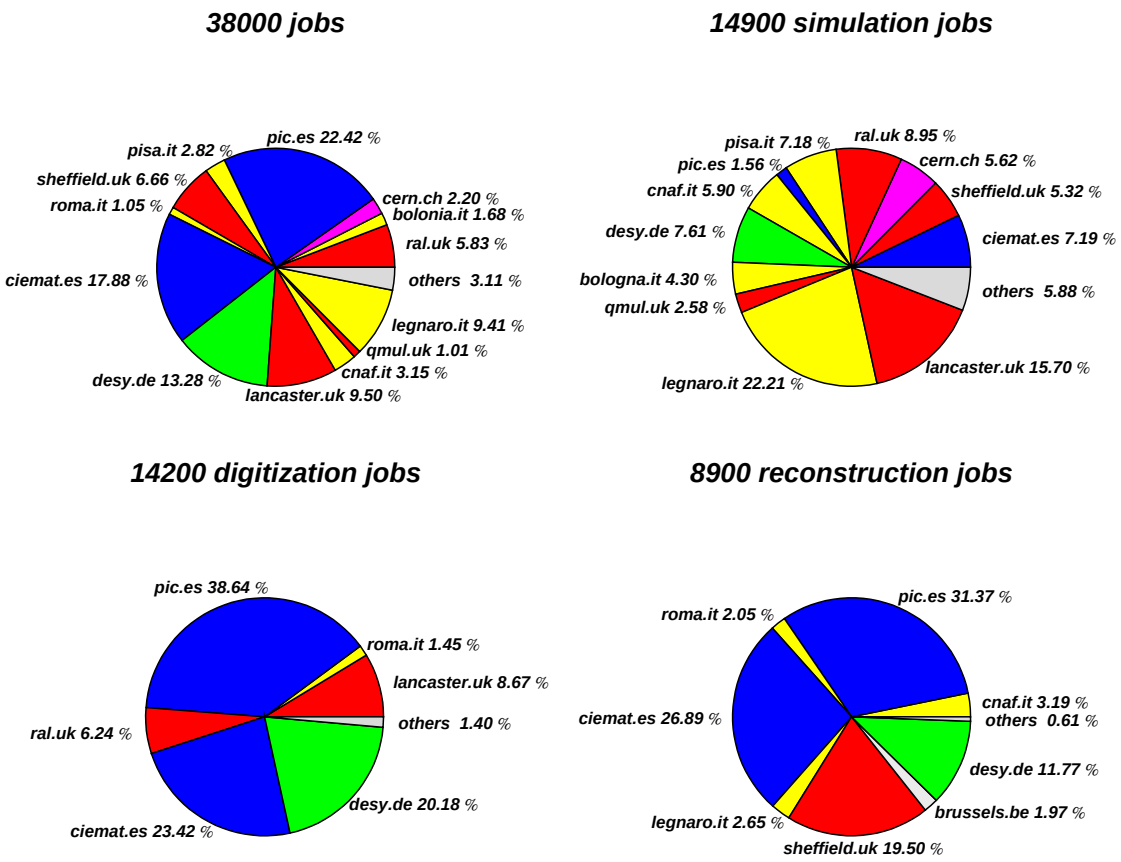


Figura 4.5: Distribución por sitios de los trabajos ejecutados: total (superior izquierda), simulación (superior derecha), digitalización (inferior izquierda), reconstrucción (inferior derecha). Los centros del mismo pais aparecen con el mismo color.

reconstrucción) la eficiencia ha aumentado significativamente, pasando de un valor global del 66 % durante la fase de implementación al 86 % en la etapa final. En la tabla 4.1 se compilan los porcentajes de trabajos fallidos durante las dos etapas, así como los valores globales. En general, la frecuencia de ocurrencia de errores en casi todos los capítulos listados en la tabla disminuye a medida que se introducen mejoras en el sistema, especialmente los relacionados con el manejo de datos (lectura y escritura, remota y localmente).

En la figura 4.7 se puede encontrar un ejemplo de procesado de todos las etapas para un Dataset completo. Muestra el número de trabajos en ejecución en función del tiempo para las fases de simulación, digitalización y reconstrucción. En verde se muestran los trabajos que finalmente acabaron con éxito, mientras que en rojo se muestran aquellos que fallaron. Los periodos de inactividad entre etapas se deben a la falta de versiones actualizadas del software.

La figura 4.8 muestra la distribución de tiempos (izquierda) y tamaño (derecha) por suceso procesado para las diferentes etapas de la producción. En media, son necesarios unos pocos minutos para simular un suceso mientras que la digitalización y la reconstrucción tardan generalmente menos de un minuto por suceso. Los tiempos de procesamiento más cortos corresponden a la digitalización sin pile-up. Los outputs de mayor tamaño son los resultados de la etapa de simulación. La tabla 4.2 recoge los números más relevantes para los tiempos de procesado y tamaños de outputs, junto con el número total de sucesos procesados.

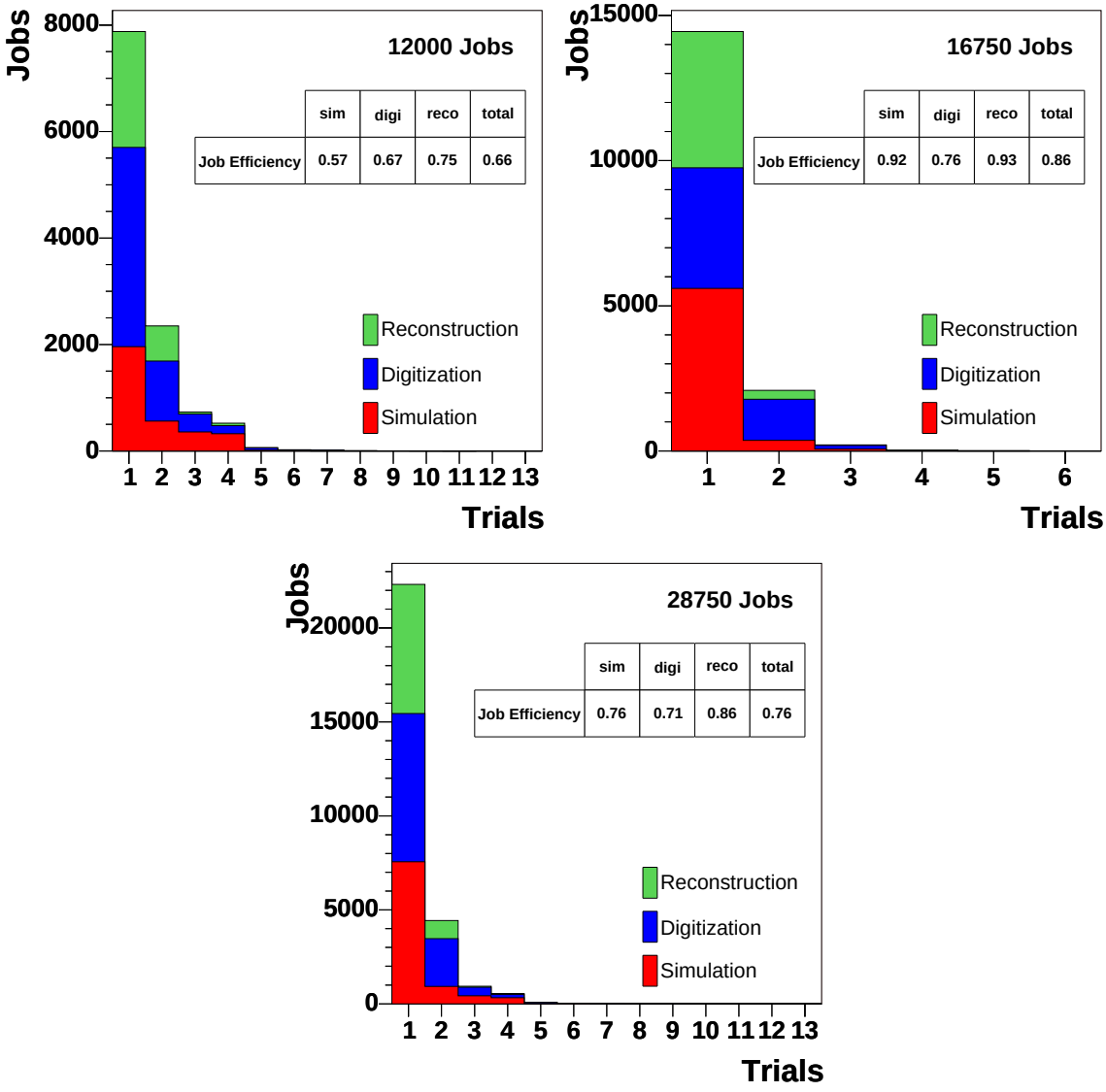


Figura 4.6: Distribución del número de intentos para ejecutar satisfactoriamente cada trabajo en las dos fases de la producción (superior), y valores globales (inferior).

Como los trabajos que ejecutan digitalización con pile-up necesitan leer del sistema de almacenamiento un número bastante alto de sucesos de *minimum bias* para cada suceso de señal, se introduce una carga adicional en el sistema de almacenamiento de algunos centros del LCG. Esto se traduce en que el tiempo total de procesamiento de los trabajos viene dominado por estas operaciones de lectura. Este efecto se puede observar en la figura 4.9 (izquierda), donde se muestra la distribución del cociente entre tiempo total de procesamiento de los trabajos y el tiempo de CPU, para las diferentes etapas de la producción. Como se esperaba, el cociente es mayor para la digitalización con pile-up, donde la CPU está parada durante una gran fracción del tiempo de procesamiento de los trabajos mientras se leen los datos del sistema de almacenamiento. Este efecto puede ser más o menos acusado, dependiendo del sistema de almacenamiento particular, como se aprecia en la figura 4.9 (derecha). En la figura 4.10 se puede ver cómo el rendimiento varía considerablemente en función del valor asignado a uno de los parámetros de

Causa de Fallo	**Fase 1**	**Fase 2**	**Global**
Lectura de los datos de entrada	4.8 %	0.5 %	2.3 %
Escritura de los datos de salida	7.4 %	0.3 %	3.3 %
Acceso al catálogo de LCG	2.8 %	1.5 %	2.0 %
Configuración del software del experimento	10.1 %	0.2 %	4.3 %
Acceso a los datos locales	8.3 %	2.1 %	4.7 %
Fallos en el programa de producción MC	<0.1 %	5.5 %	3.2 %
Errores en la configuración del sitio Grid	0.3 %	0.4 %	0.4 %
Sin clasificar	0.3 %	3.5 %	2.2 %
TOTAL	34.1 %	14.0 %	22.4 %

Tabla 4.1: Evolución de los porcentajes de fallo durante las dos etapas de la producción. Durante la etapa de implementación el porcentaje total de fallos fue del 34 %. Tras las mejoras éste valor se redujo al 14 %.

configuración de los servicios de dCache que gestionan el SE de DESY [112].

	Simulación	Digitalización	Reconstrucción
Trabajos procesados	15000	14200	9000
Sucesos procesados	16 M	11.5 M	6.5 M
Sucesos procesados cada día	44500	37000	23500
Tiempo medio por trabajo	28 h	13 h	8 h
Tiempo medio por suceso	3 min	1.5 min	1 min
Tamaño medio por trabajo	360 MB	420 MB	250 MB
Tamaño medio por suceso	500 kB	570 kB	300 kB

Tabla 4.2: Valores aproximados de las cantidades más relevantes alcanzadas durante la producción MC en LCG.

La figura 4.11 muestra la distribución de los tiempos de espera de los trabajos antes de comenzar su ejecución. El pico en 100 segundos corresponde a la latencia mínima que introduce el Grid LCG desde que se envía un trabajo hasta que el sistema detecta que está en ejecución. Los pasos intermedios son el envío del trabajo al RB, desde éste a un CE, el encolamiento del trabajo en el sistema de colas local y el informe del estado del trabajo desde el CE al RB. El escaso número de trabajos en la cola de la distribución demuestra que se ha hecho siempre un gran esfuerzo para localizar centros con recursos disponibles y enviar los trabajos a estos centros. Además, el sistema aún no estaba bastante desarrollado y carecía, por tanto, de un mecanismo automático de envío de trabajos. Esta carencia no ha permitido usar siempre la totalidad de los recursos disponibles.

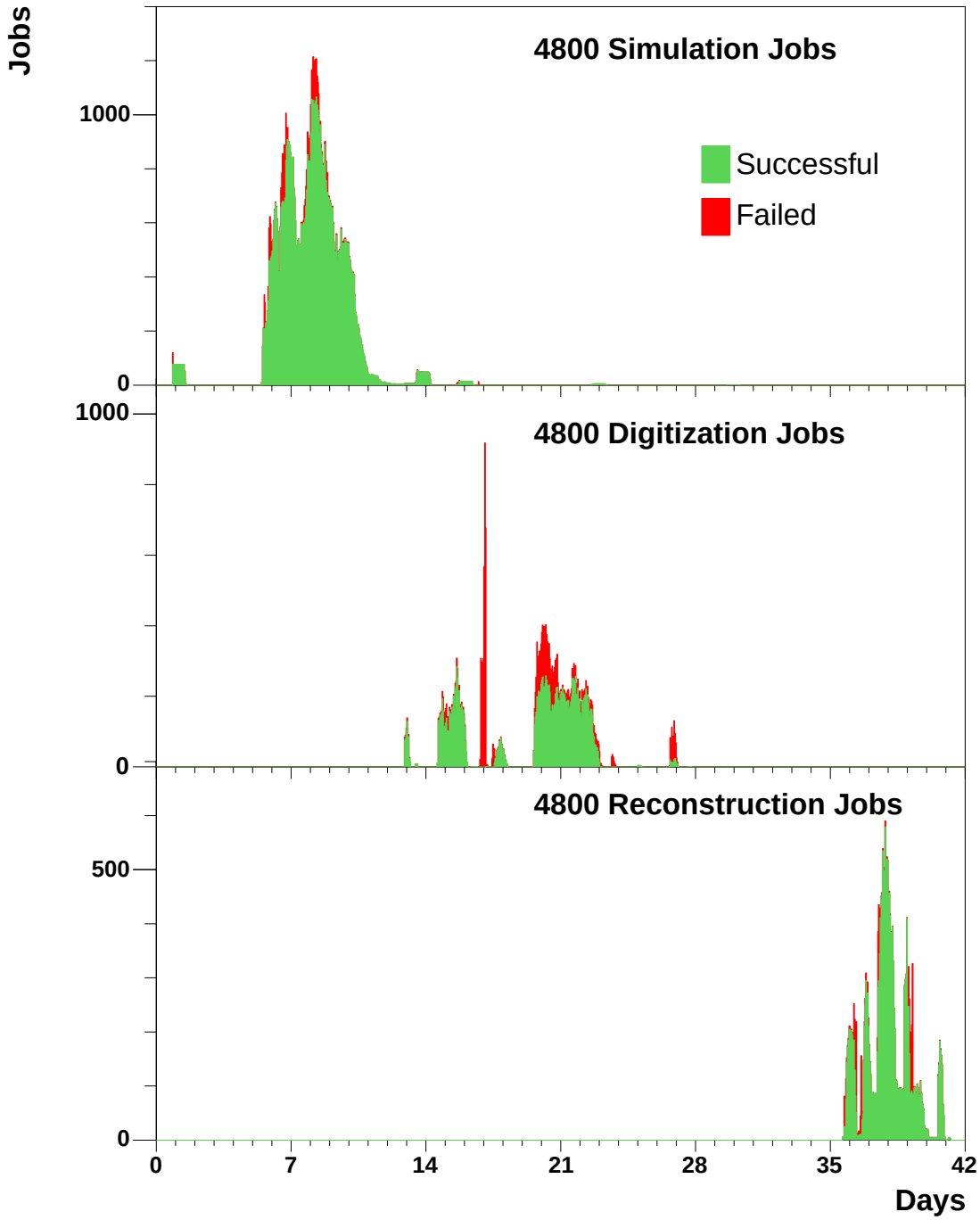


Figura 4.7: Número de trabajos en ejecución en función del tiempo para la simulación, digitalización y reconstrucción correspondientes un Dataset completo. Cada trabajo contribuye en todos aquellos bins que cubren el tiempo completo en que estuvo en ejecución. En verde se representan los trabajos que finalmente acabaron con éxito, mientras que en rojo se marcan los que acabaron fallando en algún momento de su ejecución.

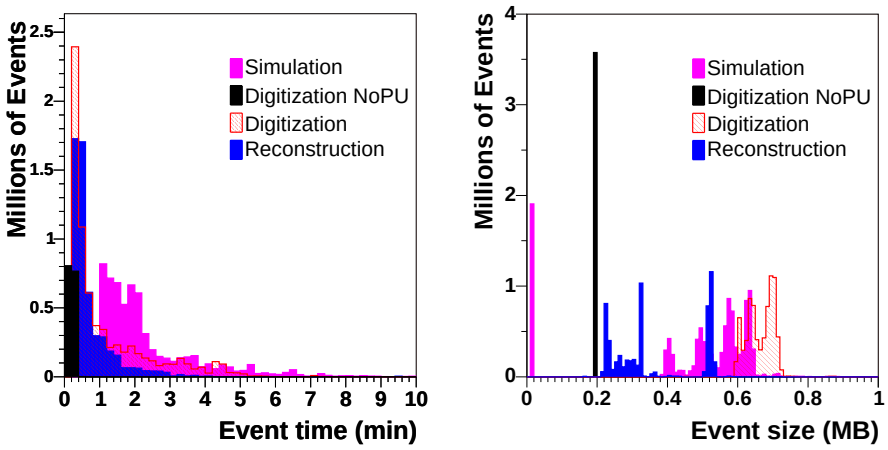


Figura 4.8: Distribución del tiempo de procesamiento por suceso (izquierda) y del tamaño del output obtenido (derecha), para todas las fases de la simulación Monte Carlo.

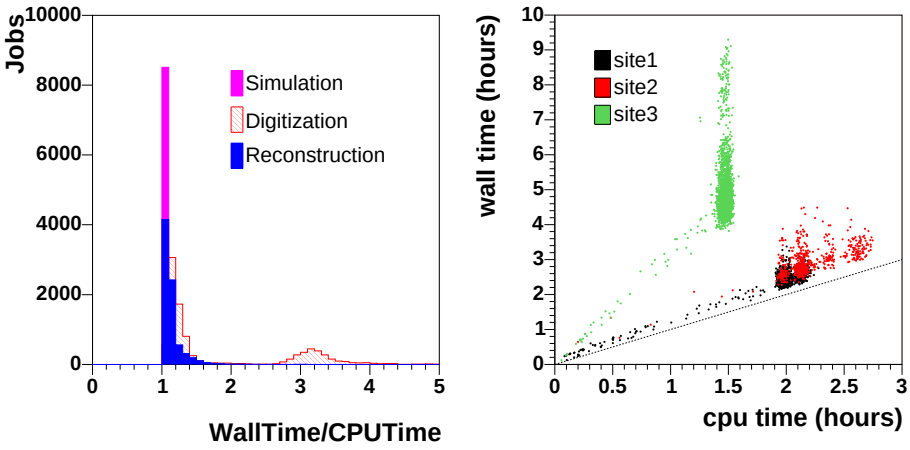


Figura 4.9: Relación entre el tiempo total de procesamiento y el tiempo real de CPU, para las diferentes etapas de la simulación Monte Carlo (izquierda) y en diferentes centros (derecha).

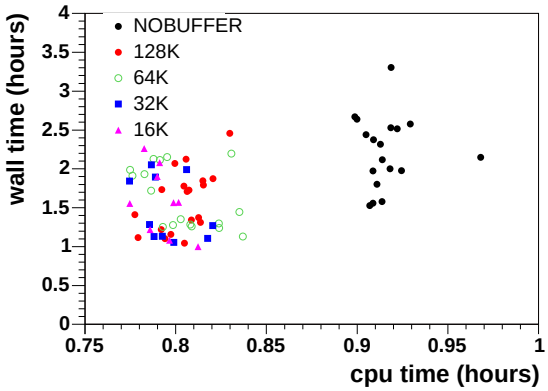


Figura 4.10: Influencia de las variables de configuración de dCache en el rendimiento de los trabajos de digitalización procesados en DESY.

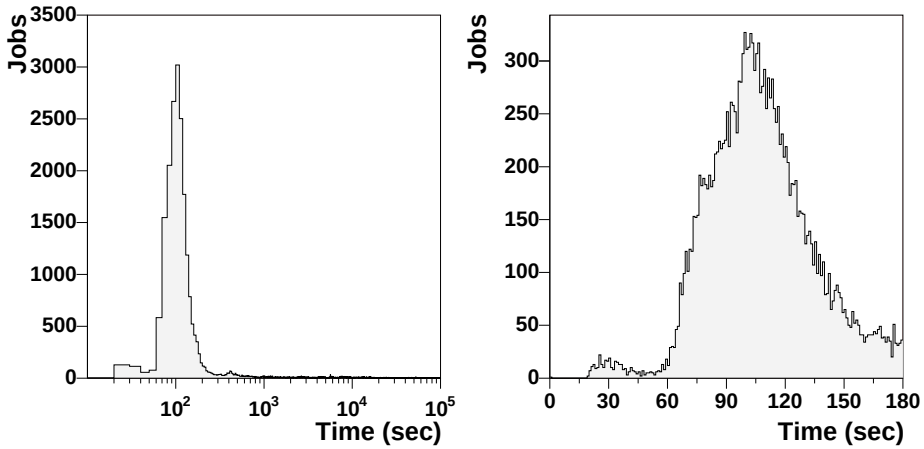


Figura 4.11: Distribución de los tiempos de espera de los trabajos antes de comenzar su ejecución. La figura de la derecha es un zoom de la distribución.

4.2. Nuevo sistema de producción: ProdAgent

El sistema de producción Monte Carlo cambió en 2006. McRunjob fue reemplazado por ProdAgent. El nuevo sistema ha sido diseñado con el objetivo de conseguir un mayor nivel de automatización, facilidad de mantenimiento, escalabilidad, evitar *single points of failure*[4], gestión más apropiada de los errores para hacer un uso más eficiente de los recursos disponibles, y la posibilidad de operar en múltiples entornos Grid. El objetivo final es alcanzar una escala de producción mayor, pasando de unos pocos cientos de CPUs a varios miles para cada instancia de producción. Además, ProdAgent ha integrado el sistema de producción con el nuevo modelo de datos de los sucesos -*Event Data Model* (EDM)-, sistema de gestión de datos y entorno de procesamiento de datos de CMS. RefDB ha sido reemplazada por DBS, PubDB por DLS, y los catálogos XML de POOL por un catálogo local trivial (TFC) en cada centro. Finalmente, el nuevo EDM incorpora conceptos como el de file block, que permite la gestión y análisis simultáneos de conjuntos de ficheros.

McRunjob no estaba suficientemente preparado para conseguir un alto grado de automatización de las tareas de producción MC en LCG. No disponía de unas herramientas de monitorización y de gestión de errores adecuadas que permitiesen el procesamiento de trabajos en un entorno altamente distribuido y con un cierto grado de inestabilidad e ineficiencia. Como consecuencia, el sistema de producción no era lo bastante robusto y eficiente, y presentaba un límite de escalabilidad que prácticamente imposibilitaba la gestión de más de mil trabajos simultáneos, necesitando dedicación exclusiva por parte de los operadores. Se hizo necesario el diseño de un nuevo sistema de producción que permitiese la gestión automatizada de las tareas de preparación, envío, seguimiento y reenvío en caso de fallo de los trabajos de producción, y el registro de los datos en las diferentes bases de datos y sistemas de gestión de datos (sistemas de localización y transferencias, de bookkeeping, etc.) Por otra parte, el nuevo EDM y el nuevo entorno de procesamiento de CMS permiten simplificar el flujo de los trabajos de procesamiento y la publicación de los datos producidos, bastante complejos de gestionar con McRunjob. En particular, el no poder ejecutar las distintas etapas de la cadena de simulación Monte Carlo de una sola vez (necesitando un trabajo distinto para cada una de ellas) y la generación de los ficheros de metadatos eran dos motivos de retraso. En el desarrollo del nuevo sistema se ha tenido en cuenta, por tanto, toda la experiencia adquirida durante la fase de operaciones con McRunjob en LCG.

4.2.1. Arquitectura del sistema

Para conseguir los objetivos que se persiguen en el nuevo sistema de producción (especialmente la escalabilidad, facilidad de mantenimiento, y flexibilidad), éste se ha desarrollado para que sea lo más modular posible. Así, el sistema está compuesto por tres grandes módulos: ProdRequest, ProdManager (o ProdMgr) y ProdAgent. ProdRequest recoge las peticiones de los usuarios y grupos de física, ProdMgr se encarga de gestionar estas peticiones y, por último, las instancias de ProdAgent las ejecutan. La figura 4.12 muestra esta relación entre los distintos módulos del sistema.

El primer módulo es ProdRequest. Es una aplicación de front-end para los usuarios y los administradores, con una interfaz web, a través del cual los grupos de física solicitan los sucesos que desean sean simulados. Estas solicitudes reciben comúnmente el nombre de *Requests*. Las condiciones de procesamiento de un conjunto de sucesos solicitados (como el Dataset de input y de output, versión del software, parámetros de configuración, etapa dentro de la cadena de simulación, etc.) reciben el nombre de *Workflow*. ProdRequest se puede usar como una interfaz local que inyecta workflows directamente a una instancia local ProdAgent, o como un servidor al que ProdMgr accede para organizar las peticiones de acuerdo a las políticas y prioridades establecidas. Una vez que los administradores han aprobado los requests, ProdRequest se encarga de crearlos, gestionarlos y monitorizarlos. Para llevar a cabo estas tareas ProdRequest crea un fichero con las especificaciones del workflow. Se trata de un fichero XML que define las tareas de procesamiento, contiene los ficheros de configuración, los detalles del Dataset, etc. Una vez creado el fichero de especificación, ProdRequest pasa toda esta información a la componente ProdMgr.

[4]SPOF, expresión en inglés de uso generalizado para hacer referencia a aquellos componentes de un sistema cuyo fallo puede provocar el bloqueo del sistema completo.

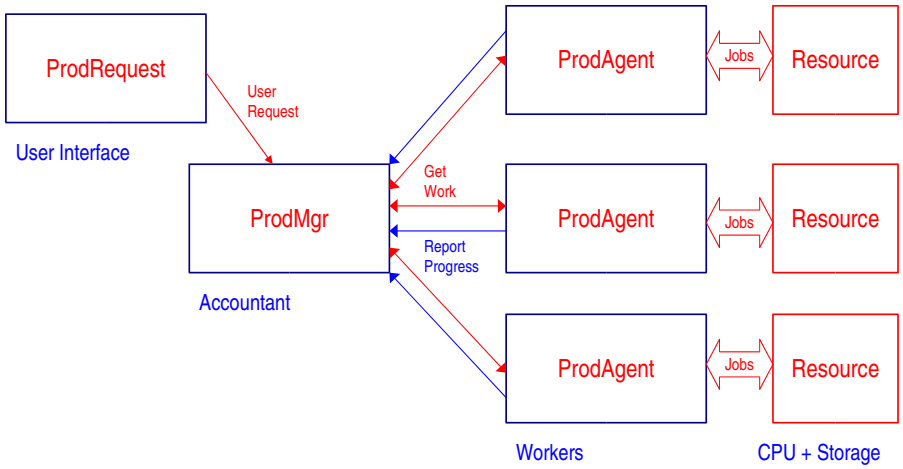


Figura 4.12: Arquitectura con los distintos módulos del sistema de producción Monte Carlo: ProdRequest, ProdManager, y las múltiples instancias particulares de ProdAgent.

Una vez que ProdMgr recibe de ProdRequest la información sobre los requests solicitados, se encarga de gestionar y contabilizar las peticiones. Para cada request, divide el número total de sucesos en bloques, o asignaciones, que se procesan individualmente. Cuando una instancia de ProdAgent solicita un trabajo, ProdMgr es responsable de proporcionarle toda la información necesaria para crear ese trabajo. ProdMgr también proporciona políticas automáticas, basadas en la evaluación del estado de la petición y comparando con un cierto umbral los valores de determinadas métricas. ProdMgr también se encarga de monitorizar cada asignación, y de tener una imagen del estado del request completo. En la figura 4.13 se puede ver la interacción entre el módulo de ProdMgr y una instancia particular de ProdAgent.

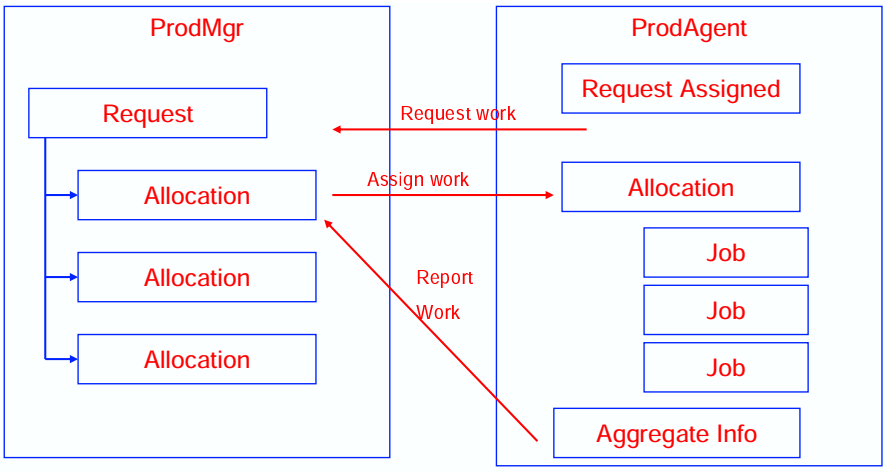


Figura 4.13: Ciclo de vida de una solicitud de producción Monte Carlo gestionada con ProdAgent y ProdManager.

Cuando una asignación de un workflow llega a una de las instancias de ProdAgent, el operador puede crear distintos trabajos para esa asignación (dependiendo del número de sucesos que desee simular en cada trabajo). La instancia de ProdAgent se encarga entonces de enviarlos a algún centro del Grid para su procesamiento. Cuando un trabajo finaliza y ha generado los sucesos pedidos devuelve un informe a la instancia de ProdAgent, copia los datos producidos en el SE local, e inserta el registro correspondiente en la instancia local DBS/DLS. Este informe incluye información sobre los ficheros de input y output,

sobre el centro donde se ejecutó el trabajo, errores, y algunos datos útiles para diagnóstico. Cuando hay suficientes datos producidos del mismo tipo se genera y envía un trabajo especial que concatena los ficheros producidos y crea un número reducido de ficheros de un tamaño adecuado (de 2 a 4 GB) para un almacenamiento y transferencia eficientes. El objetivo es tener al final un número reducido de ficheros. Esta operación es conocida como *merge*. Tras las operaciones de merge se procede a la "migración" de la información de las bases de datos locales a la instancia global de DBS/DLS, y se invoca al sistema de transferencia de datos para guardar los ficheros creados en un centro Tier-1 En la figura 4.14 se muestra un esquema que describe este proceso.

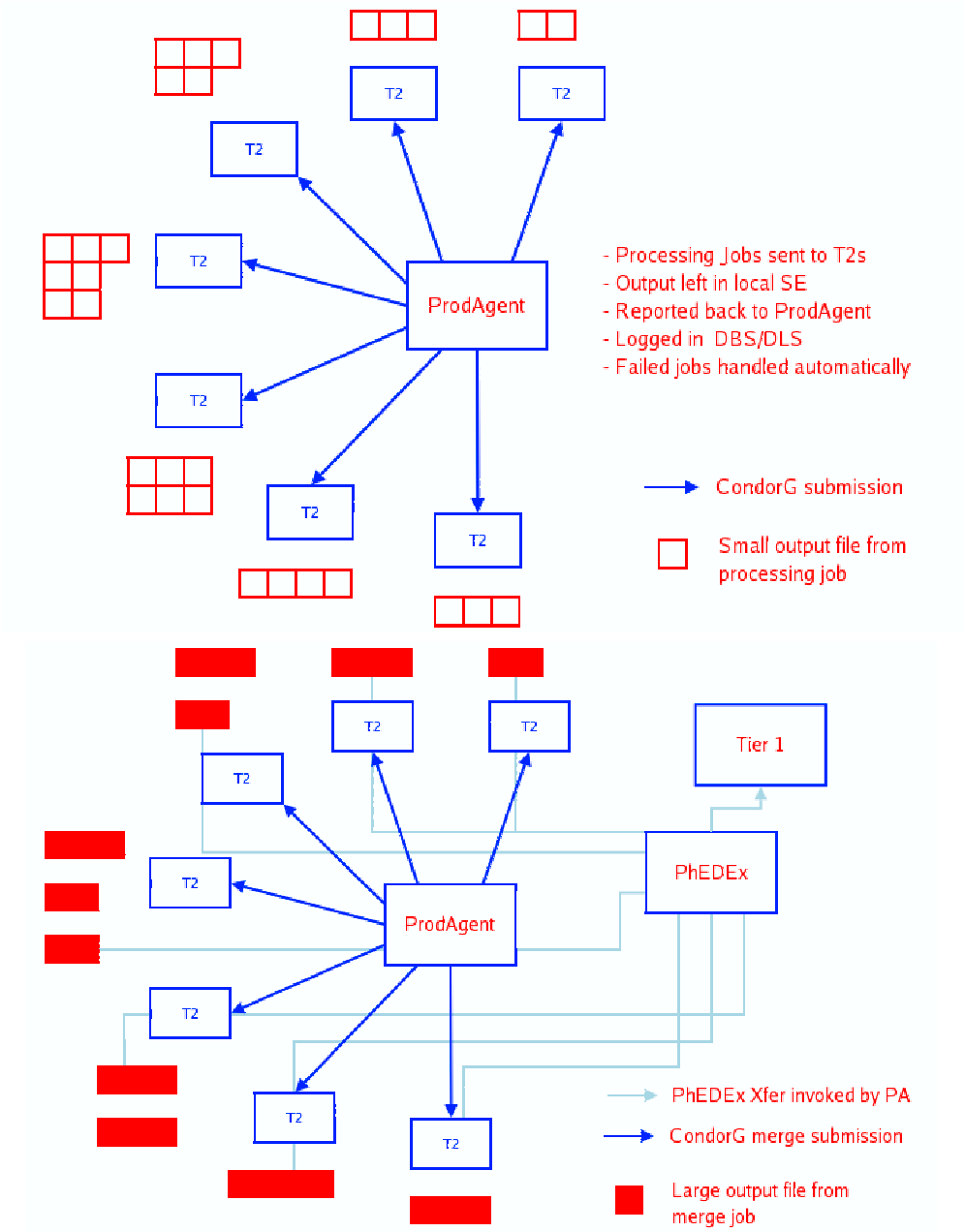


Figura 4.14: Envio de varios trabajos de producción por parte de una instancia de ProdAgent (arriba), y recolección y transferencia a un Tier-1 de todos los outputs generados mediante PhEDEx (abajo).

La escalabilidad del sistema se consigue operando varias instancias PA simultáneamente. Cada una de estas instancias usa una base de datos local para operaciones y monitorización de sus componentes, y una instancia local DBS/DLS para el registro de los datos. Los datos producidos se publican en la base de datos del sistema de transferencia de datos y en la instancia global DBS/DLS para que estén disponibles para ser transferidos y para su análisis.

El nuevo entorno de procesamiento, CMSSW, está centrado en el concepto de Suceso, o *Event*. Un trabajo de procesado de datos se compone de una serie de algoritmos que se ejecutan en un determinado orden. Los algoritmos sólo se comunican a través de los datos almacenados en el Event. Sólo existe un ejecutable y varios módulos plug-in que ejecutan los algoritmos. El mismo ejecutable se usa para el procesamiento de HLT, simulación, reconstrucción y análisis. El ejecutable de CMSSW se configura en el momento de la ejecución mediante un fichero de configuración, específico para cada trabajo, que proporciona el usuario. Este fichero de configuración dice al ejecutable qué datos debe usar, qué módulos ejecutar, y en qué orden se ejecutan esos módulos. Los módulos solicitados se cargan dinámicamente al comienzo de la ejecución del trabajo. Este entorno proporciona formas de garantizar la repetibilidad manteniendo y guardando automáticamente suficiente información histórica de todos los resultados de la aplicación. El nuevo modelo de datos y el nuevo entorno ya no hacen uso de metadatos, lo que facilita considerablemente el flujo de trabajos de producción.

Las diferentes componentes de ProdAgent funcionan como procesos separados y se comunican entre sí a través de un módulo de servicio de mensajes. Este servicio de mensajes no es más que una base de datos MySQL [113] donde unas componentes registran su estado y las acciones que han de ejecutarse, y otras componentes leen estas entradas para iniciar su ejecución cuando sea necesario. El operador de ProdAgent puede activar y desactivar estas componentes de forma independiente. Su estado está registrado siempre en la base de datos. La figura 4.15 muestra un esquema de este mecanismo. Con esta implementación, el trabajo se reparte entre estas componentes atómicas que encapsulan funcionalidades específicas.

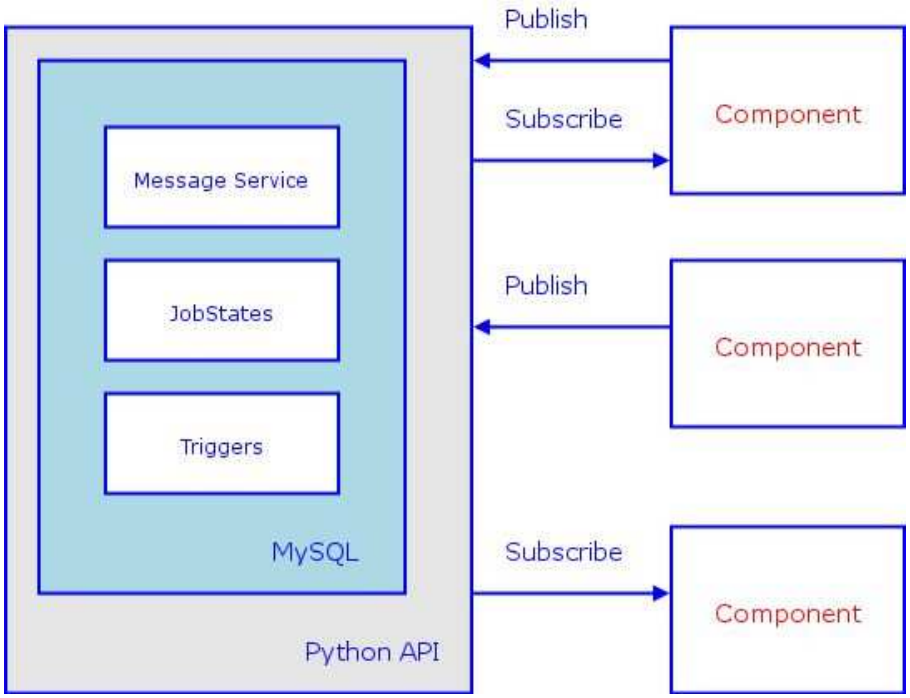


Figura 4.15: Esquema del diseño de ProdAgent. Los componentes independientes se comunican entre sí a través de una base de datos donde registran su estado y a través de cual se intercambian mensajes.

Las componentes principales de ProdAgent, cuya interrelación se puede ver en la figura 4.16, son las siguientes:

- **MergeSensor:** Comprueba el tamaño de los ficheros registrados en DBS para los Datasets seleccionados. Si, en función de su tamaño, existe un número apropiado de ficheros candidatos para el proceso de concatenación esta componente crea el correspondiente trabajo de merge. Se crea un nuevo Dataset donde se insertan los ficheros resultado de la operación merge.

- **MergeAccountant:** Actualiza la base de datos de operaciones de merge basándose en el código de retorno de los trabajos de merge.

- **RequestInjector:** Toma un fichero XML con las especificaciones de un Workflow y obtiene de él las especificaciones para crear nuevos trabajos.

- **JobCreator:** Es la componente responsable de la creación de los trabajos, a partir de las especificaciones proporcionadas por RequestInjector. También crea un espacio de caché para cada trabajo, que es gestionado por ProdAgent.

- **JobSubmitter:** Es responsable de enviar los trabajos a cualquier recurso de ejecución (granja local, Grid, etc.)

- **DBSInterface:** Actualiza DBS con información sobre los nuevos Datasets y los ficheros producidos.

- **StatTracker:** Extrae datos a partir del informe de output que los trabajos generan cuando finalizan, y guarda parte de esta información en la base de datos de ProdAgent. Esta información es útil para obtener resultados estadísticos sobre el rendimiento de los trabajos, por ejemplo.

- **Phedex:** Recopila datos referentes a los trabajos de merge finalizados a partir de estos informes de output e inyecta los ficheros generados en PhEDEx para su migración a su ubicación final.

- **JobQueue:** Actúa como una cola FIFO[5] que retiene las peticiones de creación de trabajos pendientes, previamente registradas en la base de datos de ProdAgent, hasta que hay recursos disponibles para su creación y envío.

- **DatasetInjector:** Juega un papel similar a la componente RequestInjector, pero iterando sobre un Dataset que exista en DBS para generar trabajos que procesen ese Dataset.

- **ProdMgrInterface:** Encapsula las interacciones entre ProdAgent y ProdMgr. Cuando los recursos están disponibles para ProdAgent, éste llama a ProdMgr para obtener la lista de peticiones (y sus prioridades) que le han sido asignadas. Se encarga también de llevar la cuenta de los trabajos que han finalizado con éxito o han fallado.

- **JobCleanup:** Cuando un trabajo ha finalizado borra cierta información registrada en la base de datos de ProdAgent sobre dicho trabajo y libera el espacio de caché que se había reservado para él.

- **ErrorHandler:** Es la componente que gestiona los errores. Dependiendo del tipo de error y del tipo de trabajo (de procesamiento, de merge, ...) determina si el trabajo debe o no reenviarse, y actualiza el estado del trabajo en la base de datos.

- **RssFeeder:** Crea un canal de información que puede usarse para enviar información relevante a los operadores de ProdAgent.

[5] *First In First Out.* Es un tipo de cola donde sus componentes se procesan en el mismo orden en el que entran en ella.

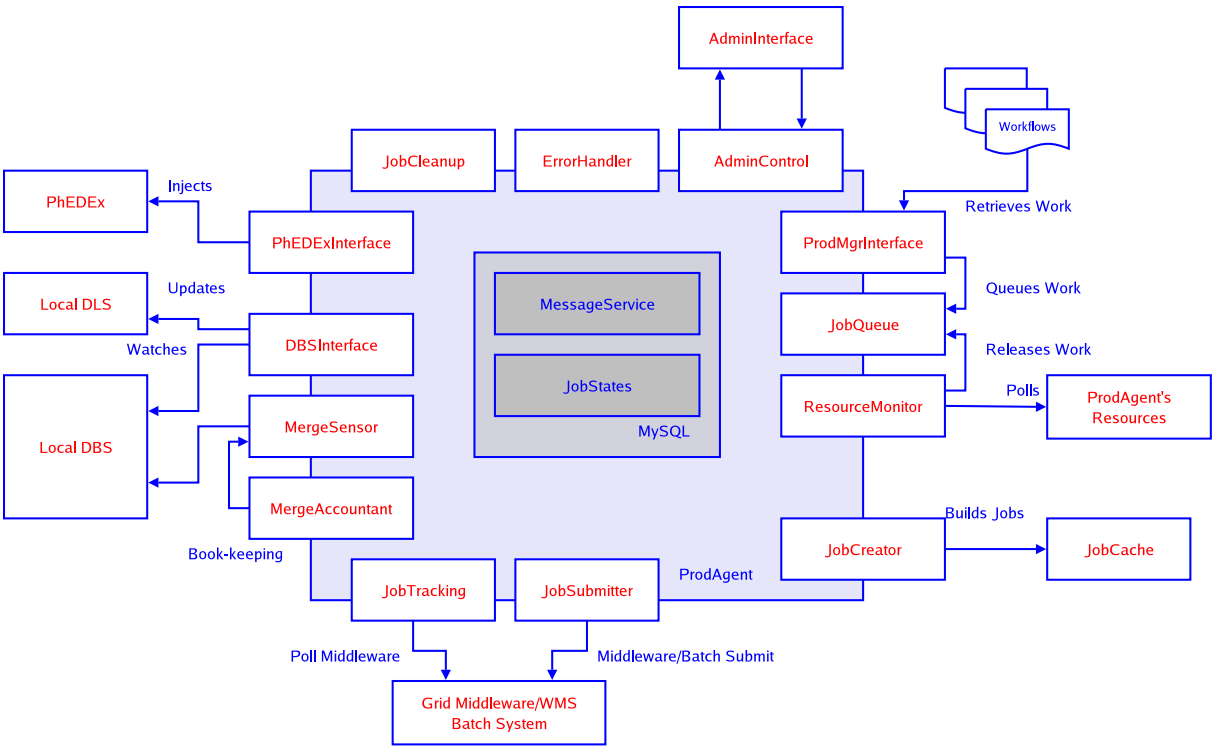


Figura 4.16: Arquitectura de ProdAgent.

4.2.2. Experiencia

Comparado con el flujo de trabajo en McRunjob, hay varias diferencias significativas. Los trabajos no usan una distribución de software específica para producción (como los ficheros DAR que usaba McRunjob). En su lugar, se hace uso del mismo repositorio de software estándar que para los trabajos de análisis. No se copian los ficheros de datos de input al disco local del Worker Node donde se ejecuta el trabajo, sino que se accede directamente a los ficheros guardados en el Storage Element a través de protocolos de acceso POSIX (como rfio o dcap). Los ficheros de output se guardan en el sistema de almacenamiento local, y sólo se intenta en un SE remoto alternativo cuando la operación en el sistema local falla. Los sistemas de almacenamiento local implementan un espacio de nombres estructurado, de forma que es suficiente el uso de un TFC para averiguar la localización física de los ficheros. No hay necesidad de usar ficheros de catálogos locales en formato XML para cada Dataset. Ya no se usa un catálogo global LCG para producción, sino que cada ProdAgent usa una instancia DBS/DLS de ámbito local para el bookkeeping y monitorización. En ProdAgent hay un paso adicional de merge de datos, y una etapa de "promoción de la información" desde las instancias locales a la global de DBS/DLS. El complejo proceso de publicación de datos de McRunjob se ha simplificado considerablemente. En particular, ya no es necesario generar los metadatos para cada Dataset (operación que debía ejecutarse en cada centro en el antiguo modelo de datos). En ProdAgent no hay recolección de datos a través de un nodo virtual PhEDEx. ProdAgent mantiene un mapeo entre SEs y nodos de PhEDEx. De este modo, durante la operación de registro de los datos en el sistema de transferencia, los datos se asocian a los nodos PhEDEx donde residen.

Los trabajos se envían a los centros que alojan datos. A diferencia de la producción con McRunjob, no existe la posibilidad de hacer copias remotas de los datos de input. Los datos pueden ser replicados con anterioridad si se solicita.

Una característica importante del nuevo entorno de procesamiento de sucesos de CMS es que puede encadenar las distintas fases de la simulación Monte Carlo. Las simulaciones se llevan a cabo usualmente en

una sola etapa (incluyendo generación, simulación, digitalización con o sin sucesos de pile-up, y reconstrucción), o en dos de forma ocasional (una de generación y simulación y otra para digitalización más reconstrucción). La primera consecuencia es la reducción del tiempo que tarda el proceso completo, al no ser necesarios los pasos de escritura y lectura de datos entre etapas. Pero también elimina algunos de los problemas típicos a la hora de acceder a las muestras de Pile-Up durante la digitalización, especialmente en algunos centros donde muchos trabajos intentaban acceder a estas muestras de forma simultánea colapsando el SE. Como esta etapa se ejecuta ahora combinada con la reconstrucción, lo que se traduce en un incremento del tiempo total de procesamiento de cada suceso, se reduce considerablemente el ritmo de lectura de estos sucesos de PU del sistema de almacenamiento, con el consiguiente aumento en la eficiencia.

ProdAgent ha heredado algunas de las soluciones que se implementaron para mejorar la robustez de McRunjob. Como hacer varios intentos en las operaciones de escritura de los datos de output, en varios SEs diferentes, o gestionar el menor número posible de ficheros de input y de output (sólo un fichero de datos, no hay ficheros de metadatos). Aún es crítico el soporte técnico en los centros. La producción sigue sufriendo por las inestabilidades en los servicios Grid (como RBs y CEs muy sobrecargados que no actualizan el estado de los trabajos), y los equipos de producción necesitan poder hacer un seguimiento de los problemas que ocurran en los centros. Por todo esto se sigue usando una estrategia de listas blancas (sólo los centros fiables y de alto rendimiento).

La figura 4.17 muestra la escala alcanzada por una de las instancias de ProdAgent (operada por el grupo del CIEMAT) durante la campaña de producción del verano de 2006, como fase de preparación para el ejercicio de computación CSA06 (descrito en la sección 5.3). Cada instancia puede enviar, aproximadamente, 3000 trabajos diarios. Este límite viene impuesto por la forma serializada en que se ejecuta esta operación (trabajo a trabajo). Cada operación de envío necesita entre 20 y 30 segundos, hasta que se obtiene una respuesta del RB. Esto incluye las tareas de autenticación, envío del Input Sandbox, interactuar con el servicio de red del RB, etc. Dado que un trabajo de producción típico del nuevo sistema (que incluye simulación, digitalización y reconstrucción) necesita entre 12 y 24 horas para procesar unos 500 sucesos (un suceso cada 2-3 minutos), cada instancia de ProdAgent puede producir del orden de un millón y medio de sucesos diarios. Se puede alcanzar la escala de producción deseada sin más que operar varias instancias de ProdAgent independientes en paralelo. Actualmente, ProdAgent está en fase de implementar las operaciones tipo bulk para enviar grupos de trabajos de forma simultánea, e incrementar así el número de trabajos que cada instancia de ProdAgent puede gestionar.

Los recursos de producción están distribuidos entre varios equipos, cada uno de los cuales hace uso de una lista específica de centros. De esta forma cada equipo se encarga de estar en contacto con los administrados locales de los centros que le han sido asignados. Este contacto permanente con los administradores de unos pocos centros permite una mejor y más rápida gestión ante la aparición de problemas en los centros. Sólo aquellos centros que mantienen con el tiempo un buen nivel de fiabilidad y que ofrecen una cantidad adecuada de recursos permanecen en las respectivas listas blancas de cada equipo de producción.

Los trabajos se envían con un Grid proxy que incluye una extensión (*role*) que identifica al usuario como operador de producción. Esta extensión permite que los trabajos, una vez que llegan al CE, se ejecuten en el WN bajo la identidad de un usuario local *virtual* de producción con ciertos privilegios. Así, estos trabajos tienen prioridad para el sistema local del gestión de trabajos con respecto a los de otros usuarios de la misma VO.

Cuando una parte del procesamiento necesita de un Dataset concreto como input, los trabajos se envían sólo a aquellos centros que tengan una copia de ese Dataset. A diferencia de McRunjob, el nuevo sistema de producción no puede descargar ficheros de input desde un SE remoto. La condición local del TFC impide la determinación del SURL de un fichero a partir de su LFN. Si es estrictamente necesario procesar un Dataset en un centro que no contiene ninguna copia, se replica a ese centro (mediante el sistema de transferencias de datos de CMS) antes de enviar los trabajos de procesamiento. Comparado con el antiguo EDM usado por McRunjob, el número de ficheros de input que los trabajos de producción

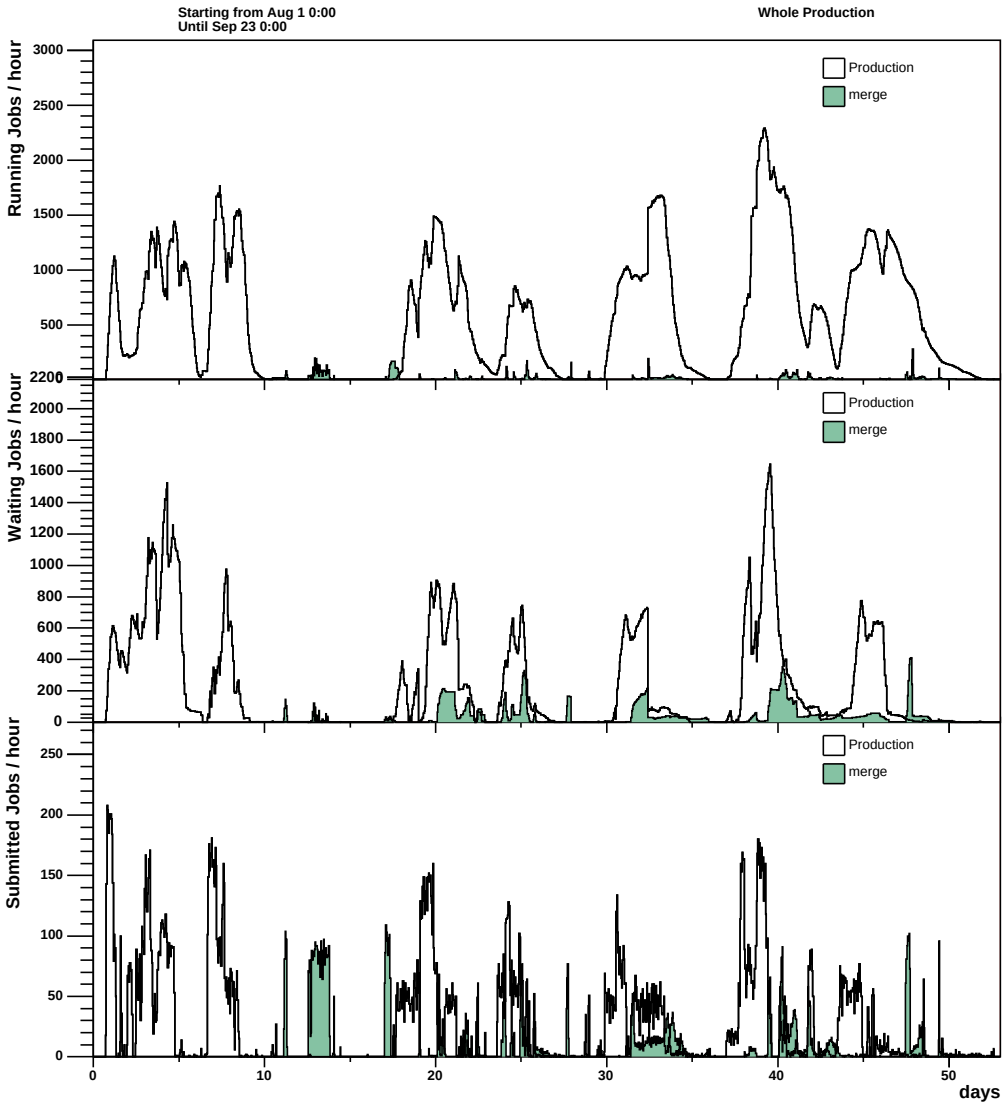


Figura 4.17: Número de trabajos en ejecución (arriba), encolados (medio) y enviados (abajo) por una instancia de ProdAgent en función del tiempo para la pre-producción del verano de 2006.

necesitan se ha reducido, pasando de 3 ficheros EVD y varios de metadatos a sólo un fichero de input con datos de sucesos. Los ficheros de output se copian exclusivamente en el SE local. Al igual que se hacía con McRunjob, esta operación se intenta varias veces en caso de fallo, y se hace uso de un SE remoto si finalmente no se puede llevar a cabo la copia local. Este SE remoto está especificado en un fichero de configuración local.

La inyección del flujo de trabajos en ProdAgent sigue siendo una operación manual. La ausencia de automatización en la inyección de trabajos se traduce en periodos temporales en los que no se explotan en su totalidad los recursos disponibles. Por tanto, el envío de trabajos y el número de trabajos en ejecución no fueron uniformes durante la producción del verano "pre-CSA06". Actualmente, el sistema está en

proceso de implementar una nueva componente para encolar trabajos a nivel de ProdAgent, monitorizar la cantidad total de recursos disponibles, y lanzar trabajos en concordancia con esos números. Esto permitirá gestionar una mayor cantidad de recursos.

La figura 4.18 muestra el número de sucesos acumulados con el tiempo, tanto para los trabajos de procesado como para los de merge, gestionados con la instancia de ProdAgent que operaba el grupo del CIEMAT de producción. A una parte de la muestra procesada no se le pudo aplicar el proceso de merge a causa de la rotura de un disco en uno de los centros. En la figura 4.19 se puede ver la distribución por centros de los sucesos producidos. Estos resultados corresponden a la instancia de ProdAgent operada en el CIEMAT y representan, aproximadamente, un cuarto de la producción total del verano del 2006, que fue operada por cuatro equipos.

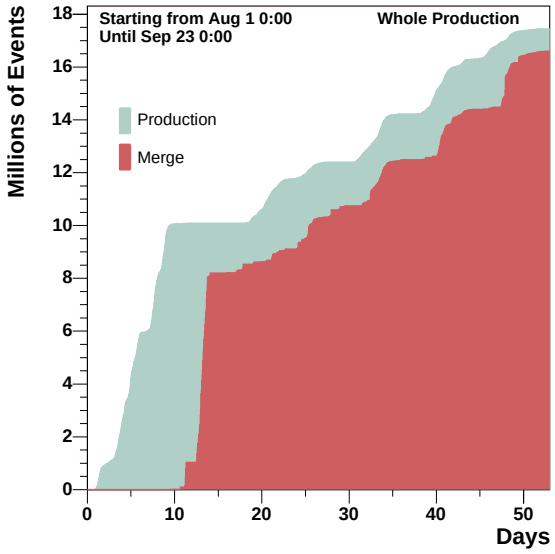


Figura 4.18: Número de sucesos acumulados en función del tiempo, para las operaciones de procesado y de merge, durante la pre-producción del verano del 2006, correspondientes a una instancia de ProdAgent.

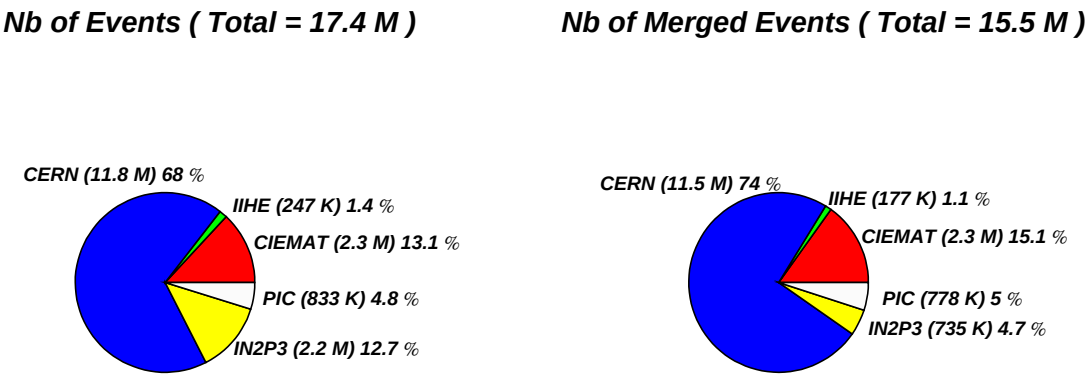


Figura 4.19: Distribución de centros donde se ejecutaron los trabajos de producción de sucesos, y los correspondientes de merge, durante la pre-producción del verano del 2006, correspondientes a una instancia de ProdAgent.

En la figura 4.20 se muestra el tiempo de procesado, para varios Datasets, tanto para trabajos de procesado (izquierda) como de merge (derecha). Como se puede comprobar, este tiempo varía de forma significativa dependiendo del tipo de suceso físico que se simula. El tiempo menor corresponde a sucesos de minimum bias, bastante más sencillos que los sucesos de señal (como, por ejemplo, los que contienen interacciones de *Electroweak*). En general, los sucesos de señal suelen necesitar cuatro veces más tiempo que los de minimum bias. El mismo razonamiento se aplica en el caso de trabajos de merge, ya que el tamaño de cada suceso de minimum bias también es un cuarto del de los sucesos de señal, y el tiempo necesario para las operaciones de merge es proporcional al tamaño de los ficheros que se leen y procesan. Los dos picos que se pueden apreciar, claramente diferenciados, en la muestra de minimum bias para los trabajos de merge son consecuencia directa del distinto rendimiento ofrecido por los sistemas de almacenamiento en aquellos centros donde se ejecutaron estos trabajos. En la figura 4.21 se puede ver la distribución de tiempos de esta muestra exclusivamente, distinguiendo por centros. Los trabajos ejecutados en el CIEMAT fueron los más rápidos, mientras que el segundo pico corresponde al CERN.

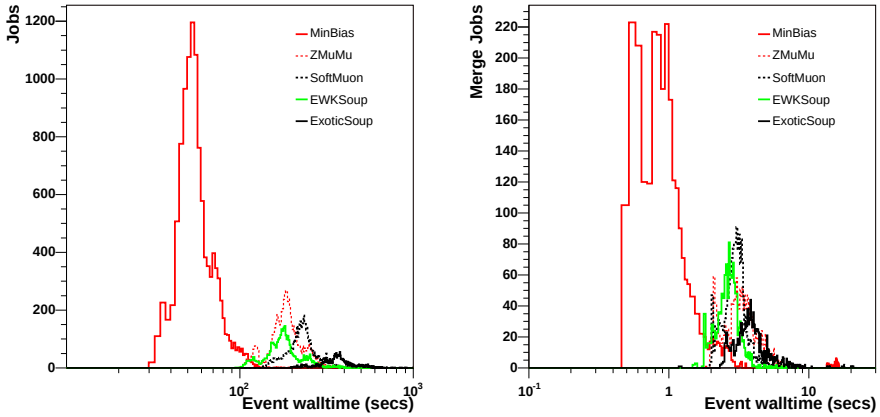


Figura 4.20: Distribución del tiempo de procesado por suceso para los trabajos de producción (izquierda), y los correspondientes de merge (derecha) durante la pre-producción de verano del 2006.

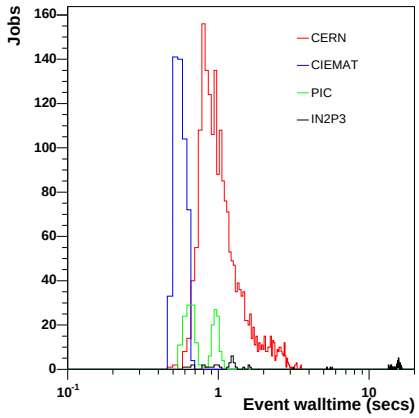


Figura 4.21: Distribución del tiempo de procesado por suceso para un determinado workflow, distinguiendo por centros, durante la pre-producción de verano del 2006.

En la tabla 4.3 se muestran los porcentajes de errores (relativos al número total de trabajos enviados) tanto para trabajos de procesamiento como de merge. Los valores más altos corresponden a fallos en la aplicación (incluyendo el acceso a los datos) y a ineficiencias en los servicios Grid. Se puede comprobar que el porcentaje de fallos por esta causa es mucho mayor, casi el doble, para los trabajos de merge que para los de procesamiento. Esto es debido a que los trabajos de merge acceden a un número mucho mayor de ficheros de input que los de procesamiento.

Causa de fallo	**Trabajos de procesamiento**	**Trabajos de merge**	**Global**
En la aplicación y en el acceso a los datos	5.5 %	10.6 %	6.6 %
En el acceso al software del experimento	2.6 %	1.3 %	2.3 %
En la operación de stageout	4.0 %	3.2 %	3.8 %
Ineficiencia del Grid	5.3 %	11.0 %	6.7 %
Sin clasificar	<0.01 %	0 %	<0.01 %
Total	17.3 %	26.1 %	19.4 %

Tabla 4.3: Distribución de las causas de fallo para los trabajos de producción y de merge, y los valores globales.

Las ineficiencias en los servicios Grid incluyen fallos en los servicios globales (Resource Brokers y Sistemas de Información) y en las componentes locales de los centros (Computing Elements y Worker Nodes). Los errores asociados a los servicios Grid suelen ocurrir cuando no se puede comunicar al RB el código de salida de los trabajos (porque el trabajo ha desaparecido por problemas de hardware en el WN o ha sido cancelado por el sistema local al alcanzar el tiempo máximo permitido en cola, por ejemplo), o durante su envío si el nodo de destino desaparece momentáneamente del sistema de información. Esto último explica por qué el porcentaje relativo de errores asociados al Grid es mucho mayor para los trabajos de merge. Estos trabajos se envían a nodos concretos (aquellos donde están guardados los datos), mientras que los trabajos de procesamiento pueden enviarse y ejecutarse en cualquier centro. Si un centro en particular desaparece temporalmente del sistema de información, los trabajos de merge que se manden a ese sitio fallarán mientras que los trabajos de procesamiento se ejecutarán en cualquier otro nodo.

Los errores durante la operación de guardado de los datos de output en el sistema de almacenamiento local suman, aproximadamente, un 4 % del total. Finalmente, en torno al 2 % de los trabajos fallan por no poder acceder al repositorio donde está instalado el software del experimento (este acceso suele ser via NFS). En cualquier caso, tanto para trabajos de procesamiento como de merge, este apartado es el que ofrece un menor porcentaje de errores.

La figura 4.22 muestra el número de intentos que fueron necesarios para ejecutar con éxito cada trabajo (distinguiendo entre procesamiento y merge). La eficiencia para los trabajos de procesamiento es de un 83 %, para los de merge es de un 74 %, y el valor global está en torno al 80 %.

Un soporte eficaz por parte de los centros sigue siendo imprescindible para conseguir una buena eficiencia en las tareas de producción. La mayoría de los fallos en los trabajos están relacionados con problemas en los centros (como fallos temporales en los servicios Grid, problemas leyendo y/o escribiendo datos en el sistema de almacenamiento local, mal funcionamiento de los WN o del sistema local de gestión de colas, o dificultades para acceder al software del experimento). Una pronta reacción por parte de los administradores locales es imprescindible para minimizar las ineficiencias en la producción. Por ejemplo, un WN desconfigurado donde todos los trabajos fallan puede verse como un *agujero negro* que atrae constantemente nuevos trabajos, que fallarán inevitablemente. La necesidad de investigar los problemas en los centros y lo limitado de la escala que puede alcanzarse con una instancia individual de ProdAgent son las principales razones que justifican la existencia de varios equipos de producción. Se espera una reducción progresiva de los recursos humanos necesarios para operar la producción Monte Carlo del experimento a medida que los centros y los servicios Grid sean cada vez más fiables y el sistema de

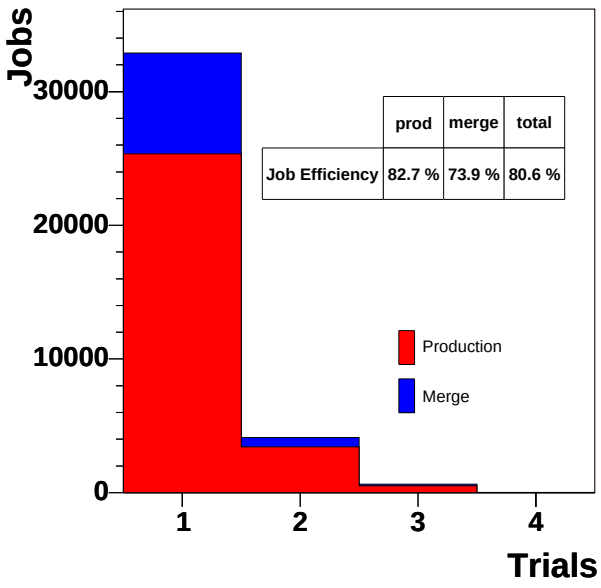


Figura 4.22: Distribución del número de intentos para cada trabajo de producción y de merge durante la preproducción del verano de 2006.

producción incorpore las bulk operations, que permitirán a cada instancia individual manejar un número mucho más grande de trabajos.

4.2.3. Monitorización

Durante las primeras operaciones de producción con ProdAgent se desarrolló una herramienta de monitorización específica [114]. Esta herramienta puede obtener información de distintas bases de datos (locales y remotas), de forma que permite controlar los trabajos correspondientes a distintas instancias de ProdAgent operadas en paralelo. La figura 4.23 muestra un esquema de esta herramienta, y 4.24 un ejemplo del output que proporciona. La información incluye variables como el número de sucesos procesados, de trabajos enviados, encolados, en ejecución y ejecutados (con y sin éxito), distribuciones de errores, etc. Esta información se puede clasificar por centros y por workflows, y se puede obtener para los trabajos de procesamiento y de merge. Se puede, además, restringir el período de tiempo, el workflow o el centro que se desea analizar. Esto permite depurar con mayor facilidad las posibles causas de error.

4.2.4. Futuros desarrollos

Las tareas de desarrollo en el sistema de producción Monte Carlo continúan, con el objetivo de mejorar la automatización, robustez, rendimiento y escala que se puede alcanzar, reduciendo al mismo tiempo la cantidad de recursos humanos necesarios para las tareas de operaciones.

Recientemente se han implementado algunas funcionalidades como la gestión de bloques de datos (cierre automático de un bloque cuando alcanza un tamaño determinado o un cierto número máximo de ficheros), la migración desde las instancias locales de DBS/DLS a la instancia global, y la inyección en el sistema de transferencia. Se automatizará el procesamiento multi-etapa (es decir, el procesamiento de varios trabajos donde el output de uno es el input del siguiente) mediante la introducción de una nueva componente, que recibirá el nombre de *ProcSensor* (o sensor de procesamiento). Esta componente inspeccionará DBS buscando datos de input disponibles, que ya hayan sufrido el merge, para iniciar su procesamiento. Se está perfeccionando la automatización en la gestión de los workflows mediante las componentes ProdRequest y ProdManager. De momento son los operadores los encargados de introducir

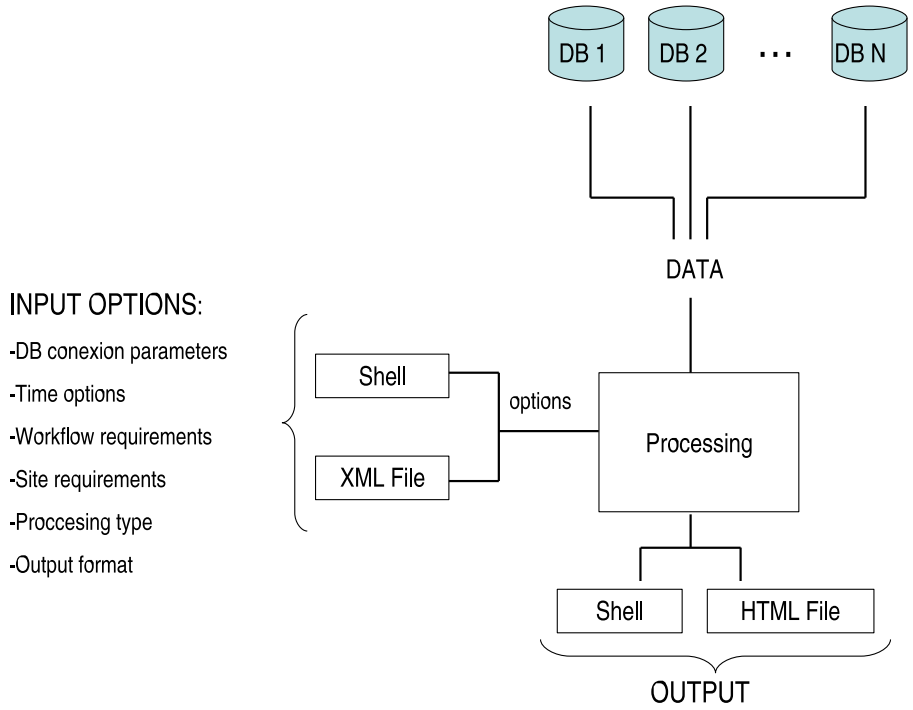


Figura 4.23: Esquema de la herramienta de monitorización desarrollada para la producción de Monte Carlo con ProdAgent.

los workflows en las instancias de ProdAgent. También se automatizarán las tareas de finalización de los workflows (eliminando los trabajos que queden pendientes, borrando entradas innecesarias en la base de datos y liberando el espacio de caché reservado).

En lo que concierne a la escala que se puede alcanzar con el sistema, el número de trabajos que una instancia individual de ProdAgent puede gestionar aumentará drásticamente con la introducción de bulk operations durante la creación, envío y seguimiento de los trabajos. Las CPUs disponibles para producción MC se aprovecharán de forma más eficiente mediante la introducción de una nueva componente de monitorización de recursos. Esta componente buscará recursos disponibles (preguntando al sistema de información Grid o a la componente JobTracking de ProdAgent) y lanzará nuevos trabajos, previamente inyectados y encolados a nivel de ProdAgent.

Para evitar que los gestores locales de colas cancelen los trabajos al alcanzar el tiempo máximo permitido de permanencia en la cola se pasará del actual sistema de trabajos, basados en sucesos, a uno de trabajos basados en tiempos. Los trabajos no procesarán entonces un número fijo de sucesos, sino tantos como se pueda durante un determinado periodo de tiempo (ajustado a la longitud de las colas en los centros).

Se han identificado otros factores que implican un uso ineficiente de los recursos de CPU. Algunas veces los trabajos se quedan colgados por diversos motivos como, por ejemplo, intentando acceder a los datos de input. Estos trabajos colgados bloquean la cola, pero sin ejecutar ningún tipo de procesamiento. Este tipo de trabajos se pueden identificar fácilmente comparando el tiempo total con el tiempo de CPU real, y el sistema de gestión de colas local los puede cancelar. ProdAgent también podría cancelar aquellos trabajos que llevan en ejecución un tiempo muy superior al esperado. Otra causa de ineficiencia es la

```
./PAmonBOSSDB.py [DB options] --type=status

             Evt Success Failed Done Running Sched Waiting Aborted Cancel Submitted
ciemat.es 1254214    2714    193 2778      19   172       2     734     33      3747
   pic.es  486426    1297     22 1268      39    66       0     311      0      1686
  ifca.es  326562     748     34  782      10    65       0     211      0      1068
  cern.ch 1269136    3483    182 3652     132   238       4    2464      0      6503
    TOTAL 3336338    8242    431 8480     200   541       6    3720     33     13004

./PAmonBOSSDB.py [DB options] --type=codes

            0 No 0  1 64 65 134 137 139 10032 10035 60312
  cern.ch 3482  182 54  6 44   4   2  51    10     0    11
ciemat.es 2586  189  0  1  1   0   0   0     0     1   186
   pic.es 1292   22  0  0  0   1   0   0     0     0    21
  ifca.es  748   34 23  2  5   4   0   0     0     0     0
    TOTAL 8108  427 77  9 50   9   2  51    10     1   218

./PAmonBOSSDB.py [DB options] --classify=workflows --type=status

               Evt Success Failed Done Running Sched Waiting Aborted Cancel Submitted
  DY_ee_mass_10 100       5      1    6       0     0       0       0      0         6
DY_mumu_mass_10  60       3      0    3       5     0       0       0      0         8
          TOTAL 160       8      1    9       5     0       0       0      0        14

./PAmonBOSSDB.py [DB options] --classify=workflows --type=codes

               0 No 0 60312
  DY_ee_mass_10 5    1     1
DY_mumu_mass_10 3    0     0
          TOTAL 8    1     1
```

Figura 4.24: Ejemplo del output devuelto por la herramienta de monitorización de la producción Monte Carlo.

presencia de nodos que, por tener algún tipo de problema en su configuración, cancelan de forma casi instantánea todos los trabajos que les llegan. Estos nodos aparecen siempre, por tanto, como disponibles en el sistema de información, por lo que el sistema de colas local les seguirá enviando constantemente nuevos trabajos que serán cancelados. Esta situación es en la actualidad difícil de detectar y evitar en tiempo real, pero podría ser detectada con un buen sistema de monitorización.

Se está implementando un sistema de monitorización y accounting global para las tareas de producción mediante una nueva componente de monitorización en ProdAgent. Para cada instancia de ProdAgent, esta componente mandará a un servidor global, de forma fiable y periódica, estadísticas sobre los trabajos ejecutados. Esta información se puede analizar con posterioridad proporcionando información valiosa para la optimización del sistema. Esta extensión complementará al actual sistema de monitorización global, basado en el envío no fiable de informes individuales desde los WNs al Dashboard.

4.3. Sistema de transferencia de datos

El sistema de transferencia de datos de CMS, PhEDEx [53], ha sido diseñado para gestionar las transferencias de datos del experimento con el menor esfuerzo por parte de los operadores, automatizando los flujos de trabajos, desde la distribución a gran escala de los Datasets hasta las transferencias fiables y escalables de ficheros individuales.

Generalmente, la infraestructura sobre la que se sustenta la gestión de los datos es poco fiable. Para

hacer el sistema más fiable, robusto, y de alto rendimiento, incluso cuando se basa en infraestructuras inestables, PhEDEx hace uso de principios bien establecidos del diseño de sistemas asíncronos. El uso de estos principios y de la combinación de algunas técnicas ya conocidas permite a PhEDEx ser eficiente en un entorno operativo.

El sistema PhEDEx es, básicamente, un gestor de distribución de datos a gran escala. Esto significa que no proporciona herramientas de bajo nivel para ejecutar las tranferencias, interaccionar con los sistemas de almacenamiento, o para las funciones de catalogación. Las transferencias se ejecutan haciendo uso de las herramientas de copia punto a punto suministradas por LCG, poco fiables por construcción. La interacción con los sistemas locales de almacenamiento se hace a través de agentes locales mediante *scripts* configurables.

4.3.1. Arquitectura del sistema

El diseño e implementación de PhEDEx se ha regido por un conjunto de principios arquitecturales. El principio clave es que las funcionalidades más complejas deberían mantenerse en unidades atómicas y diferenciadas, y que el reparto de responsabilidades en cada etapa del flujo de trabajo de transferencia debería hacerse mediante el paso de mensajes entre estas unidades. Es deseable que el número y contenido de información de estos mensajes sea el menor posible. Dado que el sistema consta de componentes activas es razonable modelar estas unidades como agentes más que como servicios pasivos sin estado.

Una de las reglas principales que se siguió en el diseño del sistema fue evitar la creación de servicios que no sean necesarios. En su lugar, se consideró menos perjudicial un pequeño incremento en la complejidad de las componentes de transferencia del "cliente".

El diseño de PhEDEx está caracterizado por una cierta abstracción por capas [99]. Muchas de las operaciones y herramientas externas a PhEDEx no son muy fiables. Para manejarlas, se han envuelto estos sistemas y herramientas poco fiables dentro de capas más robustas y construir así un sistema de transporte de datos fiable. Las capas que componen la arquitectura de PhEDEx (ver figura 4.25) son las siguientes:

- **Transferencias punto a punto y tecnologías no fiables.** Operaciones básicas que usan los componentes fundamentales del sistema: srm, gsiftp, etc.
- **Transferencias fiables punto a punto, o *single hop*.** Recuperación en caso de fallos y reintento de las transferencias.
- **Transferencias fiables enrutadas, o *multi hop*.** Entrega eficiente de responsabilidades de nodo a nodo en la cadena de transferencia. Gestiona agrupaciones de migraciones y copias de conjuntos de ficheros de cinta a disco.
- **Transferencias de Datasets.** Monitoriza las transferencias. Activa y desactiva las operaciones. Modificación dinámica de la ruta en caso de fallos. Recolección automática de ficheros.
- **Gestión de réplicas.** Asignación de ficheros a los destinos basándose en subcripciones. Determinación de la réplica mejor (más cercana) para ser transferida. Gestión de réplicas globales basada en demandas.

Para limitar la complejidad del sistema, estas operaciones son ejecutadas por una serie de procesos autónomos, robustos y persistentes, conocidos como agentes. Estas componentes de software sólo dependen indirectamente unas de otras, de forma que ninguna componente conoce la existencia de las demás ni sus funciones. Esto ha suministrado al sistema niveles cada vez mayores de autonomía, permitiendo tomar decisiones basadas en las condiciones locales. Un agente es un sistema de computación encapsulado que, colocado en un cierto entorno, es capaz de una actuación autónoma y flexible en ese entorno para satisfacer sus objetivos de diseño:

- resuelven problemas que sean claramente identificables;

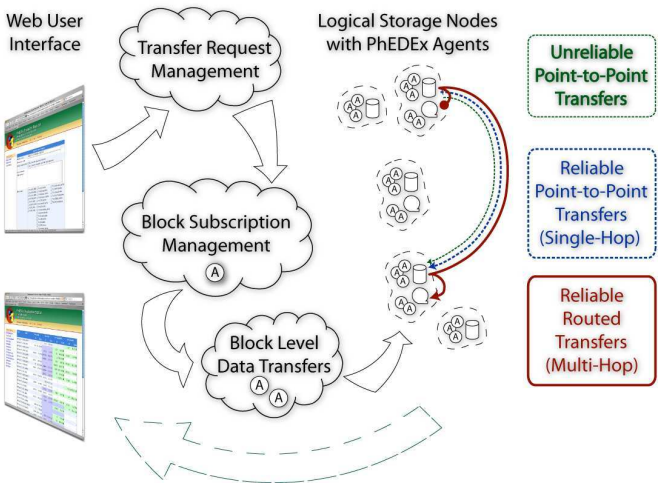


Figura 4.25: Reparto de responsabilidades entre las capas que forman la arquitectura de PhEDEx.

- están ubicados en un entorno particular, reciben información del estado de dicho entorno y actúan sobre el mismo;
- están diseñados para desempeñar un papel específico, con unos objetivos particulares para resolver problemas particulares;
- autonomía, controlan por si mismos su propio estado y su comportamiento;
- son capaces de un comportamiento flexible resolviendo problemas para satisfacer los objetivos de diseño, reaccionando ante un cambio en el entorno, y adoptando nuevos propósitos oportunistas y tomando iniciativas.

Estos agentes comparten información sobre el estado de las réplicas y de las transferencias mediante lo que se conoce como *arquitectura blackboard* [6]. El blackboard también guarda cierta información de alto nivel sobre el enrutamiento a través de la red, las subscripciones a los Datasets y otra información sobre la infraestructura. La figura 4.26 muestra un esquema de esta arquitectura blackboard.

Cuando una tarea es gestionada por un conjunto de procesos independientes se hace necesaria una asignación ordenada y organizada de responsabilidades. En PhEDEx, las responsabilidades de las tareas generalmente están pre-asignadas a instancias conocidas de los agentes (aunque podrían asignarse de forma más dinámica). Cuando se completa una tarea, el agente correspondiente fija su estado en el blackboard. A menudo, esto provoca la creación de una nueva tarea para otro agente. El flujo de trabajo de una transferencia se define, por tanto, por estos intercambios de estados.

Las componentes que conforman PhEDEx son:

- Una base de datos Oracle [116] para la gestión de las transferencias, conocida como *Transfer Management Database and Bookkeeping* (TMDB). Esta base de datos es la que ofrece las funcionalidades del blackboard en la arquitectura del sistema.

[6] La arquitectura blackboard está definida en función a su componente principal, el blackboard. Un blackboard [115] es una estructura de datos que puede ser leída y modificada por programas llamados fuentes de conocimiento -*Knowledge Source* (KS)-. Cada KS se especializa en la resolución de una parte particular de una tarea completa. Así, todas las KS trabajan en forma conjunta en la búsqueda de una solución.

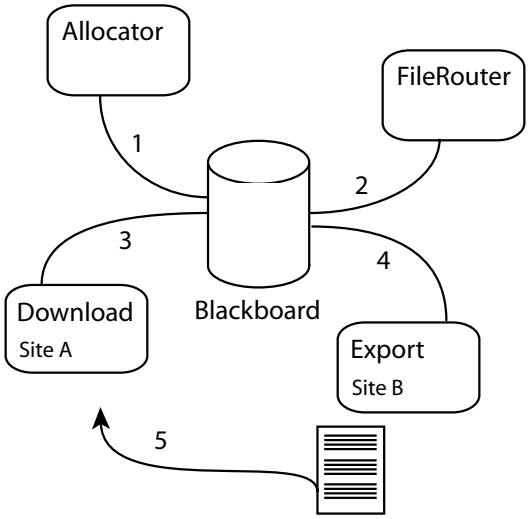


Figura 4.26: Esquema de la arquitectura blackboard de PhEDEx.

- Agentes de transferencia que gestionan el movimiento de los ficheros entre centros. Incluye agentes para la migración de ficheros a sistemas de almacenamiento masivo, y agentes para manejar estas unidades de almacenamiento de forma eficiente basándose en demandas de transferencias.
- Agentes de gestión, encargados de asignar ficheros a los distintos destinos basándose en las subscripciones de los centros a los datos, y agentes para mantener la información de enrutamiento a través de la topología de las transferencias de los ficheros.
- Herramientas para manejar las peticiones de transferencia.
- Agentes locales para manejar los ficheros localmente. Estos agentes interactúan con el sistema de almacenamiento local y realizan todas las operaciones necesarias para que los ficheros estén disponibles para ser copiados o transferidos. Estas operaciones previas incluyen tareas como la inyección en el TMDB, la copia previa desde cinta a disco para facilitar su posterior gestión, etc.
- Herramientas de monitorización via web.

4.3.2. Implementación del sistema

4.3.2.1. Flujo de trabajo

En la figura 4.27 se puede ver el flujo de trabajo de PhEDEx. Cuando hay que transferir uno o varios ficheros a un destino determinado se introduce en el TMDB una subscripción. Los agentes centrales de enrutamiento seleccionan la mejor ruta y crean en el TMDB una solicitud de transferencia para cada fichero que debe ser transferido. El agente de transferencia del nodo de destino ve entonces que hay datos asignados al nodo y marca las solicitudes de transferencias como *wanted* (solicitado). El agente de exportación del nodo de origen detecta la existencia de ficheros marcados como *wanted* en su nodo y, si están en disco, los marca como *available* (disponible). En caso contrario espera a que el agente que se encarga de las copias de ficheros desde las cintas declare que éstos están en disco. Finalmente, el agente de exportación suministra el SURL al TMDB y el agente de transferencia procede a la descarga de los ficheros.

4.3.2.2. Agentes básicos del sistema

La arquitectura de agentes, donde éstos se reparten las tareas y ejecutan acciones de forma independiente, se ha llevado a la práctica en PhEDEx mediante un modelo de dos niveles. Así, las responsabilidades se

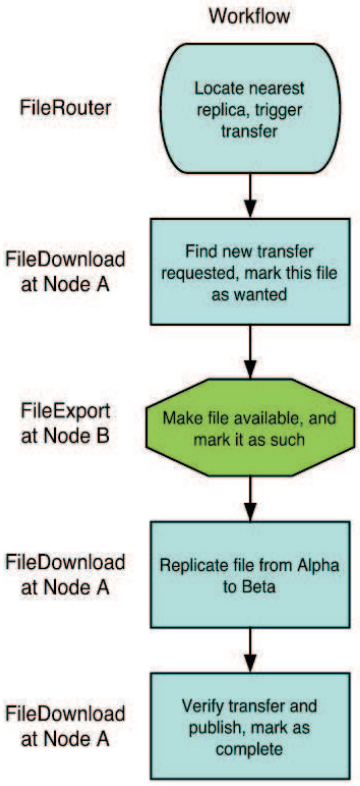


Figura 4.27: Workflow de una transferencia gestionada por PhEDEx.

reparten entre agentes de ámbito local y global.

Los agentes de ámbito global se encargan de las tareas de enrutamiento, monitorización, etc. Los agentes locales asumen la responsabilidad de interaccionar directamente con el sistema de almacenamiento local. De esta manera se suplen algunas de las carencias de las que todavía adolece SRM (como la falta de una interacción apropiada con los sistemas de cintas que permita ejecutar de forma eficiente movimientos masivos de ficheros a los discos). Esto es especialmente significativo en el caso de los Tier-1, donde siempre existe almacenamiento en cinta. En el caso de los Tier-2, donde sólo hay discos, es más fácil la interacción con el sistema de almacenamiento a través de los servicios SRM (y no sería estrictamente necesario el uso de agentes locales). Sin embargo, en general, una razón de peso para hacer uso de agentes locales es que permiten la implicación directa de los administradores locales, lo que acelera la resolución de problemas. En cada nodo local de la topología hay varios tipos de agentes, dependiendo del tipo de centro.

Agentes de Transferencia. Se encargan de obtener los ficheros, por lo que debe haber al menos uno en cada centro. Las transferencias se pueden optimizar ajustando algunos parámetros como el tiempo de espera hasta que se completa una transferencia individual, el número de ficheros asociados a una transferencia, y el número de transferencias ejecutadas en paralelo. Para evitar enlaces excesivamente lentos, debido a un bloqueo de las transferencias durante largos periodos de tiempo, se puede hacer que varios agentes se ejecuten en paralelo, cada uno responsable de las transferencias asociadas a ciertos enlaces.

Agentes de Exportación y Borrado. El agente de Exportación es el encargado de suministrar el PFN

local (utilizando las reglas de mapeo entre PFN y LFN contenidas en el TFC) y de declarar los ficheros como transferibles. En cada centro debe, por tanto, haber al menos uno en ejecución. La cualidad de transferibilidad de los ficheros no es inmediata en los sistemas de almacenamiento con cinta, pues los ficheros no están disponibles para ser copiados hasta que no estén en un disco. En caso de que el sistema sólo disponga de disco no es necesaria ningún tipo de intervención (y generalmente SRM se encarga de las operaciones de copia más simples).

Agentes de gestión de MSS. En los centros que disponen de sistemas de almacenamiento MSS (con discos y cintas), como son todos los centros Tier-1, son necesarios dos agentes encargados de los movimientos de los ficheros de cinta a disco (*stager*) y de disco a cinta (*migrator*), respectivamente. PhEDEx gestiona los ficheros de forma distinta si éstos se encuentran disponibles para su lectura en disco o si se hayan ubicados en una cinta. Como hay varios sistemas distintos de almacenamiento MSS, no se puede desarrollar un agente común válido para todos ellos. Por el contrario, cada tipo de almacenamiento debe disponer de su propia versión del agente. El agente de migración suele ser trivial, pues es el propio sistema de almacenamiento el que realiza las migraciones de disco a cinta, y este agente simplemente se encarga de chequear si los ficheros han sido realmente copiados, y de marcarlos como tal en TMDB.

Agentes de inyección. El agente de inyección es el que actualiza la información en TMDB para que los nuevos ficheros sean conocidos por PhEDEx. Además, el agente inyecta cierta información sobre los ficheros como el LFN, su tamaño, el *checksum*[7], el nodo donde está, o el Block y Dataset al que pertenece.

4.3.2.3. Robustez en las transferencias

Dada la importancia crucial de las transferencias de datos, se ha hecho un gran esfuerzo para optimizar estas operaciones. Teniendo en cuenta que cada operación, por sencilla que sea, puede fallar, se ha dotado al sistema de operadores seguros para numerosas tareas (como la escritura de ficheros temporales o la creación de subprocesos, por ejemplo), de modo que las operaciones más simples son robustas.

La técnica implementada para resolver fallos es bastante simple. Se registra lo sucedido, se vuelve hacia atrás, y se intenta de nuevo la operación tras un cierto tiempo de espera que permita la recuperación de la componente del sistema que está causando los fallos. Este tiempo de espera aumenta si vuelve a haber errores. El número de reintentos es un parámetro configurable. Este algoritmo se conoce con el nombre de *cool-off*. La forma de implementar los reintentos es sencilla: se limpian ciertas cachés que guardan el estado local, y es la falta de consistencia del sistema global la que "dispara" la ejecución del nuevo intento.

Para aumentar el rendimiento en las operaciones de transferencia se implementó un cierto nivel de paralelismo en las mismas mediante dos mecanismos simultáneos. Por una parte, se ejecutan varias transferencias en paralelo abriendo un *thread* para cada una de ellas, y en cada transferencia se abren varios *sockets* TCP[8] [117] a través del protocolo GridFTP para enviar datos en paralelo para cada uno de ellos.

También se lleva a cabo chequeo de errores internos para evitar la propagación de errores entre partes del sistema que podrían ser independientes. También se verifica, de forma independiente, la existencia y tamaño de los ficheros para cada transferencia.

4.3.2.4. Operaciones de enrutamiento

El enrutamiento de los datos entre centros se consigue gracias a la forma en la que PhEDEx simula los centros. Se usa una red virtual que describe una cierta topología en la que los nodos representan recursos de almacenamiento, independientemente de la infraestructura de red subyacente. Desde un punto de vista

[7] una suma de verificación o checksum es una forma de control de redundancia, una medida muy simple para proteger la integridad de datos, verificando que no han sido corrompidos.

[8] siglas de *Transmission Control Protocol*, o protocolo de control de transmisión. Es el protocolo más usado en Internet, y puede ser usado por los programas para establecer conexiones entre ellos a través de las cuales se envían datos.

técnico, esta topología es un grafo con pesos, generalmente no completamente conectado. Esto permite a PhEDEx cachear temporalmente los datos en centros intermedios antes de llegar a su destino, y gestionar así mejor la carga en las componentes centrales del sistema.

Se mantiene esta red virtual mediante un algoritmo de *camino más corto*. Los caminos más cortos se calculan mediante el algoritmo de Dijkstra [118]. Se guarda en el blackboard la lista de vecinos con información de los pesos de los enlaces estáticos. Los agentes de enrutamiento actúan en (y de parte de) cada nodo en la red, y usan el algoritmo de Dijkstra para refrescar dinámicamente el árbol de cobertura mínimo desde su nodo hasta todos los demás nodos de la red. La información del árbol de cobertura mínimo (que incluye datos sobre el origen, el destino, los nodos intermedios, etc.) se guarda en una tabla de enrutamiento en el blackboard.

Un agente de enrutamiento de ficheros actúa de parte de cada nodo en la red, y es responsable de iniciar un conjunto de replicaciones que forman parte de una operación de transferencia desde la fuente al destino. El agente usa la tabla de enrutamiento para determinar el camino más corto entre origen y destino, e inicia la primera transferencia en la cadena insertando una fila en la tabla de estados de las transferencias dando información del origen y el destino. Cuando se marca la transferencia como completa se recalcula de nuevo la réplica más cercana para cada fichero (teniendo en cuenta que existe una nueva copia del último fichero transferido) y se inicia la siguiente transferencia en la cadena.

Hay que mencionar, sin embargo, que la opción de copias usando nodos intermedios (multi-hop) está desactivada. Esto es así porque harían falta recursos adicionales en los nodos intermedios y PhEDEx aún no ha desarrollado adecuadamente la operación de borrado de las cachés intermedias una vez que el fichero ha llegado a su destino final.

4.3.2.5. Implementación de las políticas y prioridades específicas de CMS

Uno de los mayores logros de PhEDEx ha sido que permite cumplir, de forma satisfactoria, los requisitos particulares del modelo de computación de CMS.

El mecanismo por el cual los centros (modelados como simples nodos de la topología) se subscriben a los datos, y son agentes independientes los que ejecutan las operaciones de replicación de los datos a los centros subscriptores, resulta altamente flexible y dinámico. Gracias a esto ha resultado fácil implementar las políticas de flujos de datos de CMS (que incluyen el flujo de datos reales desde el CERN hasta a los Tier-1 y de éstos a los Tier-2, los resultados de los pases de filtrado desde los Tier-1 a los Tier-2, los resultados de la producción Monte Carlo desde los centros Tier-2 a los Tier-1, y todas las operaciones de réplica entre centros Tier-1). En la topología actual de PhEDEx, por tanto, el Tier-0 está conectado a todos los Tier-1, éstos lo están entre sí, y cada Tier-2 está conectado también a todos los Tier-1. No se trata de una topología completamente conectada porque no hay conexión entre los nodos Tier-2. Los centros Tier-2 están conectados a todos los Tier-1 (y no sólo al suyo de referencia) porque los datos no estarán bien entendidos durante los primeros años de operación, lo que obligará a ejecutar los análisis sobre los RECO Data y RAW Data (repartidos entre todos los Tier-1) en lugar de con AOD (para los que cada centro Tier-1 almacena una copia completa).

PhEDEx permite, además, implementar una serie de políticas de priorización de las actividades. Uno de los aspectos fundamentales en la distribución de datos en los experimentos de Física de Altas Energías es la gestión de las prioridades en las transferencias. Usualmente, las operaciones de escritura de datos tienen mayor prioridad que las de lectura. Debería poder alcanzarse fácilmente un acuerdo entre estos dos casos con un sistema de prioridades simple. Dentro de estos casos hay, además, grados de importancia. Por ejemplo, algunos RAW Data son importantes para tareas de calibración mientras que el resto lo son para su análisis. Algunos datos resultantes de los análisis podrían ser de utilidad para una amplia comunidad de usuarios. En general, las transferencias de los RAW Data a los centros Tier-1 son prioritarias. Por otro lado, debe alcanzarse también un acuerdo entre las prioridades locales y globales. Habitualmente

los grupos de física y los investigadores tienen Datasets preferidos (ligados de alguna forma a la localización). Aunque los centros desean priorizar los Datasets para los que ellos son el destino, también han de proporcionar espacio suficiente para los ficheros que están en tránsito.

Para cumplir con estas políticas y prioridades se han buscado una serie de mecanismos. Por ejemplo, los algoritmos de programación y las prioridades se traducen en el establecimiento de fechas límites (*deadlines*) más o menos estrictas. Estos mecanismos se establecieron a nivel de los agentes de enrutamiento, transferencia y exportación, con lo que el esfuerzo está distribuido y no centralizado. El único requisito es poder expresar las prioridades y las políticas de una forma configurable.

Por otro lado, como los datos están agrupados en Datasets y bloques en el modelo de computación de CMS, PhEDEx está preparado para trabajar con estos conjuntos de ficheros. Las operaciones sobre Datasets y sobre bloques son más rápidas, pues las tablas que los describen son más compactas.

Otras operaciones necesarias para la gestión de los datos en CMS (como el borrado de ficheros, la posibilidad de moverlos sin replicarlos, etc.) también se pueden ejecutar con PhEDEx mediante agentes dedicados.

4.3.3. Optimización del sistema

En el diseño original las tareas se asignaban explícitamente a instancias de agentes específicas en centros específicos. Este mecanismo se ha mejorado de forma crucial, con un comportamiento más dinámico de los agentes. Asímismo, se han desarrollado grandes mejoras en el acceso a la base de datos central, en los algoritmos de enrutamiento, las operaciones de transferencias, etc. A continuación se describen las mejoras que se han desarrollado e implementado en el sistema (o están en proceso de desarrollo).

Diseño más robusto de los agentes

Los agentes de PhEDEx no mantienen localmente su estado interno, y se pueden parar o reiniciar sin que esto implique consecuencias negativas, incluso después de un fallo del sistema. El estado permanente de los flujos de trabajos se guarda en el blackboard, y las modificaciones de esta información registrada se hace mediante operaciones seguras. Tanto los agentes como los centros están limitados a cambiar partes relevantes de la base de datos mediante una estricta política de permisos en las operaciones con esta base de datos. Esto reduce el riesgo de daños.

Se está creando un entorno más dinámico, donde los agentes colaboran para proporcionar servicios solicitando responsabilidades para desarrollar tareas. Cada agente se hace responsable de una operación o de un paso en el flujo de trabajo (la transferencia de un fichero, chequear el estado del stager, gestionar la partición de la red en un centro, etc.) Cada agente encuentra las tareas pendientes en el blackboard, las escoge y ordena por prioridad, y las ejecuta. Cuando las tareas están completas con éxito las marca en la base de datos, y posiblemente asigna indirectamente otras tareas a otros agentes para continuar con el proceso completo.

Las tareas se juntan para formar un flujo de trabajo mediante "acoplamientos" (por ejemplo, la escritura y la lectura de la información de estado en y desde la base de datos por parte de dos agentes). La definición de qué información de estado está disponible al comenzar un proceso, y cuál debería estarlo al final, es la base del diseño de los agentes, que actúan sólo cuando aparece o cambia cierta información de estado.

Por otra parte, dado que muchas de las funcionalidades que se desean para los sistemas de almacenamiento (copia de cinta a disco bajo demanda y agrupamiento inteligente de las peticiones de estas copias, gestión más sofisticada del espacio, etc.) aún no están implementadas, los flujos de trabajos de PhEDEx se han hecho más sofisticados e incorporan algunas de estas funcionalidades. Los flujos de trabajos incorporan sub-tareas como el "pre-borrado", transferencias, operaciones de chequeo, etc. Los agentes de exportación han adquirido las funcionalidades que deberían tener los sistemas de almacenamiento.

Optimización del acceso a la base de datos

Se ha comprobado que un acceso directo a los recursos de bases de datos es esencial para conseguir un buen rendimiento. Se ha descartado el uso de servicios genéricos que añaden una capa extra de procesamiento e indirección, y se prefieren las optimizaciones manuales ya conocidas. Son las componentes de software, por tanto, las que interaccionan directamente con la base de datos en lugar de hacer uso de una capa intermedia que encapsula las operaciones. Con esto se consigue un mayor rendimiento en las operaciones. Pero se necesita entonces una base de datos de muy alta accesibilidad. La elección final fue usar un cluster de bases de datos Oracle con soporte 24/7 [9].

PhEDEx incluye funciones para las operaciones con bases de datos más comunes. A parte de hacer la programación más fácil, estas funciones permiten también la resolución de problemas en un entorno distribuido.

Se intenta mantener la disponibilidad de las bases de datos en todo momento. Sin embargo, es inevitable hacer intervenciones de vez en cuando (de mantenimiento, por ejemplo). Los agentes pueden programarse remotamente para que interrumpan o reanuden sus operaciones sin que esto suponga un problema.

Se puede hacer el sistema aún más robusto descentralizando algunas de las funcionalidades de la base de datos central. Uno de los principales candidatos es la descentralización de las tablas de enrutamiento.

Para comprobar si el acceso a la base de datos de TMDB es o no un cuello de botella en las operaciones de transferencia se ejercitaron una serie de tests mediante la simulación de transferencias ficticias (transferencias sin ficheros y que, por tanto, no requieren tiempo para su ejecución). Durante el test se simuló el comportamiento de cinco nodos pues en un entorno realista habrá múltiples centros intercambiando datos unos con otros. Las operaciones se organizaron de tal forma que cada centro comenzaba con un único conjunto de ficheros que se distribuyeron a los otros centros simulados para alcanzar el máximo número de transferencias en paralelo y, por tanto, la carga máxima en la base de datos. El test se repitió varias veces para acumular estadística. La figura 4.28 muestra los resultados de estos tests. Se consiguió un pico de, aproximadamente, 50000 copias por hora. Después de este pico se llegó a una tasa, sostenida durante varios días, de unas 30000 operaciones por hora. Teniendo en cuenta que las necesidades de CMS implican mover del orden de 175 TB de datos diarios [10] en ficheros de unos 2 GB, se estiman unas necesidades de, aproximadamente, unos 90000 ficheros por día. Estos tests de escala apuntan a que el sistema será capaz de satisfacer estos requisitos.

Código de defensa

Gradualmente se han ido implementando algoritmos cada vez más refinados, haciendo que el sistema sea cada vez más autónomo. Por ejemplo, algunos de los agentes más avanzados detectan patrones patológicos e interrumpen automáticamente su funcionamiento por si mismos. En estos casos, los patrones patológicos se identifican monitorizando algunas variables locales (como el número de transferencias por hora ejecutadas con éxito), y las acciones correctivas se inician cuando estas medidas superan, por exceso o por defecto, ciertos valores umbrales.

[9] 24 horas al día, 7 días a la semana.

[10] La cantidad total de 175 TB de datos diarios se reparte de la siguiente manera:

- 25 TB/día, unos 300 MB/s, en las transferencias de RAW+RECO desde el Tier-0 a los centros Tier-1.
- 125 TB/día, unos 1500 MB/s, en las transferencias de los datos filtrados para su análisis desde los centros Tier-1 a todos los centros Tier-2 (a razón de 60 MB/s para cada uno de los aproximadamente 25 centros Tier-2 existentes).
- 25 TB/día en las transferencias de los datos simulados Monte Carlo desde los centros Tier-2 a los centros Tier-1 (1 TB diario por cada centro Tier-2).

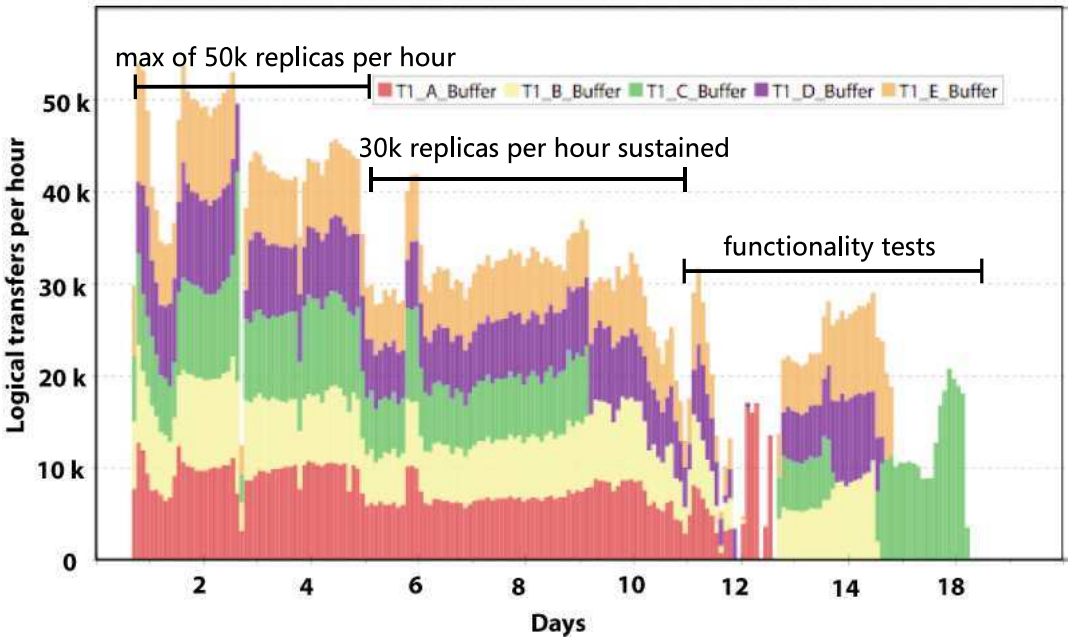


Figura 4.28: Tests de escala en el acceso a la base de datos del TMDB.

Optimización del enrutamiento: enrutamiento contractual

Con el enrutamiento *por contrato*, los agentes de enrutamiento solicitan el suministro de ficheros, y crean una solicitud formal de enrutamiento que tiene una cierta validez temporal. Entonces, los agentes en los nodos con réplicas estiman el coste de la transferencia y se ofrecen para la transferencia con esta información, indicando cuál es el siguiente nodo en la ruta. Los nodos intermedios hacen ofertas similares, y así sucesivamente hasta llegar al final. Finalmente, los agentes de enrutamiento escogen entre todas las ofertas basándose en los costes totales de cada ruta.

Otra necesidad que hay que tener en cuenta es la gestión de los fallos y las ineficiencias en las rutas. Una posibilidad es desechar una oferta de ruta completa si un nodo de esa ruta falla. Es necesario, por tanto, incluir en los costes de una cierta ruta una estimación razonable de la capacidad de los nodos para cumplir las peticiones (teniendo en cuenta situaciones como que un nodo tenga una gran cantidad de trabajo atrasados, etc.) También habría que evitar la continua oscilación entre dos rutas con fallos constantes.

Optimización del rendimiento

Aunque los agentes son *sin estado* con respecto al estado del flujo de trabajo global, algunos agentes crean cachés para mejorar las transferencias. Las cachés etiquetan los datos con una validez de varias horas. Pasado este tiempo, el registro se borra y, si sigue siendo necesario, se vuelve a extraer de la base de datos. Esto hace que los agentes sean *autocurativos*. Las cachés se usan sólo cuando el agente es la única fuente autorizada para los datos y, por tanto, sólo ha de reaccionar ante los cambios en la base de datos, y no por cambios en el estado de otros agentes.

Se puede conseguir un mejor ajuste del sistema teniendo en cuenta que los agentes son los mejor posicionados para monitorizar y reaccionar de forma activa a las condiciones locales. Controlan muchos de los parámetros de ajuste a pequeña escala (como el tamaño de las ventanas TCP, el tamaño de los bloques,

el número de transferencias en paralelo, etc.) Los agentes pueden notar cambios en las tasas de transferencias alcanzadas y modificar la demanda de enlaces de transferencia particulares. Pueden así establecer objetivos con ciertos límites, reaccionar de forma activa cuando se cumplen los objetivos, o cuando no se alcanzan durante un largo periodo de tiempo, y avisar a los mánagers cuando las operaciones se degradan significativamente.

Finalmente, se ha propuesto chequear el uso nuevas tecnologías para mejorar algunas operaciones. Por ejemplo, el uso de hardware y enlaces dedicados (como el *LCG Robust Transfer Service Challenge* que usa enlaces punto a punto de 10 Gbits Starlight, un cluster Itanium de 10 nodos duales en el CERN y hardware similar dedicado en los centros remotos). También se ha propuesto el uso de otros protocolos de transporte, pero esto requiere la incorporación de la gestión de los servicios SRM. Todas estas alternativas están aún en fase de estudio.

Capítulo 5

Integración del sistema de computación Grid para CMS

En los últimos años se han llevado a cabo una serie de tests, de escala y complejidad crecientes, con la finalidad de ejercitar los sistemas de computación Grid. Durante estos periodos se han puesto a prueba los flujos de datos y los flujos de trabajos de procesamiento de datos. Se han podido realizar medidas útiles de rendimiento, se han identificado problemas, y se han proporcionado datos que han permitido mejorar significativamente el diseño, integración y operación de los sistemas de computación.

Algunos de estos tests son específicos de CMS, pensados para establecer un sistema de trabajo distribuido que implemente de forma satisfactoria el modelo de computación del experimento, basado en diferentes arquitecturas Grid. Forman parte de estos tests específicos la serie de los *Data Challenges* (DC), y la de los *Combined Software and Analysis Challenges* (CSA). El objetivo principal de los DC es validar el modelo de gestión de datos del experimento, mientras que en los CSA se pretende comprobar el buen funcionamiento de todas las componentes del modelo de computación, desde la distribución de los datos hasta las diferentes tareas de procesado y análisis que se ejecutan sobre ellos.

Otros tests son de propósito más general, independientes de los experimentos, pensados para chequear los servicios e infraestructuras desplegados por LCG. La serie de los *Service Challenges* (SC) forma parte de estos ejercicios.

Los centros españoles PIC y CIEMAT han tomado parte activa y destacada en todos estos tests desarrollados en los últimos 3 años [119, 120]. Ambos centros han colaborado ofreciendo todos sus recursos de computación, tanto de almacenamiento como de cálculo, y los recursos humanos disponibles. En particular, ha sido muy relevante la contribución al Data Challenge 2004 (DC04) [121], el Service Challenge 3 (SC3) [122] y el Combined Software and Analysis Challenge 2006 (CSA06) [123, 124]. La complejidad y escala de estos ejercicios se ha ido incrementando progresivamente. Durante el DC04 se simuló un escenario equivalente a un 25 % de la tasa de adquisición de datos a la luminosidad de inicio del LHC (lo que equivale a un 5 % durante el régimen de mayor luminosidad). Las operaciones del CSA06 simularon unas condiciones equivalentes al 25 % de la capacidad requerida durante las primeras tomas de datos.

En todos los ejercicios se ha comprobado que un contacto directo con los centros, al igual que un cierto nivel de compromiso por su parte, es esencial para conseguir la ejecución eficiente de las actividades de computación. La naturaleza distribuida del modelo de computación de CMS y de los recursos obliga a los centros, con la ayuda de los grupos de expertos de CMS, a involucrarse activamente en la integración de los sistemas de computación, ajustarlos para conseguir el mejor rendimiento de los recursos, y reaccionar rápidamente en caso de problemas.

Los tests de escala continuan siendo una actividad extremadamente importante. Doblar la capacidad

y el nivel de actividad en cada etapa no es trivial. Los problemas asociados a la escala son algunas veces difíciles de prever, y los ejercicios concebidos para alcanzar unos ciertos objetivos de escala son de importancia crucial. Estos ejercicios tendrán continuidad durante el año 2007 en un nuevo challenge combinado, el CSA07. Este nuevo ejercicio está programado para septiembre del 2007 con el objetivo de ejercitar simultáneamente todas las actividades de computación y off-line a una escala equivalente al 50 % de las necesidades durante el primer año de toma de datos.

5.1. Data Challenge 2004

El DC04 tuvo lugar durante los meses de marzo y abril de 2004, con el propósito de probar la capacidad del modelo de computación de CMS en un régimen equivalente al 25 % de las necesidades durantes las primeras operaciones. Los principales objetivos del DC04 fueron:

- Simular una frecuencia de 25 Hz de toma de datos a una luminosidad de $2x10^{33}$ $cm^{-2}s^{-1}$ durante un mes (aproximadamente 50 millones de sucesos). Esta fase se llevó a cabo en dos etapas: una primera de producción pre-challenge y una segunda de reconstrucción sobre estos datos producidos para simular la toma de datos y proporcionar datos para los análisis físicos.
- Transferir los ficheros de datos producidos a los centros Tier-1.
- Ejecutar tareas de análisis en los centros Tier-1 sobre los datos transferidos tan pronto como éstos estén disponibles (análisis en "tiempo real").
- Ejecutar tareas de análisis de los usuarios en los centros Tier-1 y Tier-2 tan pronto como los datos se hacían públicos a la comunidad de físicos de CMS.

Se simularon 30 millones de sucesos durante la fase de pre-challenge, fueron reconstruidos en el Tier-0, y se distribuyeron a los seis centros Tier-1 que participaron en el DC04 (CNAF, FNAL, GridKA, IN2P3, RAL y PIC). Esta fue la primera vez que el PIC formaba parte, oficialmente, del conjunto de centros Tier-1 de CMS. Los datos reconstruidos fueron distribuidos desde algunos Tier-1 a ciertos centros Tier-2 seleccionados para ejecutar sobre ellos las tareas de análisis offline. Para gestionar la distribución automática de los datos desde el Tier-0 a los Tier-1, CMS puso en marcha durante el DC04 su sistema de gestión de datos, PhEDEx (descrito en la 4.3). Los centros españoles, CIEMAT y PIC, tomaron parte activa en el desarrollo e implementación de PhEDEx durante el DC04.

Se desarrolló un sistema para ejecutar el análisis sobre los datos en tiempo real en los Tier-1 a medida que éstos iban llegando. Este análisis en tiempo real proporciona una rápida retroalimentación a la reconstrucción (calibración, alineamiento, validación del software de reconstrucción, etc.) Uno de los parámetros más importantes que se midió durante el DC04 fue el tiempo transcurrido desde que los datos estaban disponibles para ser transferidos desde el Tier-0 hasta que eran analizados en los Tier-1.

La participación de los centros españoles en el DC04 fue muy significativa. El análisis en tiempo real en el PIC se demoraba solamente unos 20 minutos desde que los datos estaban disponibles para su distribución hasta que los trabajos de análisis eran enviados en el Tier-1. Algunas de estas muestras se replicaron en el CIEMAT para ejecutar sobre ellas trabajos de análisis offline.

5.1.1. Distribución de datos durante el Data Challenge 2004

Tanto el CIEMAT como el PIC instalaron la última versión del software de LCG-2 como parte del grupo de producción MC Grid. La tabla 5.1 agrupa los recursos de almacenamiento que se desplegaron en los centros españoles para su participación en el DC04. La infraestructura se completó con la instalación en el PIC de dos User Interfaces, un Resource Broker y unos 150 Worker Nodes. En las dos UI se ejecutaban los agentes de software responsables de gestionar las transferencias de ficheros desde el Tier-0, la replicación a los SE de CIEMAT y PIC, y el análisis en tiempo real en el PIC.

Centro	Tipo de SE	Capacidad	Propósito
PIC	CASTOR	3 TB de disco, 20 TB en cinta	Recibir los datos del Tier-0 y guardarlos en forma segura.
PIC	Clásico	1 TB de disco	Proporcionar acceso a los datos para el análisis en tiempo real.
CIEMAT	Clásico	200 GB de disco	Recibir las réplicas de las muestras seleccionadas desde el PIC para el análisis offline.

Tabla 5.1: Infraestructura de almacenamiento desplegada en los centros españoles para el DC04.

La figura 5.1 muestra el ancho de banda disponible (y valores aproximados de latencia) de las diferentes secciones de red desde el CERN hasta los centros españoles que estaban disponibles durante el transcurso del DC04. Los ficheros de datos eran transferidos desde el Tier-0 hasta los centros españoles a través de las redes académicas y de investigación GEANT, RedIRIS y Anella Cientifica. La tabla 5.2 compila las características principales de estas redes en el momento del DC04. La latencia total entre el Tier-1 y el Tier-2 era de 9 ms, y entre el Tier-0 y el Tier-2 era de 35 ms.

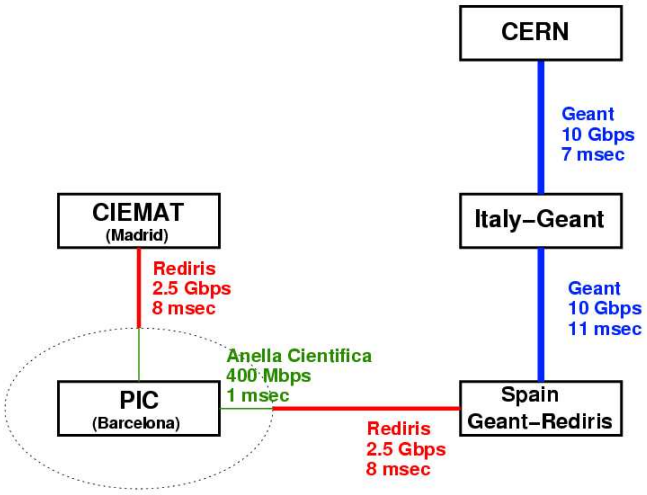


Figura 5.1: Conexiones de red entre el CERN y los centros españoles.

Red	Ámbito	Ancho de banda	Latencia
GEANT	Europeo	10 Gb/s	18 ms
RedIRIS	España	2.5 Gb/s	8 ms
Anella Cientifica	Cataluña	400 Mb/s	1 ms

Tabla 5.2: Infraestructura de red desde el CERN hasta los centros españoles para el DC04.

La figura 5.2 muestra los valores de las operaciones de transferencia de ficheros desde el Tier-0 al Tier-1 durante el período del DC04. Se transfirieron 446652 ficheros (aproximadamente 6 TB). Este flujo de datos fue bastante irregular, con una tasa de transferencia media de unos pocos MB/s, debido a que los datos no se hacían disponible para su transferencia en el Tier-0 de forma regular.

La figura 5.3 muestra el estado de los ficheros asignados al PIC en función del tiempo. Los puntos marcados

como "IN_BUFFER" muestran el número de ficheros que estaban en espera para ser transferidos al Tier-1. Se puede ver que, en general, este número era significativamente pequeño y no hubo demoras en las transferencias de ficheros del CERN al PIC durante todo el DC04. En general, los agentes de transferencia del PIC funcionaron de forma bastante estable. Sólo aparecieron problemas, en rara ocasión, durante la interacción con el TMDB (la base de datos de PhEDEx, ver sección 4.3.1). Cuando el tiempo transcurrido desde el último contacto con el TMDB superaba un cierto valor predeterminado el agente se reiniciaba automáticamente.

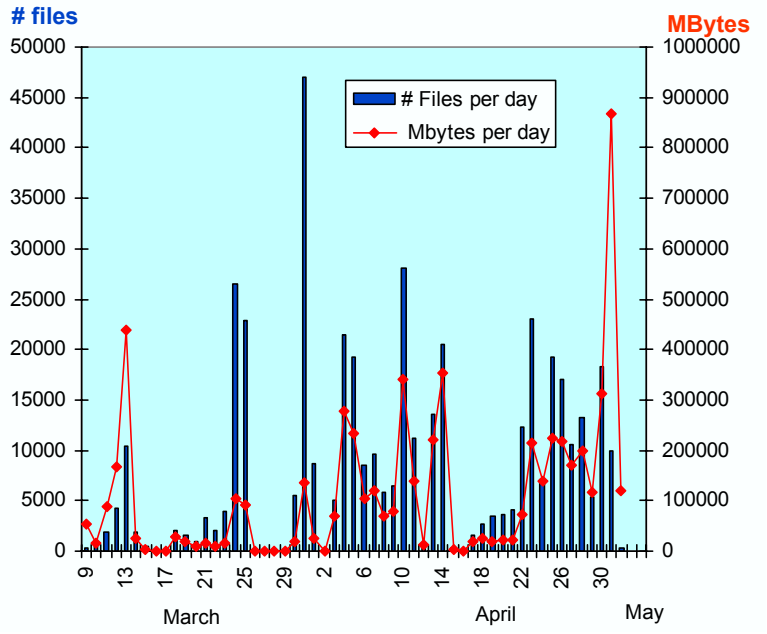


Figura 5.2: Tranferencias de ficheros desde el Tier-0 al Tier-1 del PIC.

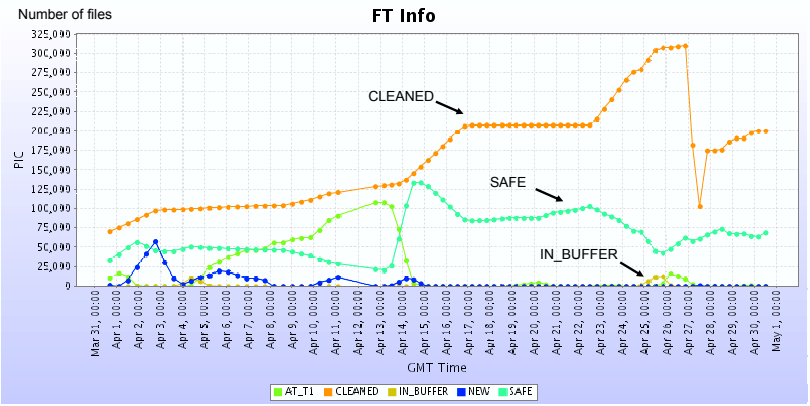


Figura 5.3: Estado de los ficheros transferidos al PIC durante el DC04.

Una de las principales causas de ineficiencia durante el DC04 fue el pequeño tamaño de los ficheros gestionados por el sistema. El tamaño medio era inferior a los 500 kB, lo que provocaba la aparición de retardos significativos en las operaciones de transferencia y almacenamiento en el SE. Estos retardos se deben al tiempo muerto que introduce cada interacción con el sistema de transferencia o de almacenamiento. A

esto había que sumar los retardos en las consultas a los catálogos (independientemente del tamaño del fichero). Estos retardos fueron compensados, en la medida de lo posible, mediante la paralelización de las transferencias gracias a las mejoras introducidas en los agentes de transferencia. Otra solución investigada fue el empaquetamiento de los ficheros de datos en archivos de gran tamaño (superior a 1.6 GB) antes de su transferencia al Tier-1. Salvo este problema, y el inherente al manejo de ficheros de pequeño tamaño, el sistema CASTOR del PIC no mostró ningún problema.

Otro de los objetivos del DC04 fue forzar el sistema de distributión de datos, en particular la red, mediante un incremento significativo de la tasa de transferencia desde el Tier-0. La tasa media de transferencia fue de unos 240 Mb/s, alcanzando picos de 320 Mb/s (compatible con los 400 Mb/s disponibles en la red de acceso al PIC). Se alcanzó un valor de 3345 ficheros transferidos al PIC en diez horas. La figura 5.4 muestra el tráfico de red durante este ejercicio de estrés del sistema. Al comienzo del test los dos SE recibieron datos a una velocidad acumulada de 200 Mb/s. Finalmente se optó por terminar el ejercicio haciendo uso de sólo el segundo de ellos debido a ciertos problema con la tarjeta de red del primero. Durante este segundo periodo, con sólo uno de los SE operativo, la tasa de recepción de datos fue de 240 Mb/s.

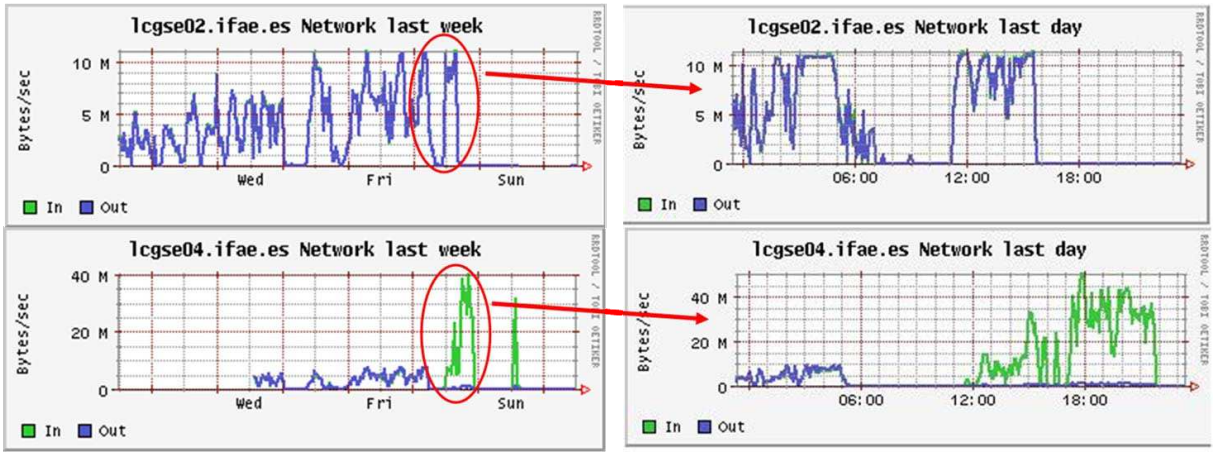


Figura 5.4: Tasa de transferencia durante el test de estrés de la red. Se usaron dos SE (arriba y abajo) para aumentar el tráfico en la red al máximo. Las figuras de la derecha muestran una imagen ampliada de los intervalos de tiempo marcados en las figuras de la izquierda.

El catálogo global de réplicas de ficheros (RLS) fue uno de los elementos más débiles durante todo el DC04. En general, la inserción de nuevas entradas en el catálogo fue una operación demasiado lenta, causando retrasos para el registro de los ficheros del Tier-0 y de las réplicas en el Tier-1. Las operaciones de consulta también necesitaban una gran cantidad de tiempo para ejecutarse. Las consultas al catálogo de metadatos fueron especialmente lentas, en parte a causa de la arquitectura de RLS, que separaba el catálogo de réplicas locales (LRC) y el catálogo de metadatos de las réplicas (RMC) en dos bases de datos diferentes. Para realizar una consulta basada en metadatos había que consultar primero al RMC, y luego correlacionar la información con la que proporcionaba el LRC. Además, la carencia de algunas funcionalidades básicas del servicio RLS impedía, por ejemplo, la ejecución de bulk operations.

El equipo del RLS implementó varias soluciones (paralelismo, herramientas basadas en EDG, sustitución de programas JAVA por otros desarrollados en C,...) que consiguieron incrementar en un orden de magnitud la velocidad de las operaciones de registro y consulta. Sin embargo, RLS se mostró como un elemento crítico, y hoy en día ya no forma parte del Sistema de Gestión de Datos de CMS.

Uno de los logros del ejercicio fue la implementación de forma satisfactoria del TMDB, que luego se

convertiría en la base de datos central de sistema de transferencia de datos de CMS.

5.1.2. Análisis de datos en tiempo real

Uno de los principales objetivos del DC04 fue implementar un sistema de análisis en tiempo real en los centros Tier-1 que demostrase la capacidad de CMS para analizar los datos a medida que estaban disponibles. De nuevo, las herramientas proporcionadas por LCG fueron utilizadas para esta tarea.

La figura 5.5 muestra el esquema del sistema de análisis en tiempo real en el PIC y su acoplamiento con el sistema de distribución de datos. A igual que ocurre con la distribución de los datos, el sistema de análisis en tiempo real se basó en un conjunto de agentes independientes de funcionalidad específica que se comunicaban entre sí a través de una instancia local del TMDB. Para acelerar el acceso de los trabajos de análisis a los datos en el Tier-1, un agente de replicación los copiaba previamente a un SE de disco. Simultáneamente se transferían los datos seleccionados al Tier-2 CIEMAT para su análisis offline.

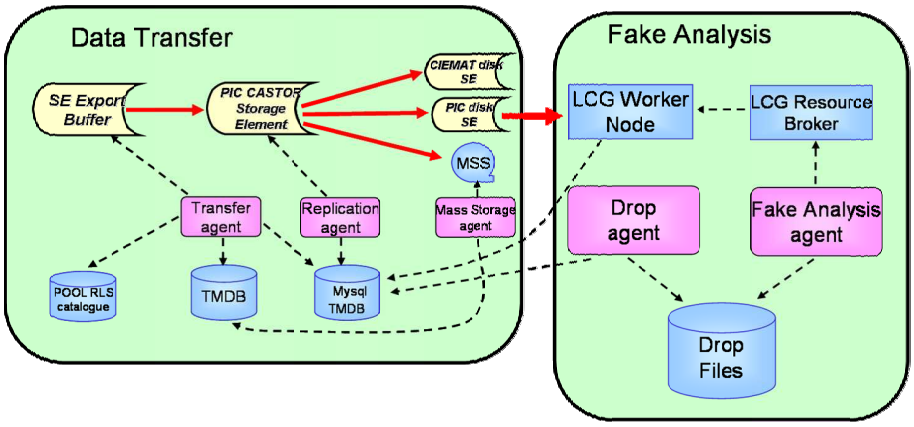


Figura 5.5: Proceso de transferencias de ficheros desde el Tier-0 al sistema de almacenamiento del Tier-1, y su acoplamiento con el sistema de análisis en tiempo real, durante el DC04.

Cuando la copia al SE de disco está completa se marca el fichero como disponible para el análisis en el TMDB. Un nuevo agente (el *Drop agent*) consulta esta base de datos de forma periódica para averiguar si hay nuevos ficheros. Un trabajo de análisis necesita un conjunto mínimo de ficheros. El agente drop, basándose en los atributos de metadatos del grupo de ficheros, determina cuándo un conjunto de ficheros está completo y localmente accesible. Si se dan las circunstancias adecuadas, inicia la preparación y envío de los trabajos de análisis. Esto se hace colocando en un directorio determinado un fichero que contiene los GUIDs y PNFS del conjunto de ficheros que se van a analizar. Un segundo agente (el *Fake Analysis agent*) comprueba periódicamente este directorio en busca de nuevos ficheros. Este agente, además, basándose en los GUIDs y PFNs del conjunto de ficheros solicitados por un trabajo de análisis, prepara un catálogo XML con los metadatos correspondientes obtenidos del RLS. El agente genera el fichero de configuración necesario para el ejecutar el software de análisis y el fichero JDL para enviar el trabajo usando los recursos Grid. El envío de los trabajos se llevó a cabo a través del RB del PIC y de un RB desplegado en el CERN específicamente para el DC04. En cualquier caso, los trabajos se ejecutaron siempre en el PIC. El agente de análisis guardaba en el TMDB local la hora y fecha del envío de los trabajos de análisis.

Se mandaba al Grid un programa *wrapper*[1] encargado de ejecutar el trabajo de análisis propiamente dicho. Este programa se encarga de configurar adecuadamente el entorno para el trabajo de análisis, descarga los metadatos desde un SE, y comprueba si se pueden acceder directamente desde el WN a los ficheros de datos a través de rfio. Si los ficheros no son accesibles directamente el programa se encarga de

[1] programa que controla el acceso y ejecución de otro programa.

copiarlos desde el SE al disco del WN. El programa también se encarga de copiar el software de análisis, previamente replicado en el SE de cada uno de los centros del LCG que lo solicite. Finalmente, la fecha y hora en la que comienza y termina el programa de análisis quedaban registradas en el TMDB local.

Los análisis que se ejecutaron eran simulados pues no se realizaron análisis físicos reales (este tipo de análisis se conoce usualmente como *fake analysis*). En su lugar, simplemente se chequeaba la integridad de los ficheros (leyendo su contenido). Los trabajos de análisis se ejecutaron sobre todo tipo de Datasets. El principal objetivo de este ejercicio fue demostrar que los datos podían analizarse localmente en tiempo real y medir algunos parámetros importantes relacionados con el rendimiento de estas operaciones. No fue necesario ejecutar trabajos de monitorización, y fue suficiente con procesar la información que los agentes iban registrando en la base de datos.

Uno de los parámetros más importantes que se deseaba medir durante el DC04 fue el tiempo de respuesta desde que los datos se hacían disponibles en el Tier-0 para su distribución hasta que se enviaban los trabajos de análisis para su procesamiento en los Tier-1. El PIC fue el Tier-1 que mostró los mejores resultados de esta medida, como se puede ver en la figura 5.6. La media de este tiempo de respuesta fue de unos 20 minutos, y el mínimo fue de 5 minutos, aproximadamente.

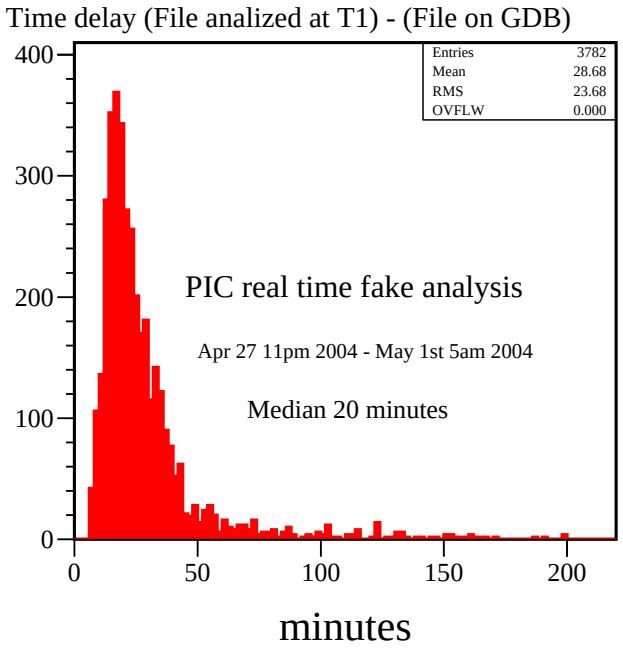


Figura 5.6: Tiempo de respuesta transcurrido desde que los datos estaban disponibles para su transferencia hasta el envío de los trabajos de análisis en el Tier-1 durante el DC04. Incluye la operación de copia de los ficheros en el CERN desde el *Global Distribution Buffer* (donde son escritos por los trabajos de reconstrucción) hasta el *LCG Export Buffer*, la replicación al SE CASTOR del PIC, la operación de copia la SE de disco y la preparación y envío de los trabajos de análisis.

La figura 5.7 muestra el desglose del tiempo mostrado en la figura 5.6. La figura superior izquierda muestra el tiempo necesario para la transferencia de los ficheros desde el GDB (*Global Distribution Buffer*) en el CERN al SE CASTOR del PIC. Esta operación requería, en media, 13 minutos. En la figura superior derecha se puede ver que el tiempo necesario para la replicación de los ficheros al SE de disco era siempre inferior a un minuto. Las figuras inferiores muestran el tiempo necesario para la preparación y envío de los trabajos de análisis (izquierda) y desde la submisión hasta el comienzo de la ejecución (derecha). La preparación de los trabajos necesitaba aproximadamente 1.5 minutos, y el envío unos 3 minutos. Estos

3 minutos se consumen en la interacción con el RB, el envío del trabajo desde el RB a un centro, y el procesamiento local por parte del sistema de colas de ese centro.

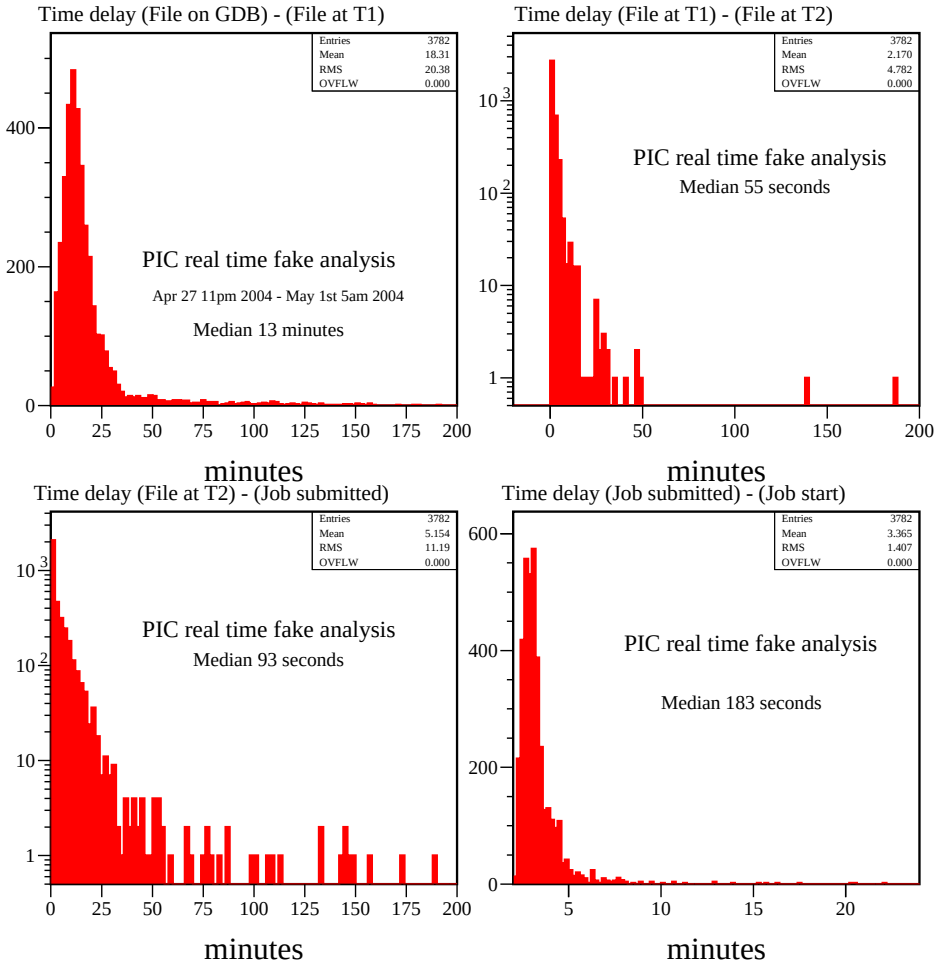


Figura 5.7: Desglose por etapas del tiempo transcurrido entre que los datos estaban disponibles para su transferencia hasta el envío de los trabajos de análisis en el Tier-1 durante el DC04.

El test de análisis fue ejecutado en el PIC en la última etapa del DC04, durante 78 horas sin interrupción. En este tiempo se ejecutaron unos 2000 trabajos de análisis. La distribución temporal de trabajos ejecutados se puede ver en la figura 5.8. Los huecos que aparecen en la figura se debieron a la falta de datos disponibles en el CERN.

Finalmente, se replicaron algunas muestras seleccionadas al Tier-2 del CIEMAT. Aunque no pudieron analizarse, debido a algunos problemas con la versión del software, la cadena de transferencia de datos Tier-0 → Tier-1 → Tier-2 quedó bien establecida.

5.1.3. Experiencia

Durante los ejercicios del DC04 las principales dificultades aparecieron en la gestión de los ficheros de metadatos. Como catálogo de metadatos se usaba RLS. Como RLS demostró ser lento en las consultas a los metadatos, durante el DC04 se probaron varias soluciones para evitar su uso en la medida de lo posible: usar la base de datos TMDB para guardar los ficheros de metadatos, incluir el LFN en los

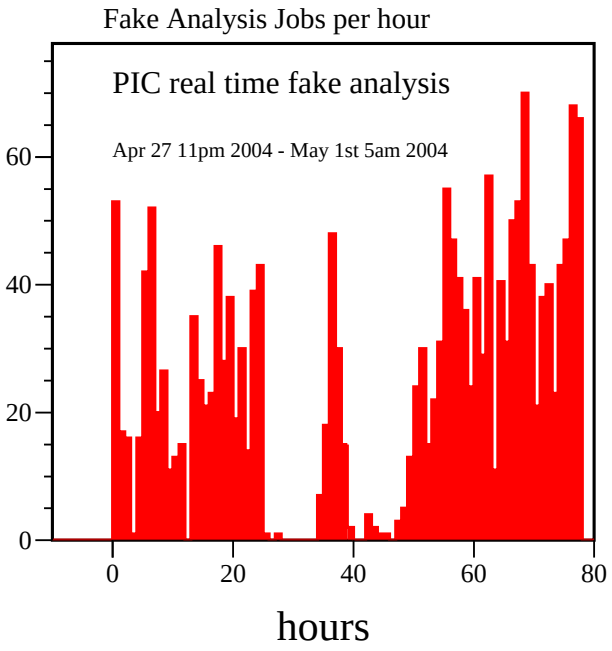


Figura 5.8: Trabajos de análisis ejecutado en el Tier-1 durante el DC04.

metadatos, consultar los metadatos a través de su GUID, usar ficheros XML locales, etc. Finalmente CMS decidió prescindir de RLS como catálogo global, y el uso del TMDB se ha generalizado al ser parte fundamental de sistema de transferencia de datos, PhEDEx, que vio su nacimiento durante el DC04.

También durante el DC04, al igual que ocurrió durante las primeras fases de producción MC en Grid, se comprobó que las operaciones con ficheros de pequeño tamaño son muy ineficientes. Estas operaciones incluyen las transferencias, las consultas a los catálogos, el almacenamiento, etc.

En general, el manejo de los ficheros de metadatos fue difícil y poco eficiente. El hecho de que los metadatos tuviesen que actualizarse a medida que los datos iban llegando hacía díficil el análisis en tiempo real. Actualmente CMS ya no usa ficheros de metadatos.

5.2. Service Challenge 3

El LCG Service Challenge 3 tuvo lugar durante la segunda mitad del 2005. En CMS se tomó como un test de integración para ejercitar la parte del modelo de computación relativa al procesamiento de datos a gran escala bajo condiciones realistas. Se centró en la validación de las infraestructuras para la transferencia, almacenamiento y suministro de los datos, junto con las componentes necesarias para el envío y monitorización de los trabajos. Además, se puso en juego todo el software de CMS necesario para la transferencia y publicación de datos y para los trabajos de análisis.

Los objetivos originales del LCG SC3 fueron:

- Flujo de datos estructurado. Distribución de datos desde el Tier-0 en el CERN a los centros Tier-1, envío de los datos filtrados a los centros Tier-2, y transferencia de los datos resultado de la producción Monte Carlo desde los centros Tier-2 a los centros Tier-1 asociados.
- Procesamiento masivo de datos en paralelo a las transferencias de datos. Publicación automática de los datos a medida que llegan a los centros, haciéndolos disponibles para el análisis, ejecución

de programas de filtrado en los centros Tier-1 y producción Monte Carlo en los Tier-2, y acceso a los datos almacenados para probar la capacidad de lectura masiva de datos. Estos objetivos fueron redefinidos a la vista de los resultados obtenidos en una primera fase. Se comprobó que algunas componentes no estaban suficientemente preparadas para llevar a cabo tests significativos en una situación de estrés del sistema. La producción Monte Carlo dejó de considerarse un objetivo, y los programas de filtrado fueron sustituidos por trabajos de análisis más simples.

Para intentar alcanzar estos objetivos, el LCG SC3 se desarrolló en dos fases:

- Julio de 2005: primera etapa para comprobar la capacidad de transferencia de datos entre centros en condiciones de máxima frecuencia de transferencia. Se desactivaron las transferencias de datos de CMS gestionadas con PhEDEx durante esta fase para no interferir con las pruebas del SC3.
- De Septiembre a Noviembre de 2005: segunda etapa donde se ejercitaron los servicios de computación en paralelo a las transferencias de datos a baja frecuencia.

5.2.1. Configuración de los recursos

Los recursos que se dedicaron en los centros españoles incluyen una combinación de hardware y servicios dedicados, aparte de los recursos de computación de producción previamente instalados. Un Storage Element dedicado en el PIC junto con las instancias SC3 de los servicios de almacenamiento. Los recursos de computación y de red fueron compartidos con las actividades de producción. En el CIEMAT se instalaron los agentes de transferencia y los catálogos de datos, y todos los recursos fueron compartidos también con las actividades de producción. La figura 5.9 muestra la infraestructura de red utilizada por los centros españoles durante el LCG SC3. La tabla 5.3 compila los valores de ancho de banda y latencia en las diversos tramos de esta red. Las únicas diferencias respecto al DC04 son el aumento del ancho de banda de la red Anella y el paso por Francia en lugar de Italia para las transferencias de datos desde el CERN.

Red	Ámbito	Ancho de banda	Latencia
GEANT	Europeo	10 Gb/s	15 ms
RedIRIS	España	2.5 Gb/s	8 ms
Anella Cientifica	Cataluña	1 Gb/s	1 ms

Tabla 5.3: Infraestructura de red desde el CERN hasta los centros españoles para el SC3.

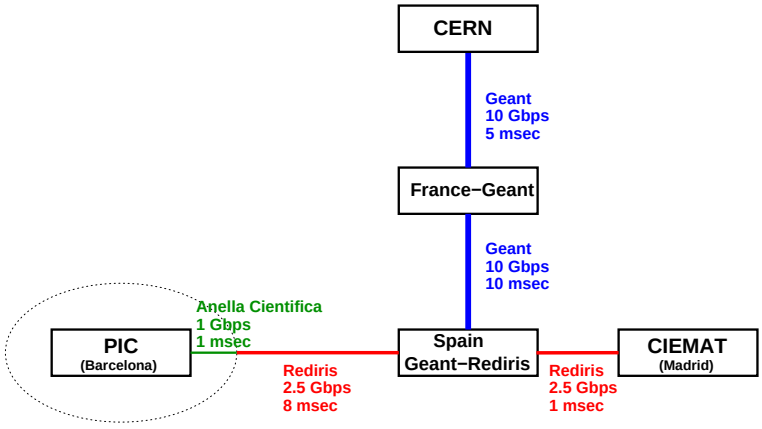


Figura 5.9: Infraestructura de red para el SC3.

La tabla 5.4 resume los recursos de almacenamiento desplegados por los centros españoles. Tanto en el PIC como en el CIEMAT se dedicaron sendos servicios de CASTOR. En el caso del PIC se configuraron dos pools de discos CASTOR distintos, como se puede ver en la figura 5.10. Un pool de disco se dedicó a operaciones de importación/exportación de datos y el otro para tareas de análisis. El primero se implementó con discos de alto rendimiento y redundantes para poder aceptar todos los datos procedentes del CERN, exportar los datos al CIEMAT, y ejecutar los trabajos de publicación. Para optimizar el acceso, este pool se configuró como un único punto de entrada SRM externo, el cual daba acceso a 4 servicios SRM internos que se escogían de forma cíclica. Para lograr un alto rendimiento de lectura el pool de datos para análisis estaba altamente distribuido, formado por los discos locales de unos 50 WNs. Este pool se alimentaba con datos procedentes de cinta para evitar interferencias en el pool de importación/exportación de datos.

No se dedicaron recursos de cálculo de forma exclusiva para el SC3. Se usaron unos 160 WNs (equivalentes a unos 200 kSI2k) en el PIC, compartidos con las tareas de producción MC y las actividades de otras VOs, y unos 120 WNs (120 kSI2k) en el CIEMAT, también compartidos con la producción MC.

Centro	**Tipo de SE**	**Capacidad**	**Propósito**
PIC	CASTOR	15 TB en disco y 15 TB en cinta	Recibir los datos del Tier-0 y guardarlos en forma segura.
CIEMAT	CASTOR	5 TB de disco	Recibir las réplicas de las muestras seleccionadas desde el PIC para el análisis offline.

Tabla 5.4: Infraestructura de almacenamiento desplegada en los centros españoles para el SC3.

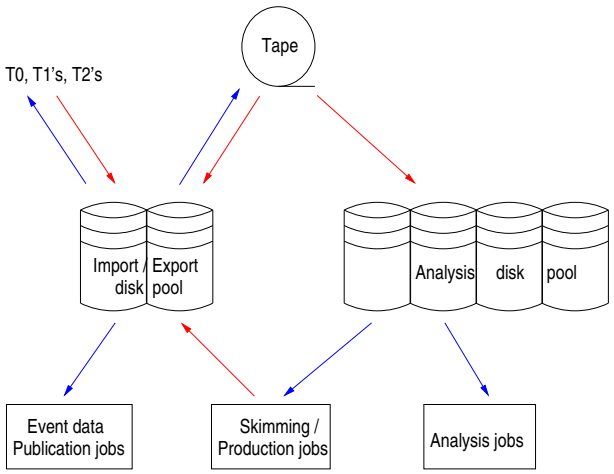


Figura 5.10: Configuración de los recursos de almacenamiento dedicados en el Tier-1 del PIC para el SC3.

La figura 5.11 muestra los servicios de computación y los flujos de datos y de trabajos durante el LCG SC3. Los datos se transferían desde el CERN a los Tier-1, y de éstos a los centros Tier-2 asociados. Los datos fueron ubicados en SEs y archivados en cinta en los Tier-1. Estos datos se distribuyeron en archivos zip de 2 GB para minimizar el número total de ficheros manejados y aumentar el rendimiento en las operaciones de transferencia y almacenamiento. Los datos fueron agrupados en Datasets, cada uno de los cuales contenía los Data Tiers correspondientes a los pasos de simulación, digitalización y reconstrucción Monte Carlo. Los distintos Data Tiers se distribuyeron por separado.

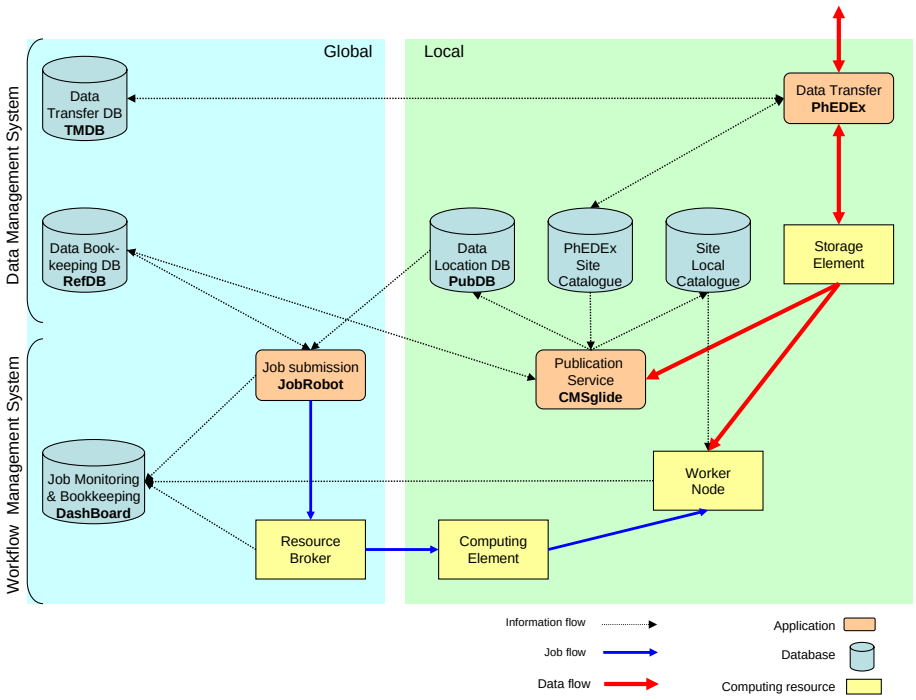


Figura 5.11: Servicios y flujos de datos y de trabajos durante el LCG SC3.

Una vez que se transfiere una Data Collection completamente, ésta debe publicarse para permitir su análisis. Este proceso incluye la creación de los metadatos y los catálogos con los LFNs, PFNs y GUIDs correspondientes. Para llevar a cabo las tareas de publicación se hizo uso de la misma herramienta que originalmente se usaba en la producción MC, CMSGLIDE. Las muestras disponibles para el análisis se anunciaban en la base de datos de publicación, PubDB.

Un agente local de PhEDEx se encargaba de ejecutar estas tareas de publicación de forma automática. Una vez que se había transferido una colección completa el agente ejecutaba un programa local de publicación que, a su vez, invocaba al servicio de publicación. Los trabajos de publicación se enviaban al sistema de colas local para paralelizar el proceso, pues la creación de metadatos consume una gran cantidad de tiempo y de recursos de red. Tras la publicación se poblaba el pool de disco de análisis con los sucesos publicados procedentes de las cintas. Cuando se había copiado a este pool de análisis la colección completa de sucesos, ésta se publicaba en RefDB como disponible para el análisis. Se siguió esta estrategia para evitar ineficiencias en los trabajos de análisis que intentasen leer ficheros solo disponibles en cinta. El espacio disponible en este pool de análisis fue suficiente para albergar todas las muestras utilizadas en el SC3.

Un generador central de trabajos (JobRobot) estuvo en ejecución continuamente durante todo el LCG SC3 enviando trabajos de análisis a los centros. Para la monitorización y bookkeeping de las actividades se hizo uso de Dashboard. JobRobot mandaba información en el momento del envío de los trabajos, y tras recuperar el output de los mismos. A su vez, los propios trabajos enviaban información en tiempo real en los momentos de inicio y fin de los procesos de análisis.

5.2.2. Etapa de flujo de datos

Esta primera fase tuvo lugar en Julio de 2005. Después de este periodo se cancelaron las transferencias de CMS gestionadas por PhEDEx para no interferir con las operaciones FTS centrales del equipo de LCG (independientes de los experimentos).

Esta fase consistió en un ejercicio de transferencia continua de datos a la mayor frecuencia posible desde disco en el CERN a disco en los centros Tier-1. La figura 5.12 muestra la velocidad de transferencia de datos promedio de cada día durante esta fase para los distintos centros Tier-1. En el caso particular del PIC (figura 5.13) se consiguió una velocidad superior a 40 MB/s durante la mayor parte del tiempo que duró el test (dos semanas), alcanzando picos de unos 85 MB/s. Estos valores hacen referencia al volumen de datos de CMS transferidos con éxito. El tráfico de red total, incluyendo sobrecargas y transferencias fallidas, fue algo mayor.

En total se transfirieron 33 TB de datos del CERN al PIC durante esta fase. Se consiguió esta tasa mediante transferencias GridFTP, ejecutando 10 transferencias de ficheros en paralelo con 10 streams TCP cada una. Se observaron tasas de transferencia bajas para cada stream TCP (de 0.5 a 2 MB/s), a pesar de la optimización de los parámetros TCP de los servidores de disco. No se hizo uso de las funcionalidades de SRM pues eran aún poco fiables y de bajo rendimiento.

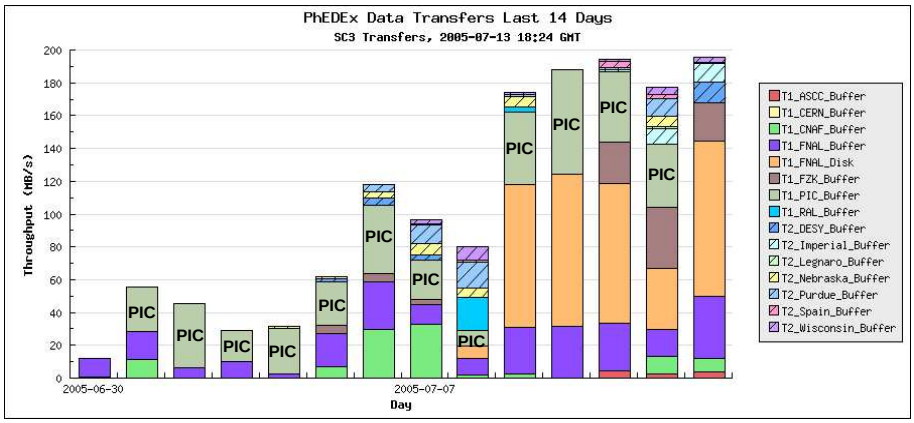


Figura 5.12: Tasa de transferencias entre el CERN y los Tier-1 durante el LCG SC3.

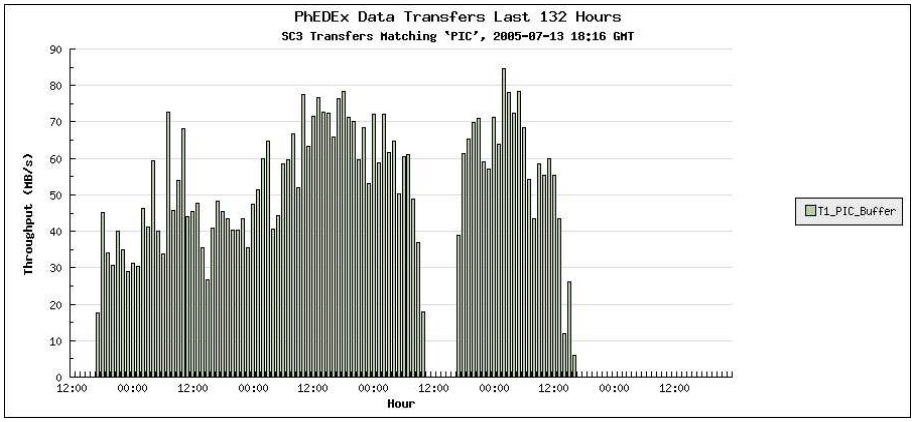


Figura 5.13: Tasa de transferencias entre el CERN y el PIC durante un periodo de 5 días en el LCG SC3.

Después de esta etapa se ejercitaron las transferencias entre los centros Tier-1 y sus Tier-2 asociados. Se alcanzó una tasa media de 18 MB/s entre el PIC y el CIEMAT durante 5 días, como puede verse en la figura 5.14. Se comprobó que el ritmo de transferencia venía limitado por la existencia de un único nodo CASTOR/GridFTP en el CIEMAT. Se consiguió duplicar este ritmo instalando un segundo nodo,

llegando prácticamente al límite del ancho de banda disponible en el CIEMAT (limitado a 320 Mb/s por un firewall[2]).

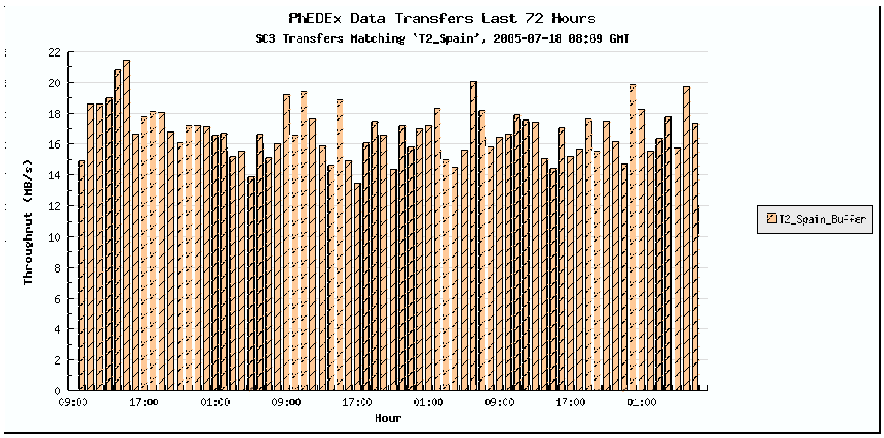


Figura 5.14: Tasa de transferencias entre el PIC y el CIEMAT durante un periodo de 3 días en el LCG SC3.

5.2.3. Etapa de servicios

Esta segunda fase tuvo lugar entre Septiembre y Noviembre de 2005. Se llevaron a cabo transferencias de datos a baja velocidad junto con la ejecución en paralelo de la publicación automática de los datos en los centros y el envío de trabajos de análisis.

Durante esta fase se ejecutaron transferencias de datos SRM mediante clientes de *srmcp*, a diferencia de la primera fase donde las transferencias GridFTP fueron ejecutadas mediante los comandos básicos del middleware de LCG (como *globus-url-copy*, por ejemplo). Al comienzo de esta segunda fase se transfirieron al PIC unos 3 TB de datos para validar el sistema de publicación automática y proporcionar datos para ser procesados por los trabajos de análisis. Durante las dos últimas semanas de esta fase, y una vez que el procedimiento de publicación automática fue establecido, la transferencia de datos aumentó gradualmente, incrementando la velocidad hasta un valor estable de unos 2-3 TB diarios. Esta evolución puede verse en la figura 5.15. En total se copiaron 13 TB de datos del CERN al PIC en esta fase, con un ritmo promedio de 11.4 MB/s, y una eficiencia media del 62 %. Estos dos valores aumentaron notablemente durante la última semana de la etapa de servicios, una vez que se entendieron y corrigieron ciertos problemas detectados con SRM y CASTOR. Se llegó entonces a una velocidad de 25 MB/s, con una eficiencia media del 76 %. Desde el PIC se replicaron al CIEMAT 2 TB de datos, a un ritmo medio de 6.5 MB/s, con una eficiencia del 63 %.

Se publicaron en total, entre el PIC y el CIEMAT, 5 millones de sucesos repartidos en 90 Event Collections. A medida que los datos se hacían disponibles se enviaban trabajos de análisis para procesarlos. Se ejecutaron, aproximadamente, 2000 trabajos en el PIC y 6000 en el CIEMAT. La figura 5.16 muestra los diferentes códigos de retorno de los trabajos y la distribución de errores en función del tiempo para el PIC y el CIEMAT, respectivamente. En ambos sitios se alcanzó una tasa de éxito en torno al 95 %. La mayoría de los fallos estuvieron relacionados con errores en la aplicación y no en la infraestructura desplegada.

Las figuras 5.17 y 5.18 muestran el número de trabajos de análisis y el volumen de datos leídos en el PIC y el CIEMAT, respectivamente. Se compartieron los recursos de computación con las tareas de producción

[2]cortafuegos. Es un elemento de hardware o software utilizado en una red de computadoras para controlar las comunicaciones, permitiéndolas o prohibiéndolas, según las políticas de red que haya definido la organización responsable.

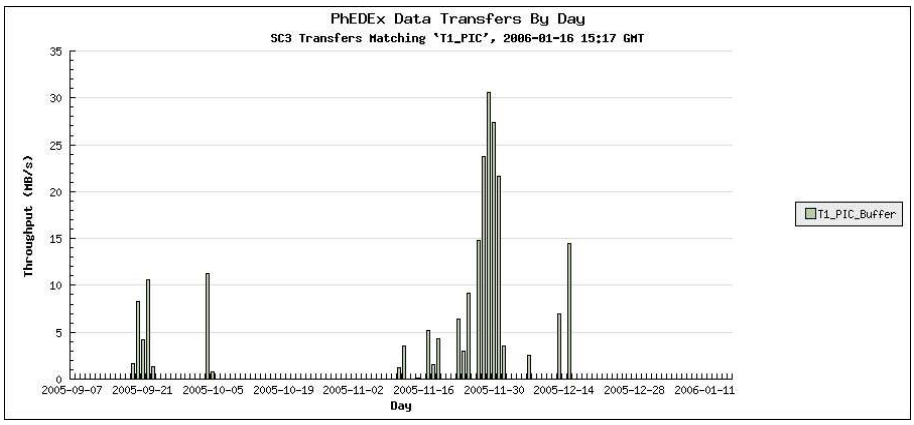


Figura 5.15: Tasa de transferencias diaria entre el CERN y el PIC durante la fase de servicio en el LCG SC3.

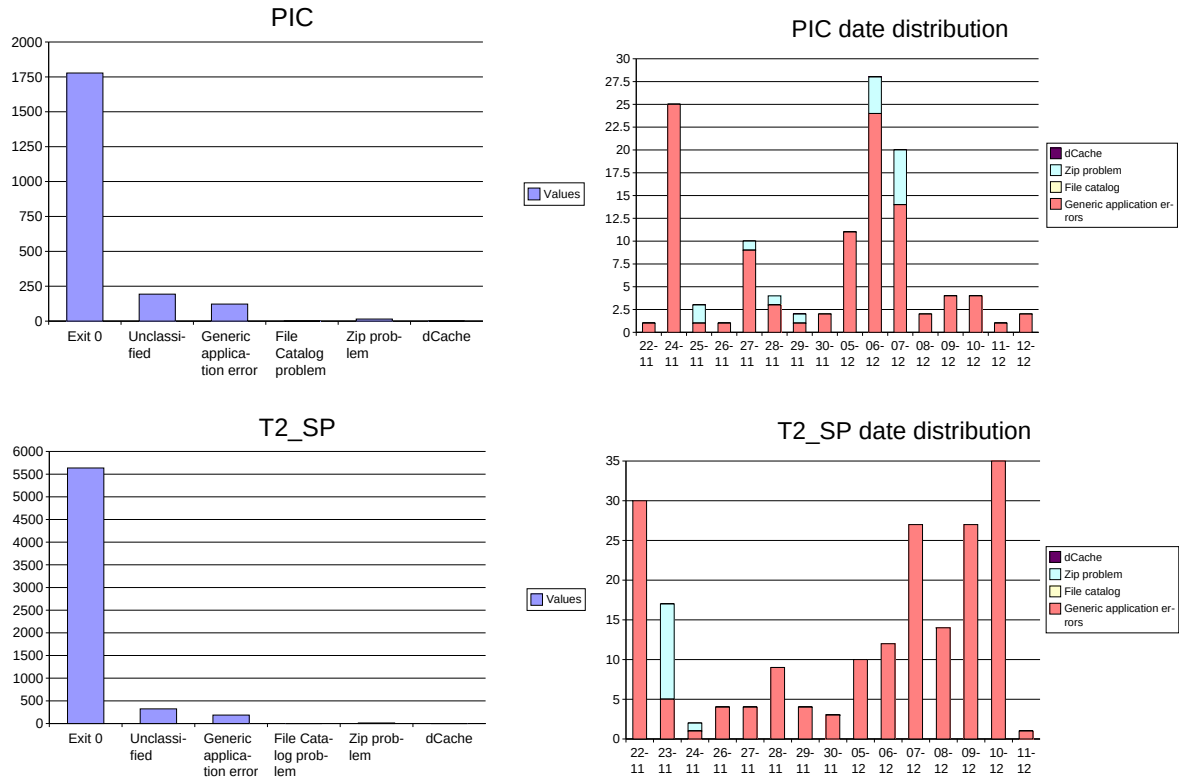


Figura 5.16: Distribución de los códigos de error devueltos por los programas de análisis (izquierda) y su evolución temporal (derecha) en el PIC (arriba) y el CIEMAT (abajo) durante la fase de servicio en el SC3.

Monte Carlo, por lo que no fue posible mantener de forma constante un gran número de trabajos de análisis en ejecución. Por otra parte, la cantidad de datos que cada trabajo podía leer estaba limitada a un valor de, aproximadamente, 1 MB/s, debido al tiempo requerido para procesar los datos. Estos dos fueron los principales factores limitantes, más que el rendimiento del sistema de almacenamiento. Se alcanzaron picos de 130 trabajos en el PIC y 110 en el CIEMAT, y valores máximos de acceso a los

datos de 200 MB/s y 80 MB/s en PIC y CIEMAT, respectivamente. En ambos casos se cumplieron las espectativas del SC3 para el rendimiento de los centros Tier-1 y Tier-2.

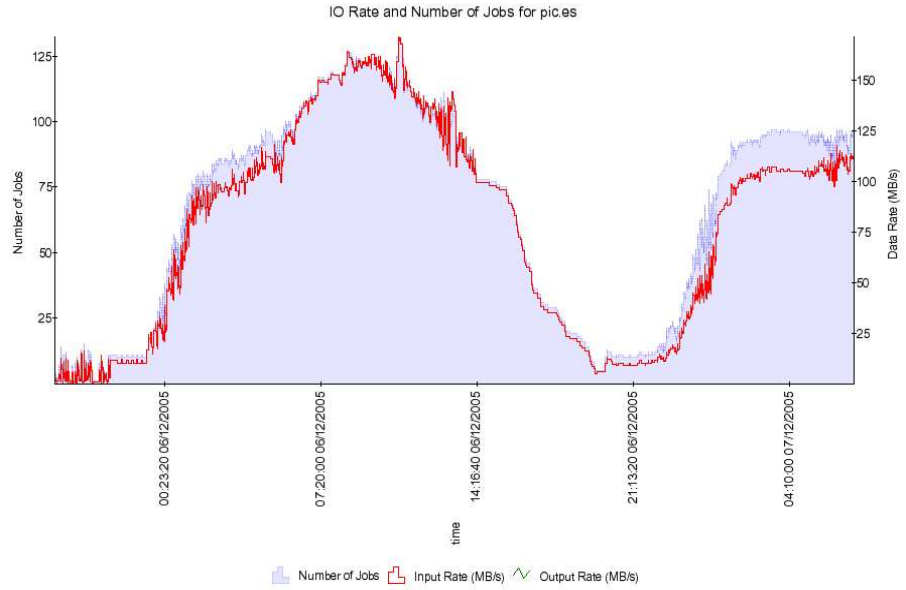


Figura 5.17: Número de trabajos y volumen de datos leídos en el PIC durante la fase de servicio en el SC3.

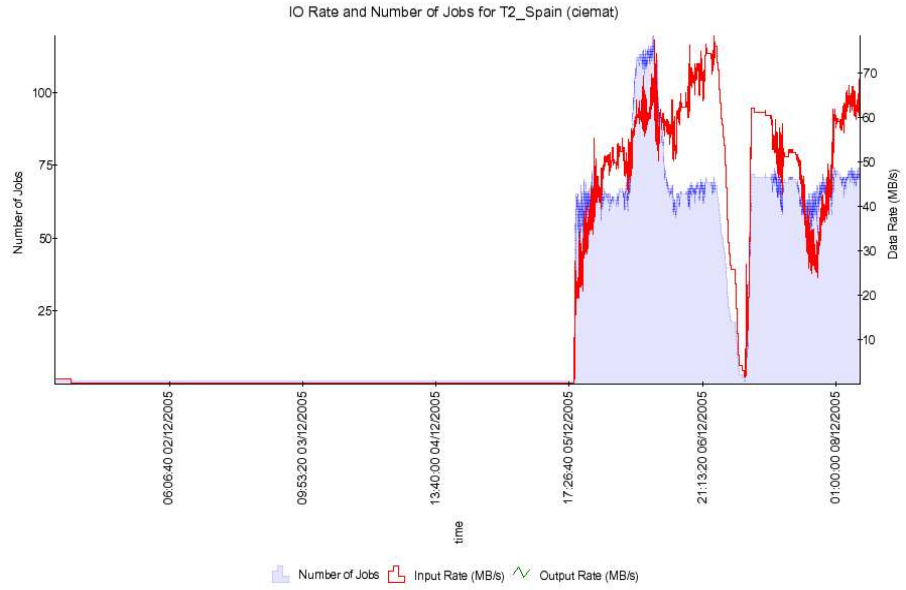


Figura 5.18: Número de trabajos y volumen de datos leídos en el CIEMAT durante la fase de servicio en el SC3.

5.2.4. Experiencia

Durante las operaciones se vio que algunos componentes del sistema completo funcionaron bastante bien, mientras que otros ofrecieron un rendimiento algo peor. En general, las herramientas ofrecidas por LCG para la gestión de los datos (el catálogo LFC, el sistema de almacenamiento DPM, el servicio de transferencias FTS, la interfaz con los sistemas de almacenamiento SRM, el sistema de monitorización R-GMA, etc.) no estaban lo bastante maduras y su funcionamiento no había sido adecuadamente chequeado antes de empezar el ejercicio. Sin embargo, las operaciones realizadas con las herramientas propias de CMS dieron muy buen resultado, tanto en la gestión de los datos (PhEDEx, RefDB y PubDB) como en la gestión de los trabajos (CRAB y JobRobot). El incremento en el tamaño medio de los ficheros (alrededor de 2 GB) y la disminución del número de ficheros (mediante la compresión en archivos ZIP) dieron resultados muy positivos.

Los principales problemas que se encontraron estaban relacionados con los servicios SRM en CASTOR. El servicio SRM declaraba las transferencias finalizadas antes de que CASTOR concluyese todas las operaciones internas. Esta situación provocaba una cierta desconfiguración en PhEDEx al no poder validar la transferencia. Otro problema que se detectó estaba relacionado con el modo en que se ejecutaban las transferencias. Era el servidor SRM del centro de destino el que iniciaba las operaciones para copiar los ficheros a su nodo. Esto provocó que el gran número de consultas para averiguar el estado de las transferencias saturase este servidor.

Se observó una baja tasa de transferencia de datos por cada stream TCP, entre 1 y 2 MB/s, dependiendo del número de streams simultáneas. Esto se traduce en un flujo de datos individual algo bajo. Se llegó a un valor de saturación de 5 MB/s cuando se ejecutaban 5 streams en paralelo, y este valor no se superó aunque se aumentase el número de streams. Tampoco se vio ninguna influencia del ajuste de los parámetros de configuración TCP en el servidor SRM sobre el flujo de datos. Por todo esto, para conseguir una tasa de transferencia de datos razonable, se tuvieron que ejecutar un número elevado de transferencias en paralelo (entre 10 y 15, con 5 streams cada una). Sin embargo, este modo de trabajo provocó una gran fragmentación de los ficheros en los discos de destino, lo que induce un bajo rendimiento en las operaciones de CASTOR. Para aliviar esta situación se decidió configurar los discos de CASTOR con un gran número de particiones y limitar las transferencias a un único stream cada vez. Otra solución que ayudó a minimizar los efectos de la fragmentación de los discos fue separar los pools de escritura/lectura de datos de los de análisis.

Al igual que ocurrió durante las primeras operaciones de producción MC en Grid, se comprobó que el complicado proceso de publicación de los datos introducía grandes retrasos en el sistema e impedía la automatización de las actividades. Para minimizar los efectos de este complejo proceso se instaló un gran pool de discos que pudiese contener grandes colecciones de datos completas. Así se pudo ejecutar la publicación de los datos a medida que estos se iban transfiriendo. Además, fue necesario ejecutar los trabajos de publicación en paralelo para acelerar este proceso.

El hecho de que aún no se hubiesen implementado en PhEDEx las políticas de prioridades de CMS fue otra causa de ineficiencia. Al no poder priorizarse las operaciones, todas las Data Collection se transferían a la vez, con lo que ninguna de estas transferencias se llevaba a cabo a un buen ritmo. Este retraso se propagaba finalmente hasta las tareas de análisis, pues no pueden ejecutarse sobre colecciones parciales.

En general se pudo comprobar que el WMS de LCG había mejorado su rendimiento de forma significativa con respecto a ejercicios anteriores. Como ocurría en las operaciones de producción MC, la mayoría de los problemas en los trabajos de análisis estuvieron relacionados con el acceso local a los datos o fallos en el software de análisis, junto con la falta de un código de errores apropiado.

5.3. Computing, Software and Analysis Challenge 2006

El CSA06 empezó el 2 de octubre de 2006 y duró, aproximadamente, seis semanas. Se diseñó para que fuese un test del 25 % de la capacidad requerida para las primeras operaciones en 2008. Los objetivos del CSA06 fueron los siguientes:

- Preparación de grandes Datasets de sucesos simulados (incluyendo algunos la clasificación de sucesos del HLT).
- Reconstrucción en el Tier-0 a 40 Hz usando el nuevo entorno de procesamiento de sucesos de CMS (CMSSW).
  - Reconstrucción a 40 Hz usando la última versión del software oficial del experimento.
  - Aplicación de las constantes de calibración guardadas en las bases de datos offline.
  - Generación de datos en formato FEVT y AOD, y de Datasets de datos filtrados, apropiados para las tareas de alineamiento y calibración, conocidos como *AlCaReco skims*.
  - Separación de las muestras etiquetadas por el HLT en unos 10 streams.
- Distribución de los Raw Data y de los RECO Data a los centros Tier-1.
- Envío de trabajos de filtrado de los datos a los centros Tier-1 y propagación de los resultados a los centros Tier-2.
- Análisis de los datos en los centros Tier-2 sobre los datos filtrados.
- Demostración de los flujos de trabajos de re-reconstrucción en los centros Tier-1.
- Demostración de los flujos de trabajos de calibración, producción de Datasets de calibración/alineamiento en el Tier-0, transferencia a los Tier-1 y ejecución de los trabajos de calibración en los Tier-1.

El CSA06 comenzó con la reconstrucción en el Tier-0, seguido de la distribución de los datos a los centros Tier-1, añadiendo finalmente los flujos de trabajos de procesamiento de datos. Sin embargo, no se ejercitaron todos los flujos de datos y de trabajos posibles del modelo de computación de CMS. No se ejercitaron las transferencias de datos entre centros Tier-1 y desde los Tier-2 a los Tier-1. No se llevó a cabo producción Monte Carlo en los Tier-2 de forma simultánea a las actividades de análisis. No se incluyó el procesamiento (y gestión del almacenamiento) de HLT en el Tier-0. Estas tareas se ejecutarán, y se harán de forma simultánea, durante el CSA07.

5.3.1. Configuración

No se desplegaron recursos dedicados en los centros españoles para el CSA06. Como en los ejercicios anteriores, se utilizaron los recursos ya existentes utilizados habitualmente para las tareas de producción MC. Todos los recursos de CMS se pusieron a disposición del CSA06 puesto que, con la excepción de las tareas de análisis de los usuarios en los centros Tier-2, las actividades habituales se interrumpieron durante el ejercicio.

La figura 5.19 muestra el ancho de banda y las latencias aproximadas de las diferentes secciones de la red entre el CERN y los centros españoles. Estos valores se encuentran resumidos en la tabla 5.5. Hubo varias diferencias en la configuración de la red con respecto a ejercicios anteriores. El tráfico desde el CERN llega a España a través de la red europea Geant-2, en este caso desde Ginebra, con un ancho de banda de 10 Gb/s, y llega hasta un nodo en Madrid. Desde este nodo, la red académica RedIRIS transporta los datos dentro de España con un ancho de banda de 2.5 Gb/s. También se incorporaron dos tramos para el acceso hasta IFCA a través de RedIRIS, uno desde Madrid y otro desde Barcelona. El ancho de banda hasta el PIC a través de Anella Cientifica estaba limitado a 1 Gb/s. La latencia total desde el CERN al PIC era del orden de 20 ms. Las latencias entre el PIC y el CIEMAT y entre el PIC y el IFCA eran de 10 ms y 16 ms, respectivamente. En el año 2007 se ha desplegado en España un enlace de 10 Gb/s como

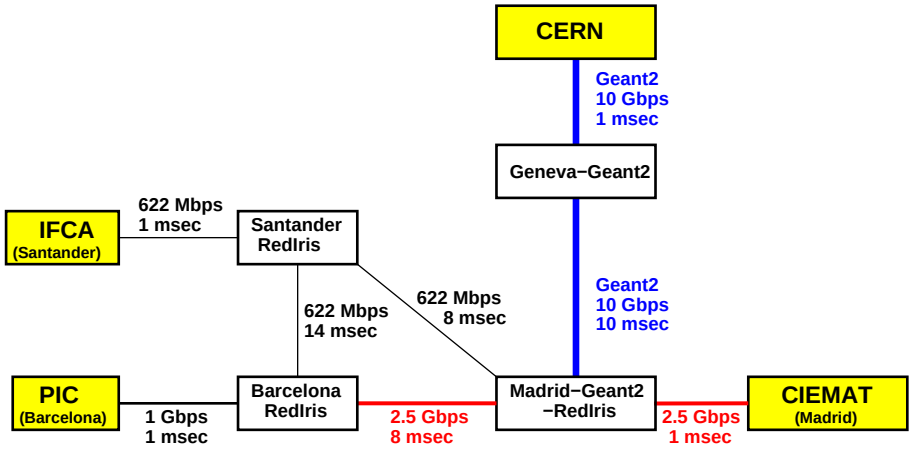


Figura 5.19: Diagrama de red con el ancho de banda y latencias aproximadas de las diferentes secciones de la red entre el CERN y los centros españoles.

Red	Ámbito	Ancho de banda	Latencias máximas
GEANT	Europeo	10 Gb/s	11 ms
RedIRIS	España	2.5 Gb/s	14 ms
Anella Cientifica	Cataluña	1 Gb/s	1 ms

Tabla 5.5: Infraestructura de red desde el CERN hasta los centros españoles para el DC04.

parte de la infraestructura de Geant-2. Gracias a este enlace, el PIC podrá formar parte de la red óptica privada del LHC para el CSA07.

La figura 5.20 detalla los flujos de trabajos y de datos y los recursos de computación en los centros españoles durante el CSA06.

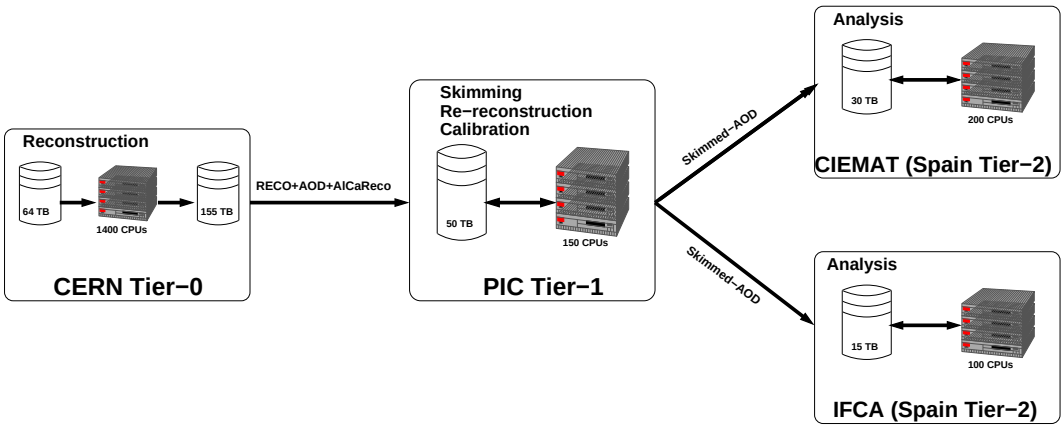


Figura 5.20: Flujos de datos y de trabajos en los centros españoles durante el CSA06.

En el PIC se recibía un flujo continuo de datos (con datos reconstruidos) desde el CERN a un ritmo medio de 22 MB/s. Las tareas de filtrado y re-reconstrucción se llevaron a cabo en el PIC, mientras que las actividades de análisis sobre los datos filtrados se ejecutaron en el CIEMAT y el IFCA. Finalmente,

se mandaron los datos para los estudios de alinenamiento (en un formato adecuado para la calibración y el alineamiento) al IFCA desde el CERN a través del PIC.

Para poder participar en el CSA06, un centro Tier-1 debía aportar un mínimo de 70 TB de espacio para almacenamiento y 150 CPUs. El PIC, al tener un tamaño entre 1/2 y 1/3 de un centro Tier-1 nominal, contribuyó con 50 TB de espacio (gestionado con dCache) y con 150 CPUs. En el caso del PIC no se ejercitaron movimientos de datos entre cintas y discos, y todos los datos estaban disponibles en discos. Este espacio de disco estaba repartido entre 15 servidores, cada uno de los cuales proporcionaba entre 3 y 4 TB. La alta distribución del espacio de almacenamiento permitió distribuir la carga de las operaciones de escritura/lectura simultáneas (transferencias de datos, lecturas y escrituras de los trabajos de procesamiento) entre los servidores.

En el caso de los centros Tier-2, los requisitos mínimos para participar en el CSA06 eran de 5 TB de espacio disponible y 20 CPUs. El CIEMAT aportó 30 TB (gestionados con CASTOR) y 200 CPUs, y el IFCA contribuyó con 15 TB de espacio (gestionado con DPM) y unas 100 CPUs.

5.3.2. Producción Monte Carlo previa al CSA06

La simulación Monte Carlo de las muestras necesarias para realizar el CSA06 se llevó a cabo unos pocos meses antes de empezar, durante el verano del 2006. Esta fue la primera producción a gran escala con ProdAgent. Se crearon cuatro equipos de trabajo encargados de esta producción masiva, uno de los cuales fue el grupo del CIEMAT. Los resultados y experiencia durante esta fase de pre-producción se describieron en el capítulo anterior (sección 4.2.2).

5.3.3. Operaciones en el Tier-1 y el Tier-2

5.3.3.1. Transferencias de datos

Los centros Tier-1 debían recibir datos desde el CERN a un ritmo equivalente al 25 % del que se producirá durante las primeras tomas de datos en 2008, y distribuir estos datos a los centros Tier-2. La tabla 5.6 compila los valores previstos y alcanzados para todos los Tier-1. En el caso del PIC, se esperaba un ritmo de recepción de datos de unos 10 MB/s durante todo el CSA06, y se alcanzó un valor promedio de unos 22 MB/s con una eficiencia superior al 97 %.

Site	Nominal (CSA) Rate	Last 30 Day average	Last 15 Day average	Outage (Days)	MSS used
ASGC	15 MB/s	17 MB/s	23 MB/s	0	(YES)
CNAF	25 MB/s	26 MB/s	37 MB/s	0	(YES)
FNAL	50 MB/s	68 MB/s	98 MB/s	0	YES
FZK	25 MB/s	23 MB/s	28 MB/s	3	NO
IN2P3	25 MB/s	23 MB/s	34 MB/s	1	YES
PIC	10 MB/s	22 MB/s	33 MB/s	0	NO
RAL	10 MB/s	23 MB/s	33 MB/s	2	YES

Tabla 5.6: Valores nominales y alcanzados para las tasas de transferencia de datos en los centros Tier-1 durante el CSA06.

La figura 5.21 (superior izquierda) muestra la tasa diaria de transferencia de datos desde el CERN al PIC y desde el PIC hasta el CIEMAT y el IFCA. Esta tasa estuvo limitada por la falta de datos disponibles en el CERN durante la primera semana del ejercicio, como se puede ver en la figura superior derecha donde se representa la cantidad de datos encolados en espera de ser transferidos. Las transferencias se llevaron a cabo sin retrasos la mayoría de las veces, excepto en algunos periodos concretos donde se inyectaron intencionadamente una gran cantidad de datos para comprobar las transferencias *bursty* [3]. La figura inferior muestra la cantidad acumulada de datos transferidos a cada uno de los centros españoles. Se copiaron del CERN al PIC unos 60 TB de datos, mientras que del PIC al CIEMAT y al IFCA se transfirieron unos 30 TB y 15 TB, respectivamente. La calidad de estas operaciones de transferencia se puede ver en la figura 5.22. Hubo algunos problemas con el sistema de almacenamiento del IFCA durante la primera semana que impidieron las tranferencias de datos al centro. La calidad de las tranferencias al CIEMAT empeoró ligeramente durante la última semana, cuando se ejercitaron transferencias no regionales Tier-1 a Tier-2. En general, las operaciones de tranferencia de datos desde el PIC a IFCA y CIEMAT fueron excelentes durante todo el ejercicio.

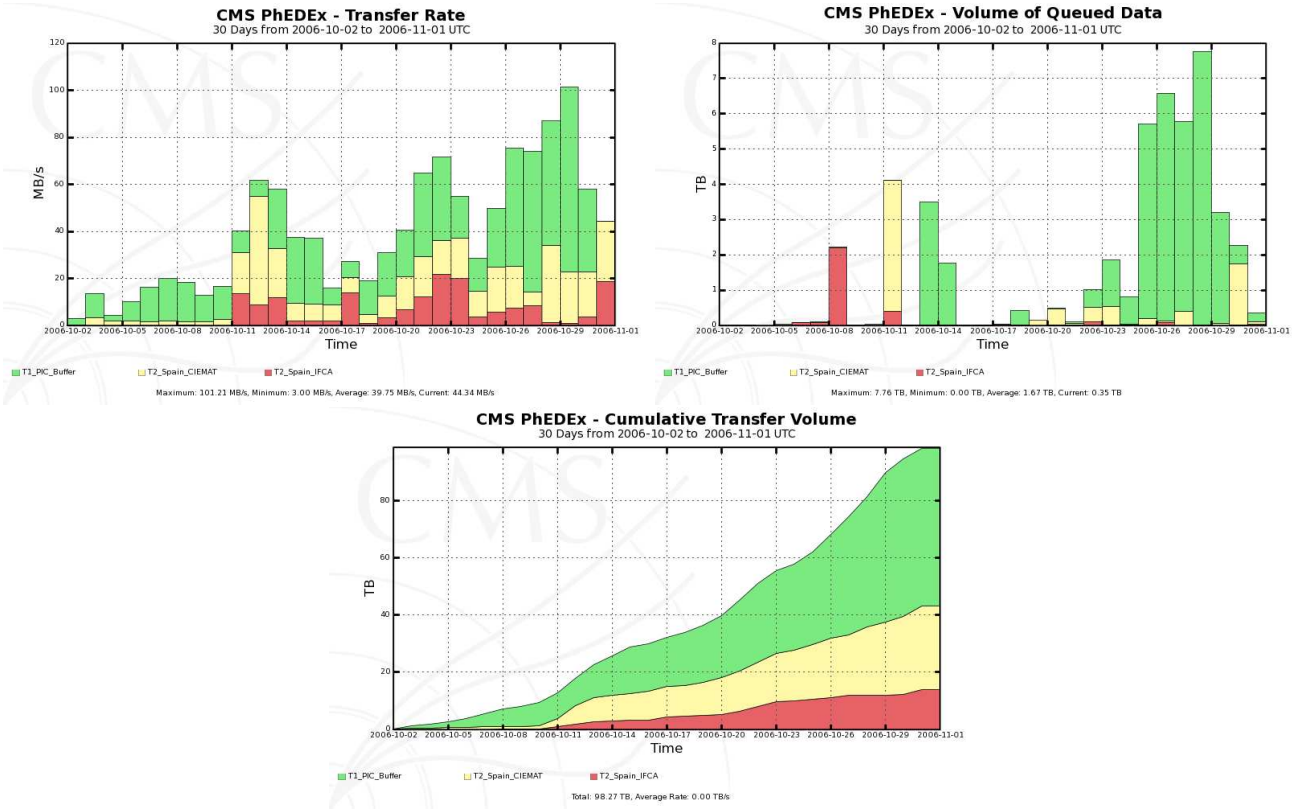


Figura 5.21: Tasas de transferencias de datos desde el CERN hacia el centro Tier-1 español y desde éste hasta el Tier-2 (superior izquierda), datos en espera de ser transferidos (superior derecha), y volumen acumulado de datos transferidos (inferior).

Se ejercitaron las trasferencias de datos bursty desde el CERN a los centros Tier-1 durante el CSA06 para chequear la capacidad de recuperación del sistema en caso de grandes retrasos provocados por posibles problemas en las transferencias. La figura 5.23 superior izquierda muestra la tasa promedio por hora alcanzada durante una de estas transferencias bursty entre el CERN y el PIC. Se consiguió un ritmo medio sostenido de unos 80 MB/s durante 10 horas, sin errores de transferencia, saturando el ancho de

[3] el término inglés bursty hace referencia a los datos que son transferidos en un corto plazo de tiempo, generalmente de forma irregular y repentina.

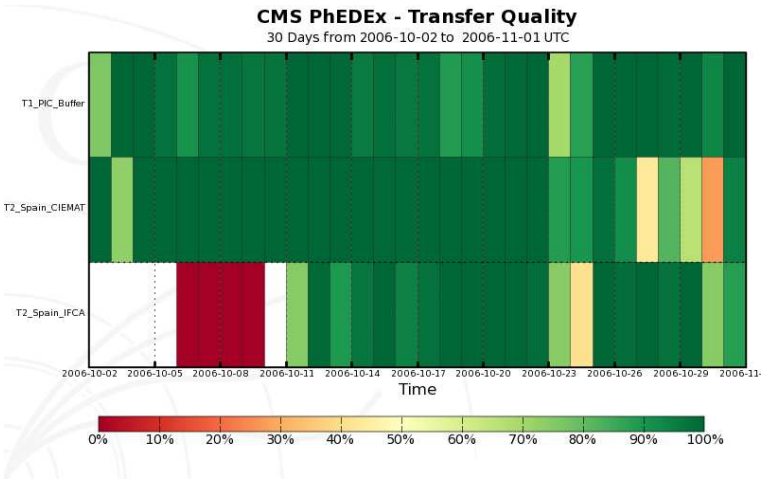


Figura 5.22: Calidad de las transferencias desde el CERN al PIC.

banda disponible entre el CERN y el PIC. La figura superior derecha muestra cómo la cantidad de datos pendientes de ser transferidos, inicialmente 3 TB, decrecía a ritmo constante. La figura 5.24 muestra la perfecta calidad de las transferencias durante este ejercicio.

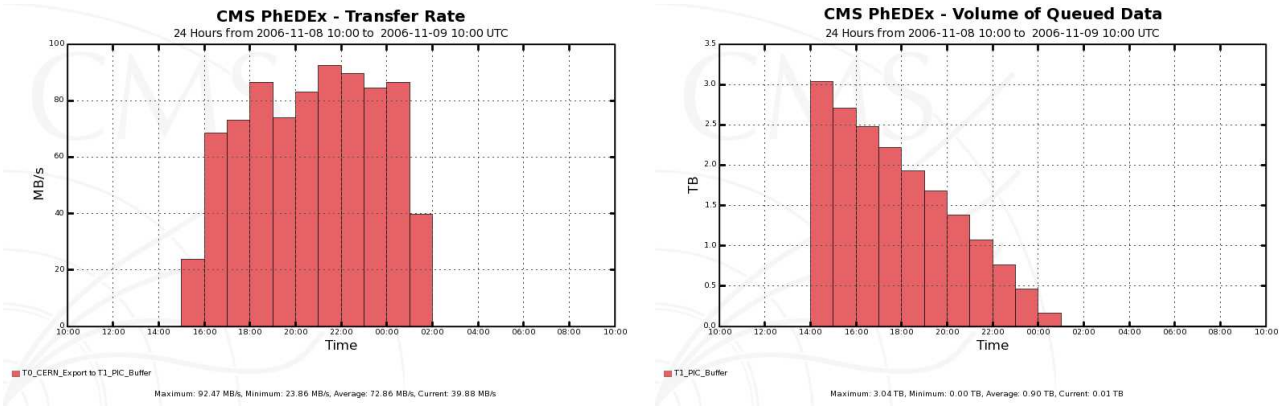


Figura 5.23: Transferencias bursty desde el CERN al PIC para simular la recuperación tras un periodo sin transferencias. Tasa de transferencia (izquierda), volumen de datos retrasados (derecha).

Según el modelo de computación de CMS cada centro Tier-1 debe guardar una copia de las muestras completas, en formato AOD, de los datos reconstruidos. Estos ficheros AOD guardan parte de la información de reconstrucción (suficiente para el análisis de datos cuando el detector está perfectamente entendido). Sin embargo, para llevar a cabo el análisis de los datos durante los primeros años del experimento se necesitará acceso a la información completa de reconstrucción. Como los datos reconstruidos están repartidos entre los distintos centros Tier-1 es necesario que se puedan ejecutar transferencias eficientes de las muestras filtradas desde todos los Tier-1 a todos los Tier-2. Para comprobarlo se ejercitó durante el CSA06 la transferencia simultánea de un Dataset desde el PIC a varios centros Tier-2. La figura 5.25 muestra la calidad de las transferencias para cada uno de los 25 centros Tier-2 que participaron en esta prueba. Se transfirió el Dataset completo a la mayoría de los centros con éxito, y sólo hubo problemas en tres centros, no achacables al PIC.

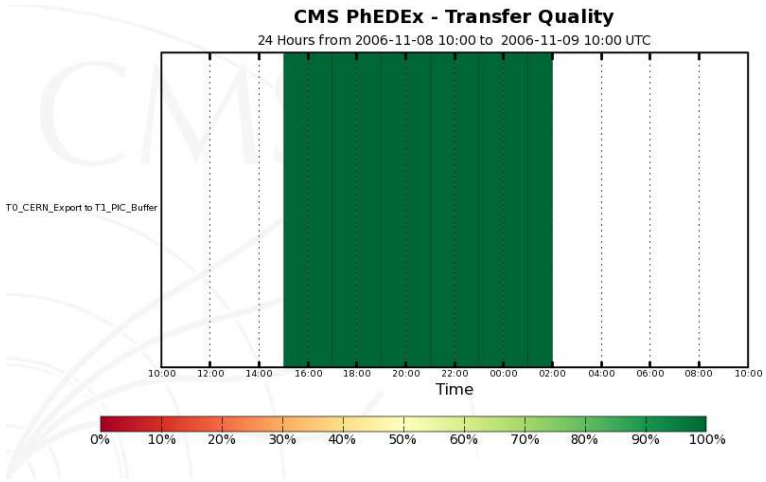


Figura 5.24: Calidad de las transferencias bursty desde el CERN al PIC.

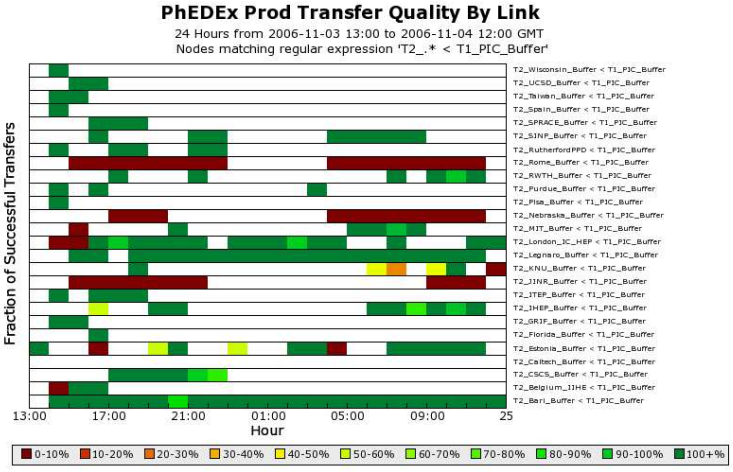


Figura 5.25: Calidad de las transferencias simultáneas de un mismo Dataset desde el PIC a 25 Tier-2 diferentes.

En el modelo de computación de CMS las transferencias desde los centros Tier-1 a los centros Tier-2 serán bursty por naturaleza. Se deberán transferir los datos filtrados desde los Tier-1 a aquellos centros Tier-2 interesados, lo más rápidamente posible, a medida que están disponibles. La figura 5.26 muestra las transferencias bursty de 5 TB de datos durante 24 horas desde el PIC al CIEMAT. Se puede ver que se realizaron a una velocidad media de 60 MB/s. En la figura 5.27 se ve cómo la eficiencia de estas transferencias fue prácticamente del 100 %. Esta cantidad de datos en espera de ser transferidos, inducida artificialmente, se ve reflejada también en la figura 5.21 para la fecha 2006-10-11.

También se probaron con éxito las transferencias no regionales, desde un Tier-1 diferente al PIC, al CIEMAT y al IFCA. La figura 5.28 muestra los datos transferidos desde el Tier-1 de FNAL, tanto al CIEMAT como al IFCA, a una velocidad media de 20 MB/s. La figura 5.29 refleja la buena calidad de estas transferencias.

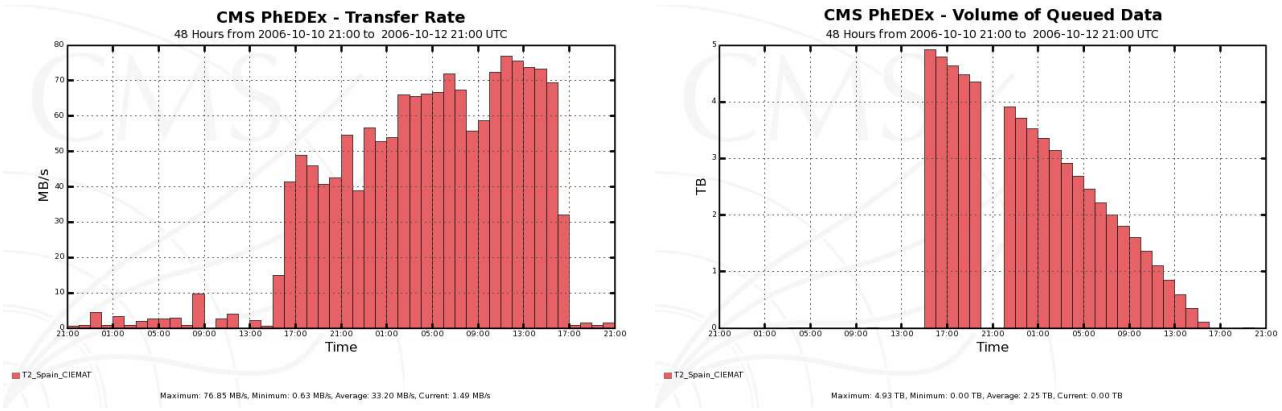


Figura 5.26: Transferencias bursty de 5 TB de datos desde el PIC al CIEMAT. Tasa de transferencia (izquierda), y volumen de datos retrasados (derecha).

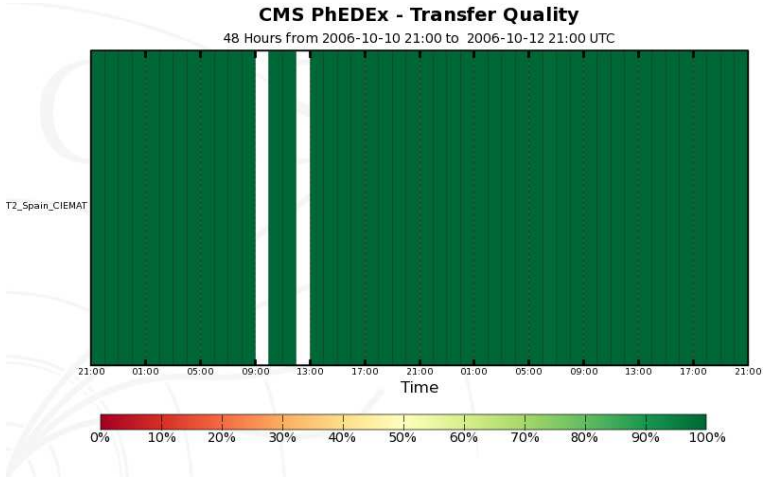


Figura 5.27: Calidad de las transferencias bursty del PIC al CIEMAT.

5.3.3.2. Filtrado de sucesos

Las operaciones de filtrado de sucesos se realizaron en los centros Tier-1. Los trabajos de filtrado aplicaron distintos filtros a los datos, produciendo los ficheros de output correspondientes a cada filtro con el número de sucesos seleccionados. Se usó ProdAgent para gestionar los trabajos de filtrado, que se encargaba de preparar de forma automática los trabajos para las muestras seleccionadas, de enviarlos al Grid, y de lanzar los trabajos de merge correspondientes cuando había un número de sucesos filtrados adecuado.

En el PIC se ejecutaron varios pases de filtrado sobre distintas muestras. Un ejemplo característico de estas operaciones de filtrado es el que se ejecutó sobre una muestra de dos millones de sucesos del tipo $Z^0 \rightarrow \mu\mu$ (generada durante la pre-producción del verano de 2006). El filtro seleccionó, aproximadamente, el 50 % de los sucesos de la muestra.

La figura 5.30 muestra el número de trabajos de filtrado enviados, en espera, en ejecución y finalizados en función del tiempo, y la figura 5.31 el número acumulado de sucesos filtrados y que han sufrido el proceso de merge. Se puede ver que se necesitaron unos dos días para filtrar la muestra completa, y un día más para la operación de merge sobre los datos filtrados.

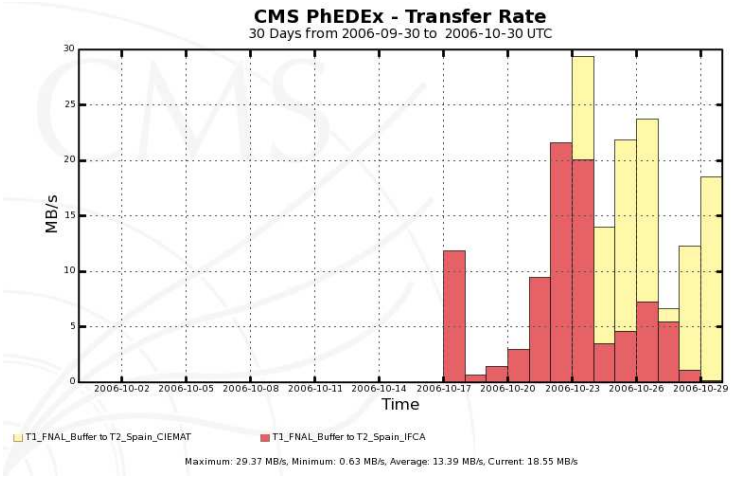


Figura 5.28: Tasa de transferencias no regionales desde el Tier-1 de FNAL al CIEMAT y al IFCA.

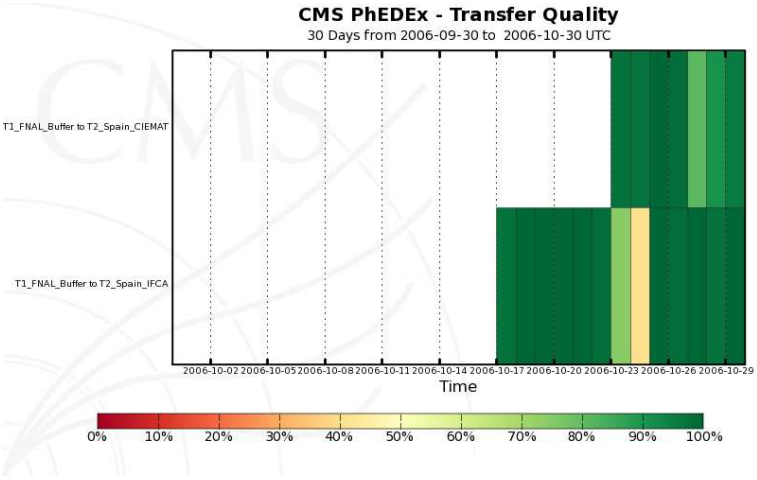


Figura 5.29: Calidad de las transferencias desde FNAL a los centros Tier-2 españoles.

La figura 5.32 muestra el tiempo de procesamiento por suceso para los trabajos de filtrado. Sólo eran necesarios unos pocos segundos para procesar cada suceso. Es notable la presencia de dos picos en la distribución, que corresponden a dos conjuntos de CPUs con diferente rendimiento.

La figura 5.33 muestra distintas medidas de eficiencia para los trabajos de filtrado y de merge, según las expresiones siguientes:

$$Grid\_eff = 1 - \frac{N_{Aborted}}{N}$$

$$Application\_eff = \frac{N_{Success} + N_{StageOutFailure}}{N_{Finished}}$$

$$Stageout\_eff = 1 - \frac{N_{StageOutFailure}}{N_{Success} + N_{StageOutFailure}}$$

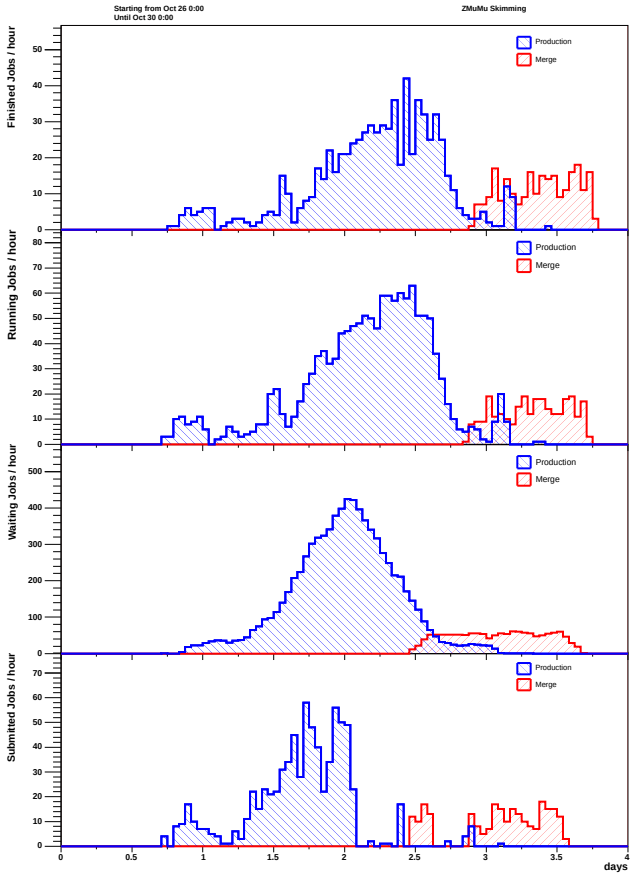


Figura 5.30: Número de trabajos de procesamiento, y los correspondientes de merge, enviados, en espera, en ejecución, y finalizados en función del tiempo correspondientes al filtrado $Z^0 \rightarrow \mu\mu$.

$$Job\_eff = \frac{N_{Success}}{N_{Finished}}$$

La *Grid efficiency* da cuenta del número de trabajos enviados que no han sido abortados debido a problemas con el middleware del Grid. La *Application efficiency* muestra la fracción de trabajos para los cuales el software de filtrado acabó satisfactoriamente. La *Stage out efficiency* da la fracción de trabajos que no tuvieron problemas a la hora de guardar el output en el sistema de almacenamiento. Finalmente, la *Job efficiency* muestra el porcentaje de trabajos que, una vez empezaron su ejecución en un WN, no fallaron por alguna razón (incluyendo ineficiencias al acceder al software del experimento o en las operaciones de escritura del output). Hubo una fase de un 15 % de ineficiencia, debido a problemas en los servicios Grid, para los trabajos de filtrado. Estos trabajos fallidos fueron reenviados automáticamente por ProdAgent. Las otras tres medidas de eficiencias mostraron valores cercanos al 100 %.

La figura 5.34 muestra la velocidad de lectura/escritura (arriba) y el consumo de memoria (abajo) a lo largo de un día en uno de los WN donde se ejecutaron varios trabajos de filtrado. Se puede ver que los trabajos de filtrado procesan los datos de input a un ritmo promedio de 1 MB/s, y guardan el output en el sistema de almacenamiento local a gran velocidad tras finalizar el proceso de filtrado. El consumo de memoria aumentaba ligeramente con el número de trabajos procesados hasta unos 600 MB.

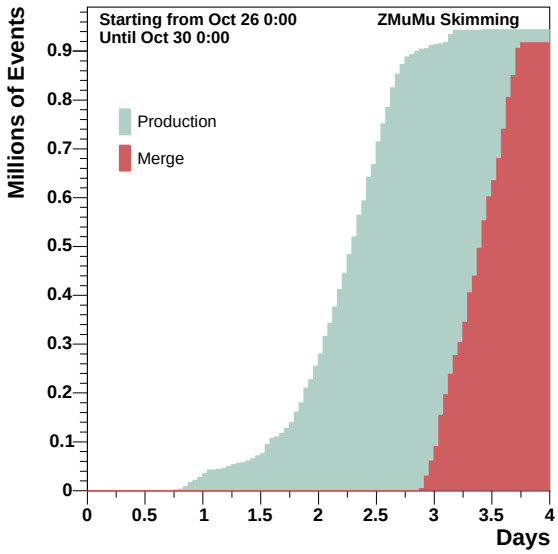


Figura 5.31: Número acumulado de sucesos en función del tiempo para los trabajos de procesamiento, y de merge, para el filtrado $Z^0 \rightarrow \mu\mu$.

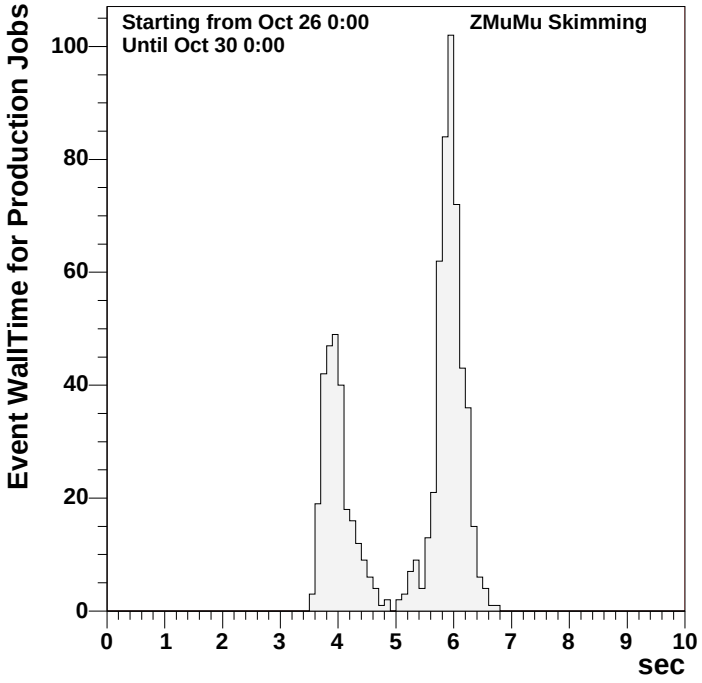


Figura 5.32: Distribución del tiempo de procesamiento por suceso para los trabajos de filtrado de $Z^0 \rightarrow \mu\mu$.

5.3.3.3. Re-reconstrucción de los sucesos

Se ejercitó el flujo de trabajos de re-reconstrucción en los centros Tier-1 durante la última etapa del CSA06. El objetivo fue reprocesar al menos 100000 sucesos en cada centro Tier-1. En el PIC se procesaron unos 900000 sucesos, como puede verse en la figura 5.35.

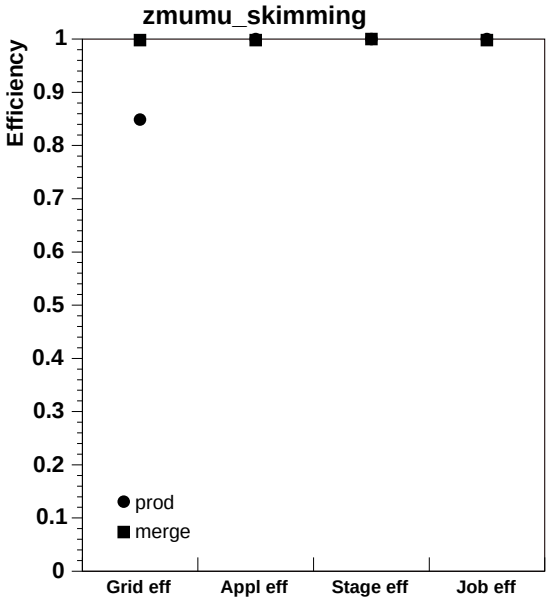


Figura 5.33: Valores de las distintas eficiencias para los trabajos de procesamiento, y de merge, correspondientes al filtrado $Z^0 \rightarrow \mu\mu$.

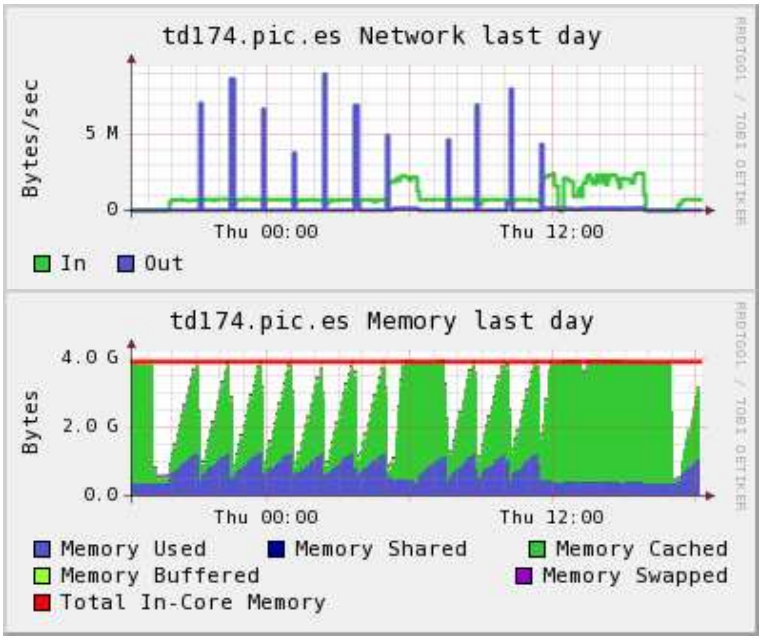


Figura 5.34: Velocidad de lectura/escritura de datos (arriba) y consumo de memoria (abajo) para los trabajos de merge correspondientes al filtrado de $Z^0 \rightarrow \mu\mu$.

La figura 5.36 muestra el número de trabajos enviados, encolados, en ejecución y finalizados en función del tiempo. Se necesitaron tan sólo dos días para reprocesar la mayoría de los datos, y un tercero para hacer las operaciones de merge de los resultados.

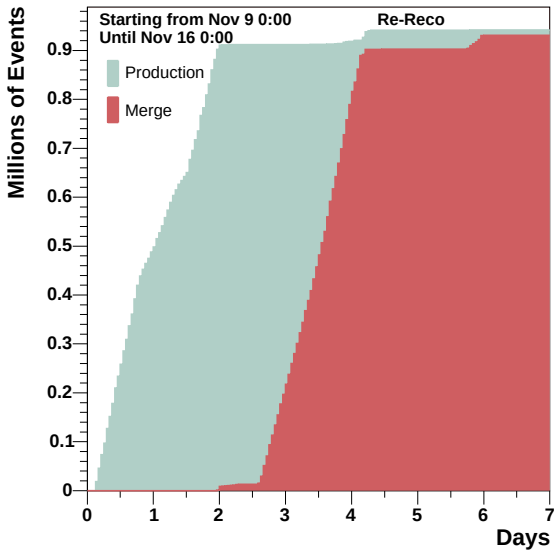


Figura 5.35: Número acumulado de sucesos en función del tiempo para los trabajos de procesamiento, y de merge, correspondientes a las tareas de re-reconstrucción.

En la figura 5.37 se puede ver la distribución del tiempo necesario para llevar a cabo el reprocesamiento de un suceso. En media, para la re-reconstrucción de un suceso del tipo $Z^0 \rightarrow \mu\mu$ se necesitaban unos 10 segundos.

Los sucesos fueron reprocesados en el PIC con una gran eficiencia, como se puede ver en la figura 5.38 Tan sólo se dio un pequeña ineficiencia del 1 % en el envío de los trabajos al Grid.

La figura 5.39 muestra la distribución de masa invariante del bosón Z que se obtiene a partir de los datos reconstruidos en el Tier-0 (caso 'ideal'), a partir de sucesos procesados a los que se ha aplicado un cierto desalineamiento como input para el algoritmo de alineamiento (conocidos como 'misaligned'), y de sucesos re-reconstruidos en el PIC usando las constantes de alineamiento derivadas del algoritmo de alineamiento ('realineados'). En la figura se aprecia claramente la degradación en la resolución de la masa y la recuperación de los valores originales tras el reprocesamiento con las constantes de alineamiento apropiadas.

Se hizo uso del caché local de FroNTier para acceder a las constantes de calibración y alineamiento para el reprocesamiento de los datos. La figura 5.40 muestra el flujo de datos (en kB/s) desde la caché de FroNTier hacia los trabajos de reprocesamiento. La línea azul corresponde a las transferencias desde la base de datos central en el CERN hasta la caché local, y la línea verde corresponde a la transferencia de la información a los trabajos de re-reconstrucción de datos directamente desde la caché local de FroNTier.

5.3.3.4. Actividades de análisis

Los grupos de Física prepararon una gran variedad de ejercicios de análisis físicos para el CSA06, con la intención de validar el flujo de trabajos de análisis. Las muestras filtradas en el PIC se transfirieron al CIEMAT para su análisis. Se filtraron las muestras $Z^0 \rightarrow \mu\mu$ y de SoftMuon usando un algoritmo diseñado por los físicos del CIEMAT. En ambos casos, aproximadamente la mitad de los sucesos pasaron el filtro. Los números de sucesos iniciales y filtrados para ambas muestras se recogen en la tabla 5.7.

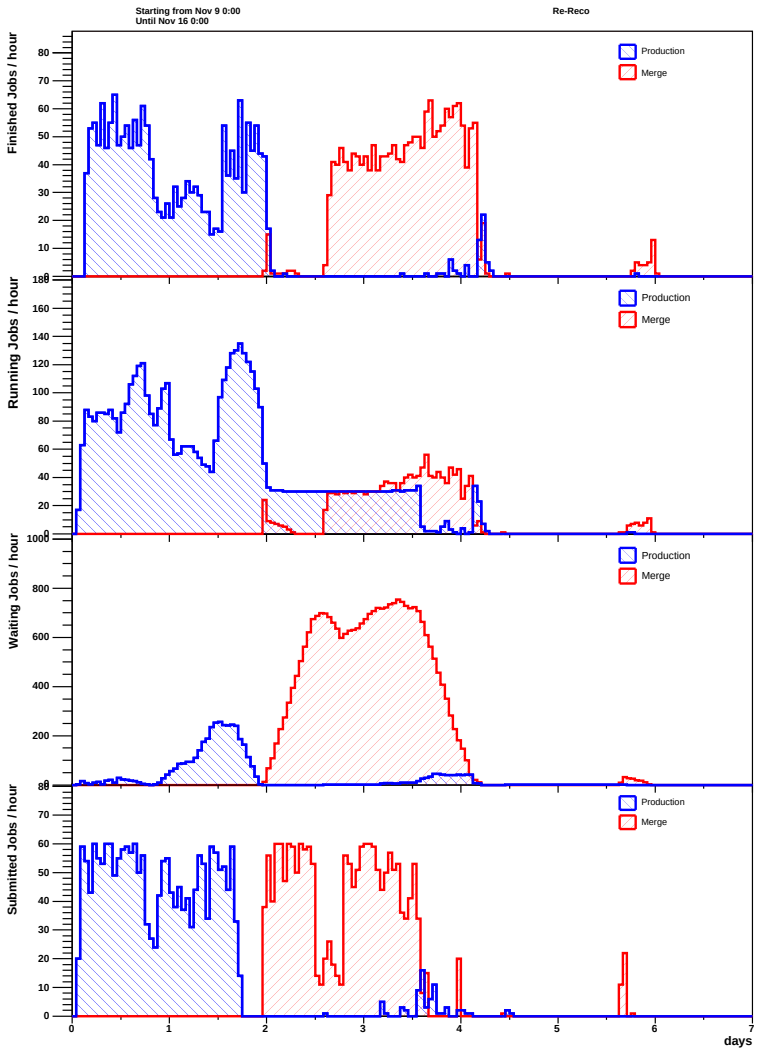


Figura 5.36: Número de trabajos de procesamiento, y los correspondientes de merge, enviados, en espera, en ejecución, y finalizados en función del tiempo correspondientes a las tareas de re-reconstrucción.

Muestra	Sucesos reconstruidos	Sucesos filtrados
ZMuMu	2.04 x 10^6	948 x 10^3
SoftMuon	1.70 x 10^6	886 x 10^3

Tabla 5.7: Sucesos reconstruidos y filtrados, para dos muestras, durante la fase de análisis del CSA06.

Los trabajos de análisis se enviaron al Tier-2 del CIEMAT haciendo uso de la herramienta oficial de CMS para la gestión de trabajos (CRAB). Las tres muestras filtradas (llamadas ZMuMu, SoftMuon y EWKSoup) se procesaron usando un programa de análisis simple, que guardaba los histogramas de output en ficheros de ROOT [125] (uno por trabajo), y copiaba estos ficheros a CASTOR en el CIEMAT a medida que se iban generando. Un script, ejecutado de forma asíncrona, se encargó de juntar los ficheros de ROOT en un único fichero para cada Dataset.

La figura 5.41 muestra la velocidad de lectura promedio (unos 2 MB/s) de los trabajos de análisis al

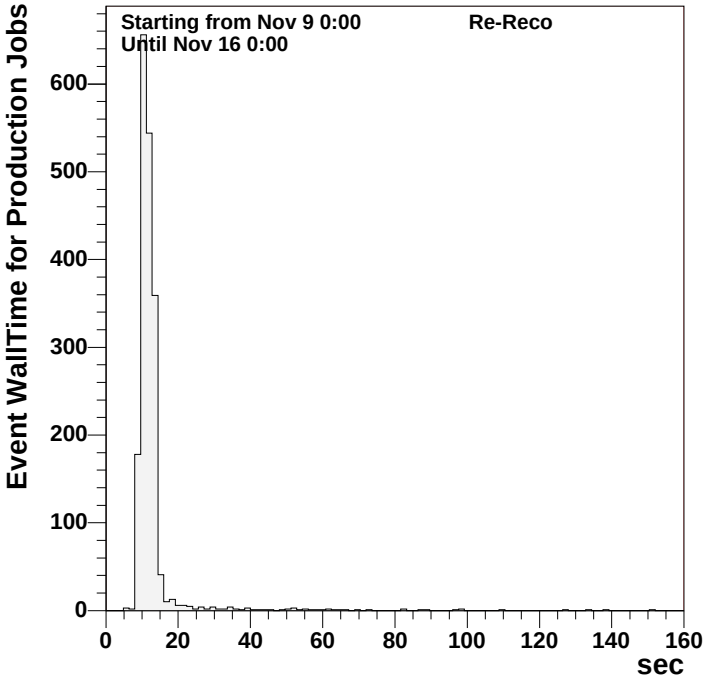


Figura 5.37: Distribución del tiempo de procesamiento por suceso para los trabajos de re-reconstrucción.

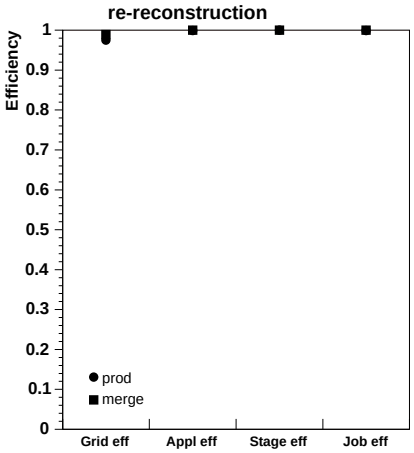


Figura 5.38: Distintas eficiencias para los trabajos de procesamiento, y de merge, correspondientes a las tareas de re-reconstrucción.

acceder a las muestras filtradas. La figura corresponde a una frecuencia de sucesos analizados de unos pocos Hz.

En las figuras 5.42 se muestra un ejemplo de la distribución de masa invariante de dimuones para sucesos procedentes de las dos muestras. Los muones globales corresponden a trazas medidas en el tracker y en las cámaras de deriva, mientras que los StandAlone sólo se miden en las cámaras. La distribución StandAlone corresponde a las medidas de P_T de los muones en la posición de entrada de las cámaras de

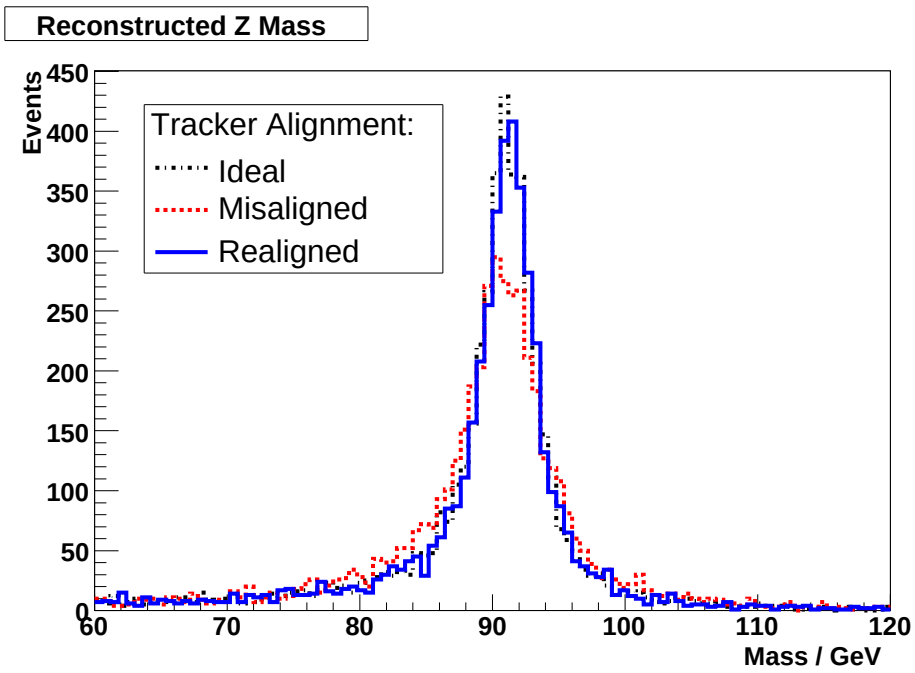


Figura 5.39: Masa invariante reconstruida del Z^0 sin tener en cuenta efectos de desalineamiento ('ideal'), teniendo en cuenta los efectos del desalineamiento ('desalineada') y después de aplicar las constantes de alineamiento durante la re-reconstrucción ('realineada').

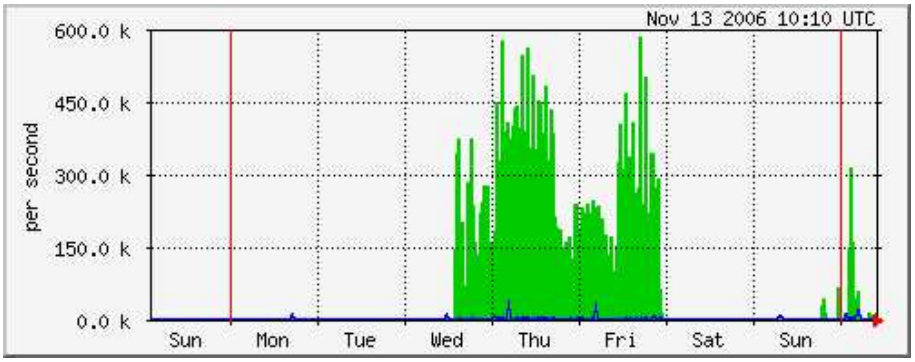


Figura 5.40: Velocidad de lectura de datos en el caché local de FroNTier utilizado por los trabajos de reprocesamiento.

deriva, que da una distribución de masa de dimuones degradada.

En las figuras 5.43 el momento de los muones StandAlone está extrapolado teniendo en cuenta la pérdida de energía en el detector, e imponiendo que los muones pasen por el vértice, lo que mejora significativamente la resolución en la masa de los dimuones.

La figura 5.44 (arriba) muestra la distribución entre los centros de los trabajos de análisis enviados por los usuarios durante el CSA06. Se ejecutó un número sustancial de trabajos de análisis en los centros españoles. Aparte de los trabajos de los usuarios, se mandaron también trabajos de análisis ficticios de forma centralizada mediante JobRobot con el objetivo de estresar el sistema de gestión de trabajos y alcanzar una escala de envío de más de 10000 trabajos diarios. Se preparaban y enviaban estos trabajos

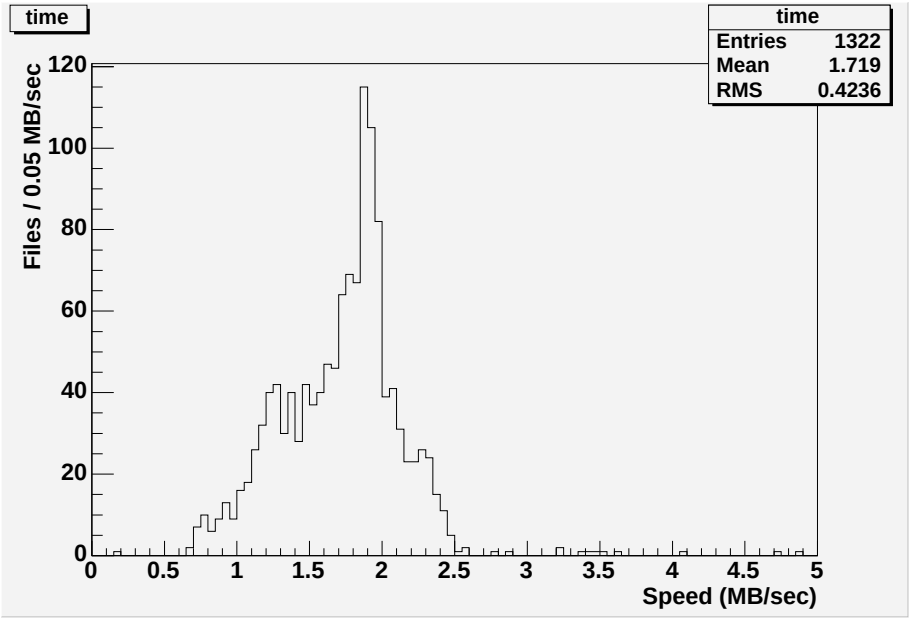


Figura 5.41: Velocidad de lectura de datos por parte de los trabajos de análisis.

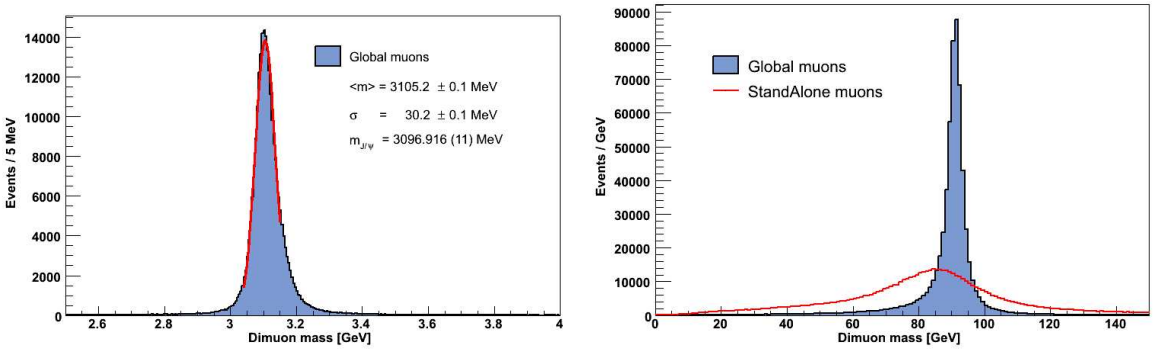


Figura 5.42: Distribuciones de masa invariante de dimuones procedentes de J/ψ (izquierda) y Z^0 (derecha) para muones globales y StandAlone.

en función de los datos publicados en los centros, y complementaban la pequeña escala alcanzada con los trabajos de los usuarios. También eran útiles para estresar los sistemas de almacenamiento de los centros mediante la lectura masiva de los datos. En total, JobRobot envió más de medio millón de trabajos. En el CIEMAT se recibieron y ejecutaron un gran número de estos trabajos a una alta frecuencia, como se puede ver en la figura 5.44 (abajo).

5.3.4. Experiencia

El CSA06 demostró ser un test completo de los sistemas de gestión de datos y de trabajos de CMS extremadamente útil y satisfactorio. Se ejercitaron varios de los flujos de procesamiento de datos de CMS y se llevaron a cabo tests de estrés de los servicios en los centros que participaron. Se cumplieron todos los objetivos previstos.

Sin embargo, no se ejercitaron todas las posibilidades del modelo de computación de CMS. Por ejemplo,

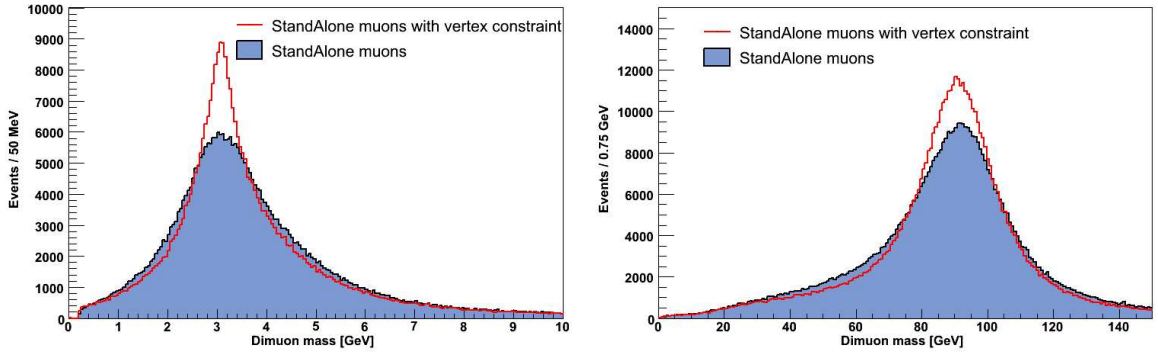


Figura 5.43: Distribuciones de masa invariante de dimuones procedentes de J/ψ (izquierda) y Z^0 (derecha) para muones StandAlone con y sin restricciones de vértice.

no se realizaron transferencias de datos entre centros Tier-1, de los Tier-2 a los Tier-1, ni transferencias no regionales de los Tier-1 a los centros Tier-2. Tampoco se llevaron a cabo tareas de producción Monte Carlo en los Tier-2 en paralelo a las actividades de análisis. No se hicieron procesados de High Level Trigger ni se probaron las funcionalidades del Storage Manager en el Tier-0. Sin embargo, es indudable que la experiencia ganada durante el CSA06 será de gran utilidad para obtener los mejores resultados posibles en el CSA07.

El CSA06 fue un gran éxito, pero con el coste de una gran carga operacional. Hubo un esfuerzo dedicado durante todo el test donde los expertos en computación, gestores, los administradores locales de los centros y las personas de contacto supervisaron la operación del sistema. Sin embargo, CMS se encuentra ya en fase de transición entre la etapa de desarrollo e integración del modelo de computación a una fase estable de operaciones. Se espera, por tanto, que la carga de operar las actividades de computación se reduzca significativamente y un equipo central de operaciones pueda gestionar los flujos de trabajos y de datos de CMS.

5.3.4.1. Gestión de datos

Las transferencias de datos desde el CERN a los centros Tier-1 funcionaron bastante bien durante el CSA06 (mejor que en ejercicios previos), en parte gracias al soporte continuado del equipo de CASTOR del CERN. Las transferencias regionales desde los centros Tier-1 a los centros Tier-2 asociados también funcionaron bien. Se vio que, por el contrario, las transferencias no regionales necesitan mejorarse significativamente.

Se ha visto que validar y optimizar un cierto enlace requiere una gran cantidad de tiempo. Hay muchos elementos involucrados en esta operación (como el ancho de banda disponible, latencias, cortafuegos, paralelismo en las transferencias, diferentes *timeouts*, configuración de los servidores FTS, configuración de GridFTP y SRM, configuración de PhEDEx, etc.) Los sistemas de almacenamiento, las implementaciones de SRM y el middleware Grid adolecen a veces aún de cierta inestabilidad y falta de fiabilidad. Además, las diferentes capas que componen los servicios de transferencias (GridFTP/SRM/FTS) son difíciles de configurar de forma óptima y de depurar en caso de problemas.

Se ha comprobado que la arquitectura FTS actual para la exportación de datos desde los centros Tier-1 puede causar serios problemas en los centros exportadores. Con la arquitectura actual, cuando un centro Tier-2 recibe datos desde un Tier-1, es el servidor FTS del Tier-1 de referencia asociado a este Tier-2 el que gestiona las operaciones. Por tanto, no es el servidor del Tier-1 de origen el que realiza las gestiones, lo que le impide controlar el número total de transferencias que se ejecutan con centros Tier-2 no asociados como destino. Cuando estas transferencias no controladas localmente son demasiadas el centro Tier-1

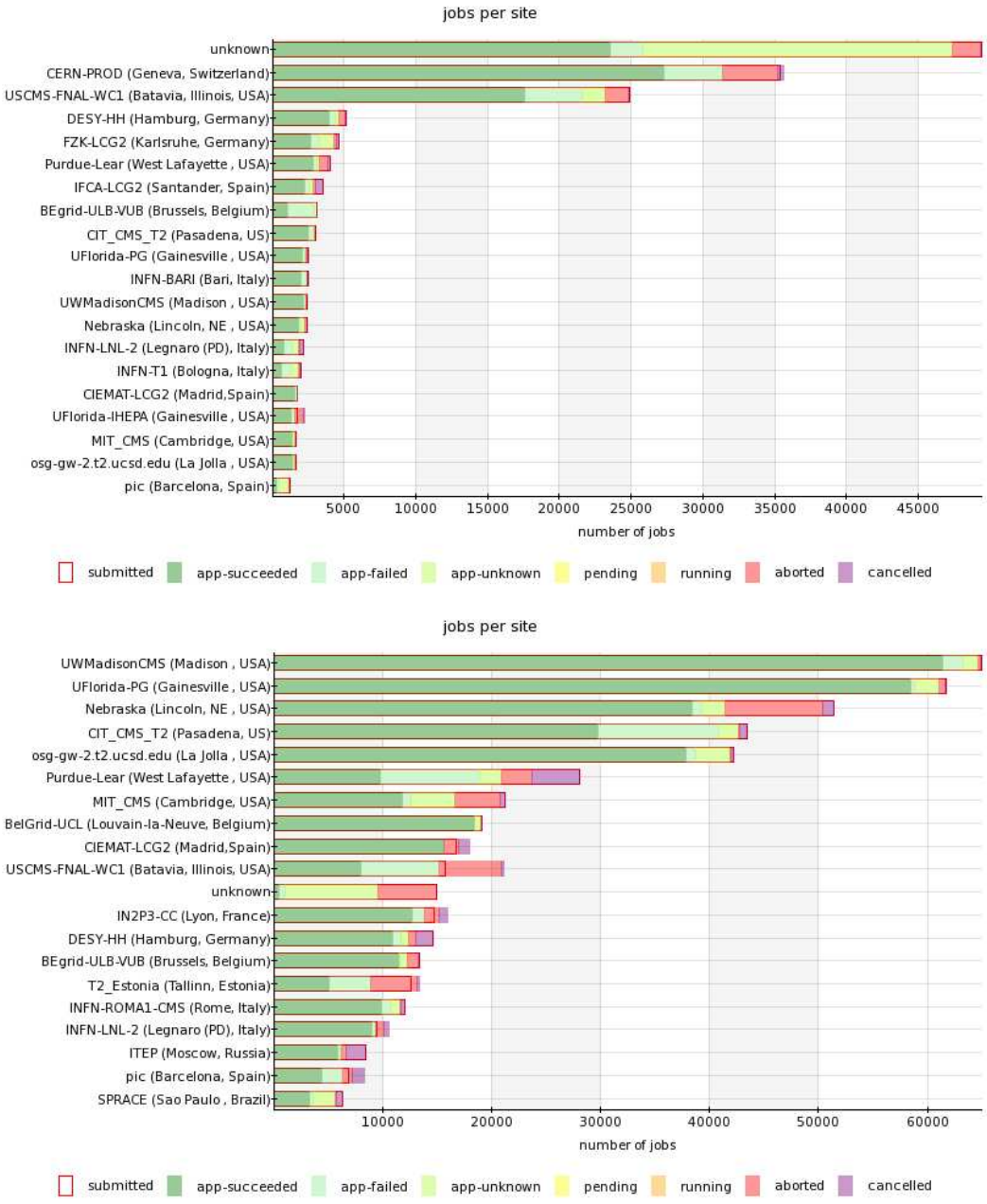


Figura 5.44: Distribución entre los centros de los trabajos de análisis de los usuarios (arriba) y de JobRobot (abajo) durante el CSA06.

puede sobrecargarse y ofrecer un servicio degradado o incluso quedar temporalmente fuera de servicio. En vista de esta circunstancia CMS ha propuesto cambiar la arquitectura de FTS para que sea el Tier-1 exportador el que gestione siempre las operaciones.

La capacidad de PhEDEx para ejecutar transferencias *multi-hop* (con nodos intermedios) implica la necesidad de recursos de almacenamiento en disco adicionales para almacenar temporalmente los datos en tránsito en los nodos intermedios. Esta posibilidad, por tanto, no se ejercitó durante el CSA06 debido a la dificultad para gestionar los datos en los nodos intermedios (como evitar que se copien en cinta en

los Tier-1 o la necesidad de borrar las copias una vez que han llegado a su destino final).

En PhEDEx no se especifica el nodo origen de una transferencia, sólo el de destino. De esta forma, si existen varias réplicas de los mismos datos, el sistema de enrutamiento de PhEDEx puede encontrar dinámicamente la mejor ruta teniendo en cuenta la topología y los registros de las tasas de transferencia de los enlaces. Aunque este enrutamiento dinámico será especialmente útil para las transferencias entre centros Tier-1 (y especialmente para la distribución de los datos AOD), no se utilizó durante el CSA06. Durante el CSA06 los flujos de datos siguieron principalmente la dirección Tier-0 → centros Tier-1 → centros Tier-2 regionales.

La gestión de las subscripciones a los datos, borrado y registro en DLS fueron operaciones ejecutadas manualmente durante el CSA06. Esto fue motivo de una gran carga de trabajo para el equipo central de PhEDEx y para los administradores locales de los centros. Estas operaciones ya han sido automatizadas en PhEDEx. Ahora, los administradores de los centros pueden solicitar, modificar o eliminar subcripciones a los datos y solicitar el borrado de las réplicas de los bloques y Datasets a través de un agent local de borrado de PhEDEx.

Durante el CSA06 no se soportaron prioridades en las transferencias. Esta posibilidad se ha implementado recientemente en PhEDEx, y ahora los administradores locales pueden cambiar las prioridades de las transferencias para cada subcripción.

La sincronización de las diferentes fuentes de información sobre los datos (el sistema de transferencias, DBS y DLS, y los sistemas de almacenamiento de cada centro) también fue una operación manual, lo que provocó algunas inconsistencias. En caso de pérdida de datos, eran los administradores locales los responsables de informar de este suceso al sistema de gestión de datos. Actualmente se están desarrollando herramientas para sincronizar automáticamente las bases de datos centrales con los datos que realmente existen en los sistemas de almacenamiento.

Durante el CSA06 se ha probado la utilidad de disponer de agentes locales de PhEDEx que interaccionan con los sistemas de almacenamiento en cada centro, especialmente en los Tier-1. Las tareas de configuración, optimización y depurado de problemas son más fáciles para los administradores locales. Además, estos agentes locales suplen la carencia de algunas funcionalidades de la interfaz de SRM con estos sistemas de almacenamiento.

5.3.4.2. Gestión de trabajos

La gestión de todos los trabajos (producción MC durante la etapa de pre-producción, filtrado, análisis, merge, procesamiento en el Tier-0, etc.) se hizo mediante ProdAgent, cuyo rendimiento fue muy bueno durante todo el CSA06. La experiencia ganada durante la fase de pre-producción MC demostró que ProdAgent era un sistema lo bastante robusto, fácil de configurar y escalable como para convertirse en una herramienta idónea para la gestión de cualquier flujo de trabajos.

La ejecución de las tareas de filtrado y reprocesado fue bastante satisfactoria durante el CSA06. Se ha probado que se puede aplicar una selección múltiple simultáneamente. Las componentes que realizan las operaciones de merge y de registro funcionaron bien. Se alcanzó el objetivo de 1 MB/s/trabajo para la velocidad de lectura de datos de los programas de análisis y filtrado. Sin embargo, no se ejercitó la recuperación de ficheros migrados a cinta antes del procesamiento. Durante el CSA06 la mayoría de los datos estaban disponibles en disco. Para automatizar la recuperación previa de los ficheros en cinta en los centros Tier-1, haciendo así más eficiente y fiable el procesamiento de los datos, sería necesario un acoplamiento entre ProdAgent y el sistema de transferencia de datos.

Los flujos de trabajos de análisis sobre los datos filtrados mediante CRAB también se ejecutaron de forma satisfactoria. El nivel de trabajos de análisis enviados por los propios físicos es aún algo bajo, por

lo que se procedió al envío de un gran número de trabajos de análisis ficticios para estresar los sistemas de entrega de datos.

Se pudo comprobar durante el CSA06 que tanto los servicios globales Grid como algunos servicios locales en los centros (como los sistemas de colas o de almacenamiento) siguen siendo inestables y poco fiables. El procesamiento de datos a través del Grid sigue consumiendo gran cantidad de recursos humanos. Se demostró así la necesidad de aumentar el nivel de automatización en ProdAgent (lo que incentivó el uso de las nuevas componentes que permiten la inyección continua de trabajos y la gestión automática de los workflows), el uso de bulk operations para aumentar la escala de trabajos que cada instancia de ProdAgent puede manejar, el cambio de trabajos basados en el número de sucesos a trabajos basados en tiempo, y el análisis del uso de los recursos y del rendimiento a través de un herramienta de monitorización y accounting de ámbito global.

Durante el CSA06 no hubo ningún paso de validación de la configuración y versiones del software. Esta fase de validación sería muy útil para identificar problemas antes de comenzar la ejecución de los trabajos de procesamiento. Se ha implementado este procedimiento, con posterioridad al CSA06, mediante la producción sistemática de un gran número de sucesos cada vez que se instala una versión nueva del software del experimento. Se ha visto, finalmente, que se debe mejorar el nivel de transparencia de cara al usuario final.

Capítulo 6

Operaciones del sistema de computación de CMS

Tras todos los esfuerzos que se han realizado durante los últimos años, y que aún continúan, el modelo de computación de CMS está en fase de ser completamente implementado de forma satisfactoria. Tanto la gestión de los trabajos como la de los datos han mejorado de forma considerable, y el nivel cada vez mayor de automatización en todas las tareas involucradas está permitiendo un uso intensivo y eficiente de los recursos Grid disponibles.

En todas las actividades (producción Monte Carlo, análisis de datos, transferencias de datos, etc.) ha habido un aumento progresivo del volumen de operaciones durante el último año, con épocas de mayor actividad durante los ejercicios de Computing, Software and Analysis Challenges (CSA).

Los centros españoles, además, han tomado un papel destacado, tanto en el diseño e implementación de las herramientas como, y gracias a ello, en la operación de los sistemas de transferencia y procesamiento de los datos.

6.1. Gestión de datos

La figura 6.1 muestra la tasa de datos transferidos entre los distintos centros de computación mediante PhEDEx durante el último año de operaciones de CMS. El valor promedio semanal máximo que se ha alcanzado en este periodo es de unos 1270 MB/s. Durante las operaciones del CSA06, en los meses de octubre y noviembre de 2006, el valor medio de las transferencias de datos fue del orden de los 300 MB/s. Después de esta etapa, las transferencias aumentaron de nuevo a partir de marzo del 2007 gracias a una transferencia continua de datos entre centros diseñada para ejercitar el sistema de transferencia y almacenamiento de datos (conocido con el nombre de LoadTest), manteniéndose desde entonces en un nivel más o menos constante de unos 800 MB/s. Esto equivale a un volumen superior a 0.5 PB transferidos cada semana.

El sistema de transferencia de datos de CMS ha demostrado, gracias a las operaciones del LoadTest, que puede alcanzar una escala importante en el volumen de transferencia de datos. Estos resultados dan confianza en que CMS está en buen camino para poder gestionar el volumen de transferencias de datos que dicta el modelo de computación de CMS. Cuando se alcanza el mayor ritmo de operaciones, el modelo de computación especifica unas transferencias de, en promedio, unos 300 MB/s desde el CERN a cada uno de los centros Tier-1 y unos 60 MB/s de los Tier-1 a los centros Tier-2 asociados. Teniendo en cuenta que existen unos 25 centros Tier-2, el total alcanza el valor de unos 1500 MB/s. También habrá unos 10 MB/s/Tier-2 (lo que equivale a 1 TB/día) de datos Monte Carlo hacia los Tier-1. En total, suponen unos 250 MB/s. En términos globales, los movimientos de datos en CMS serán de unos 2 GB/s.

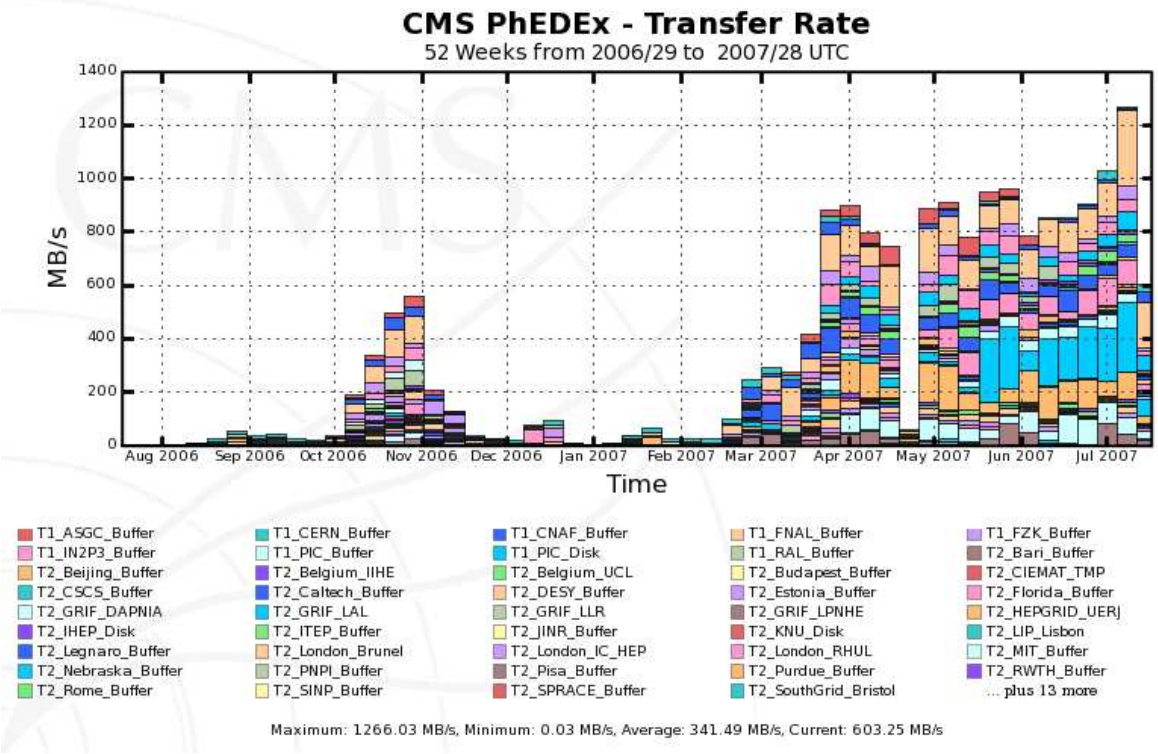


Figura 6.1: Tasa de transferencia de datos durante el último año de operaciones. Los distintos colores muestran la tasa de transferencia total a cada centro desde la suma de todos los demás.

La figura 6.2 muestra el volumen de datos transferido a los distintos centros. El valor máximo que se ha alcanzado es superior a los 730 TB de datos transferidos en una semana. Durante las operaciones del CSA06 el volumen de datos transferidos fue del orden de 25-35 TB por día. El volumen acumulado durante estas operaciones se puede ver en la figura 6.3. Durante el último año se han movido unos 10 PB de datos, donde algo más de 1 PB corresponde al CSA06, y el resto ha tenido lugar a partir de marzo del 2007.

Finalmente, la figura 6.4 muestra la calidad (porcentaje de éxitos en las transferencias individuales de ficheros) de estas operaciones durante el último año. Se aprecia que, en general, la calidad de las operaciones ha sido media/baja, aunque alternando con periodos en los que el rendimiento ha sido muy bueno. Sin embargo, como se desarrolla en la siguiente sección, la calidad de las transferencias en los principales enlaces de distribución de datos (del Tier-0 a los Tier-1 y entre los Tier-1 a sus Tier-2 asociados) es buena. Son las transferencias entre centros Tier-1 y entre Tier-1 y Tier-2 no asociados las que aún deben mejorar bastante. En cualquier caso, hay que tener en cuenta que el sistema de transferencia de datos reintenta automáticamente las trasferencias fallidas, de modo que los datos siempre llegan a su destino, aunque con una latencia mayor cuando la calidad es peor.

6.1.1. Transferencias desde el CERN a los centros Tier-1

De entre todas las operaciones de transferencia de ficheros hay algunas que en el modelo de computación de CMS tienen mayor importancia. Un ejemplo significativo es la distribución de los datos en formato RAW, RECO y AOD desde el CERN a los centros Tier-1. Estas operaciones son especialmente críticas, pues se deben llevar a cabo en tiempo real, distribuyendo las muestras a medida que son proporcionadas por el detector y reconstruidas casi en tiempo real en el CERN. Las figuras 6.5 y 6.6 muestran la velocidad y calidad de estas operaciones, respectivamente, durante el último año de operaciones. Se ha alcanzado

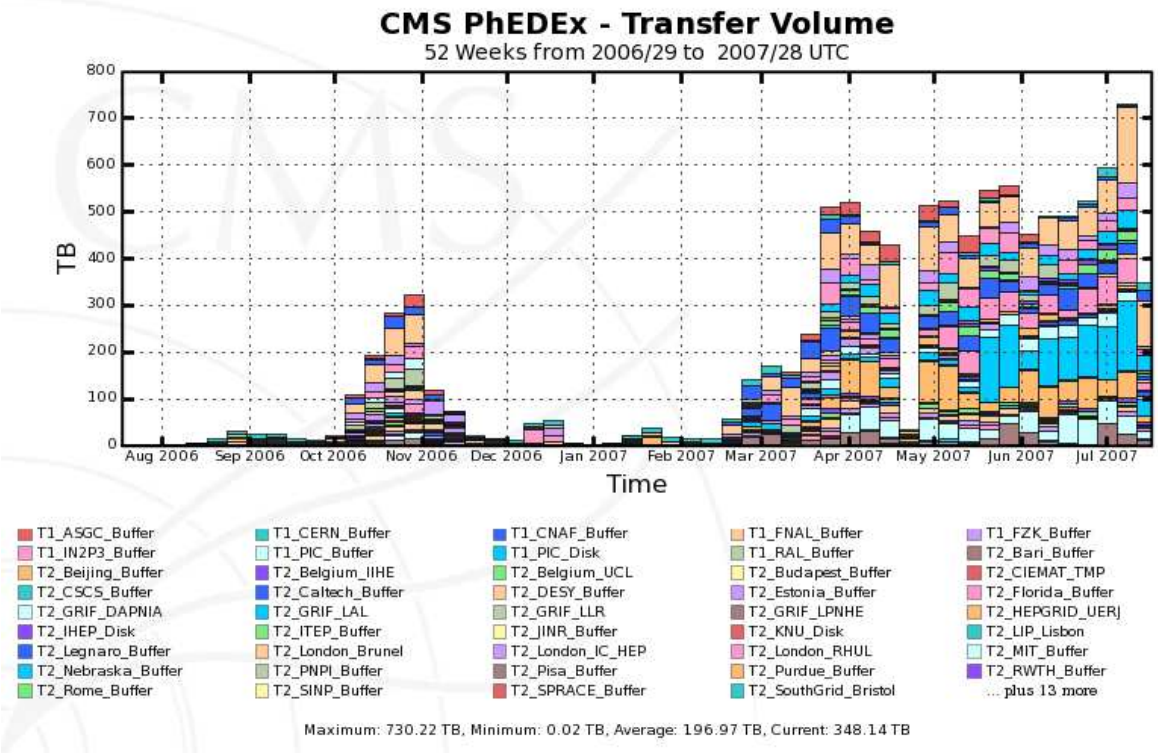


Figura 6.2: Volumen de datos transferidos durante el último año de operaciones.

un pico de algo más de 350 MB/s. Durante el CSA06, el valor medio que se alcanzó fue de unos 125 MB/s. Durante los últimos meses, sin embargo, el valor medio es superior a los 160 MB/s, compatible con las necesidades que el modelo de computación establece para este tipo de operaciones (fijado en 300 MB/s durante el régimen de mayor luminosidad). En la figura de calidades se puede comprobar que algunos centros (como RAL, CNAF y ASCG) han sufrido etapas donde el rendimiento era peor. Estos regímenes de peor calidad han correspondido, principalmente, a las operaciones de migración en el sistema de almacenamiento de datos. El paso de CASTOR-1 a CASTOR-2 no ha sido trivial, pues el sistema se ha hecho mucho más complejo (esencialmente para atender las necesidades del Tier-0). Todas las operaciones de lectura y escritura se encolan usando LSF, hace uso de una base de datos ORACLE para el bokkeeping interno, etc. Se necesita, por tanto, personal experto para optimizar y mantener el sistema.

Para un centro Tier-1 nominal la tasa de transferencia de datos desde el CERN es de unos 40 MB/s (con una tasa global de 300 MB/s a repartir entre los 7 centros Tier-1). El PIC, al tener un tamaño entre 1/2 y 1/3 de un Tier-1 nominal, debe aceptar una tasa de entre 15 y 20 MB/s, valores que ya se han alcanzado durante el CSA06. Durante el período del CSA06 los datos fueron transferidos a un ritmo promedio de 22 MB/s, con una calidad del 97 % (ver sección 5.3.3). Durante los últimos meses la tasa promedio ha sido de 34 MB/s, con picos para algunos promedios diarios superiores a 80 MB/s.

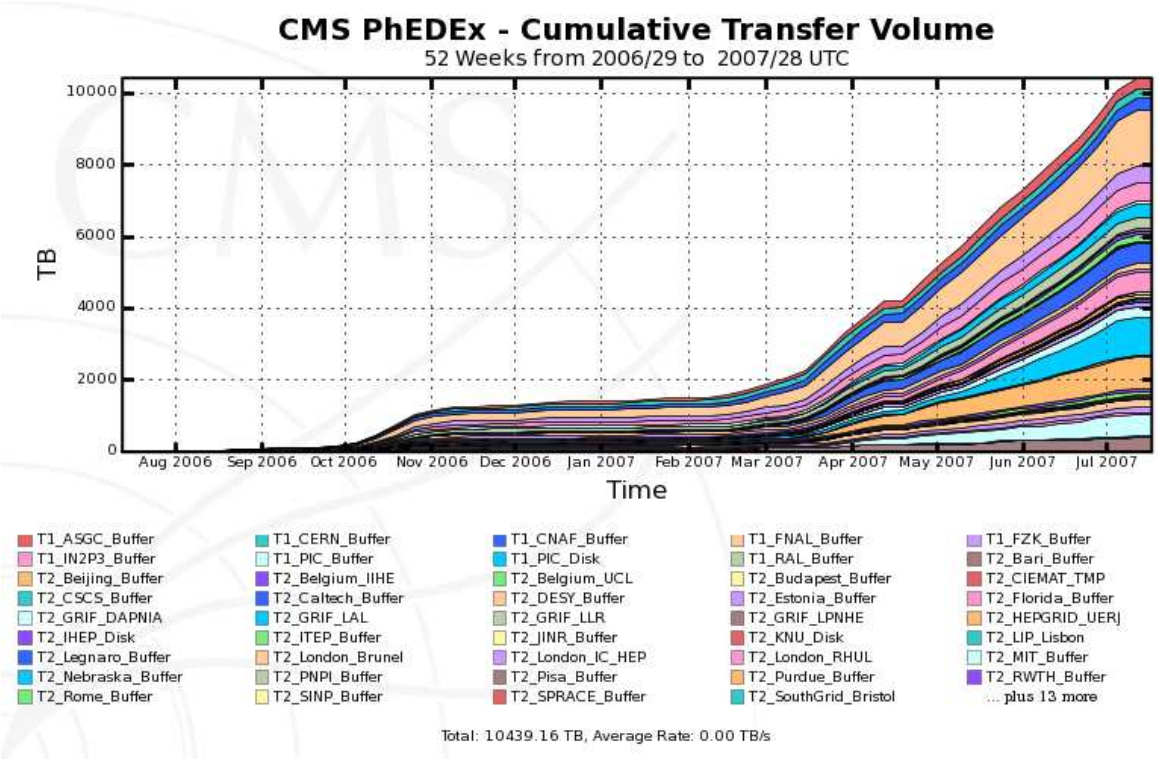


Figura 6.3: Volumen acumulado de datos transferidos durante el último año de operaciones.

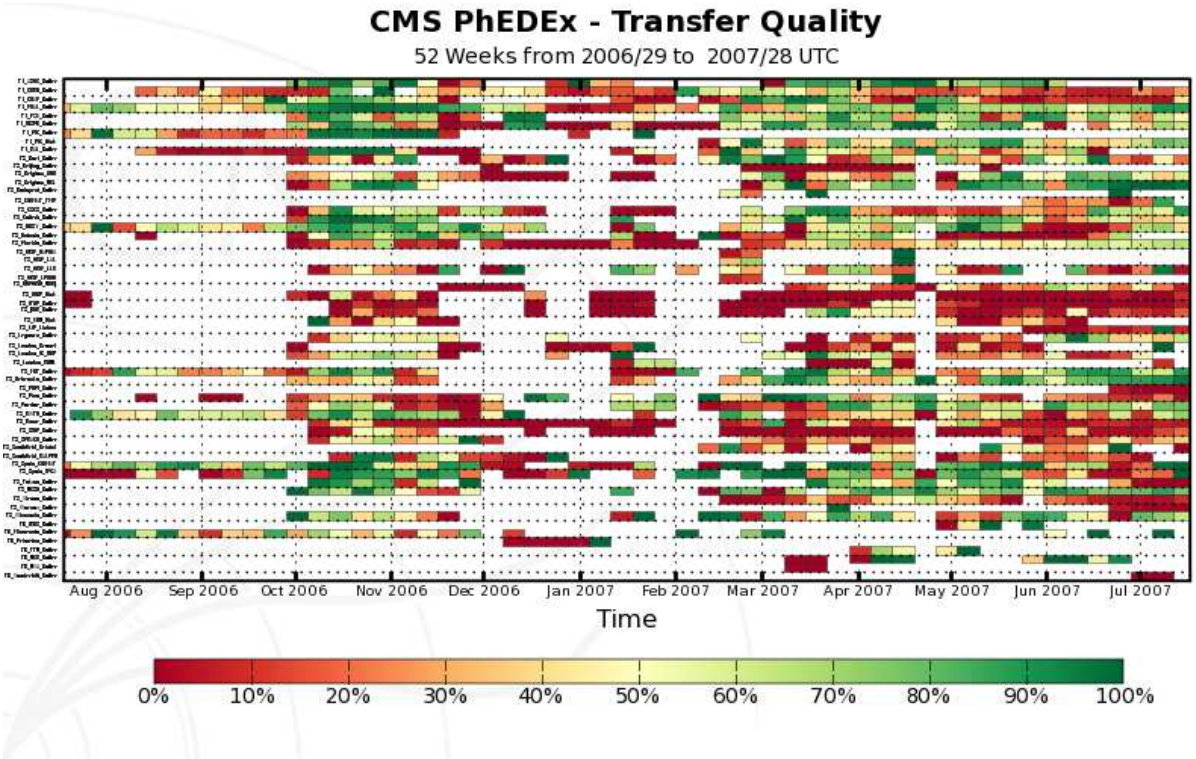


Figura 6.4: Calidad de las operaciones de transferencia de datos durante el último año.

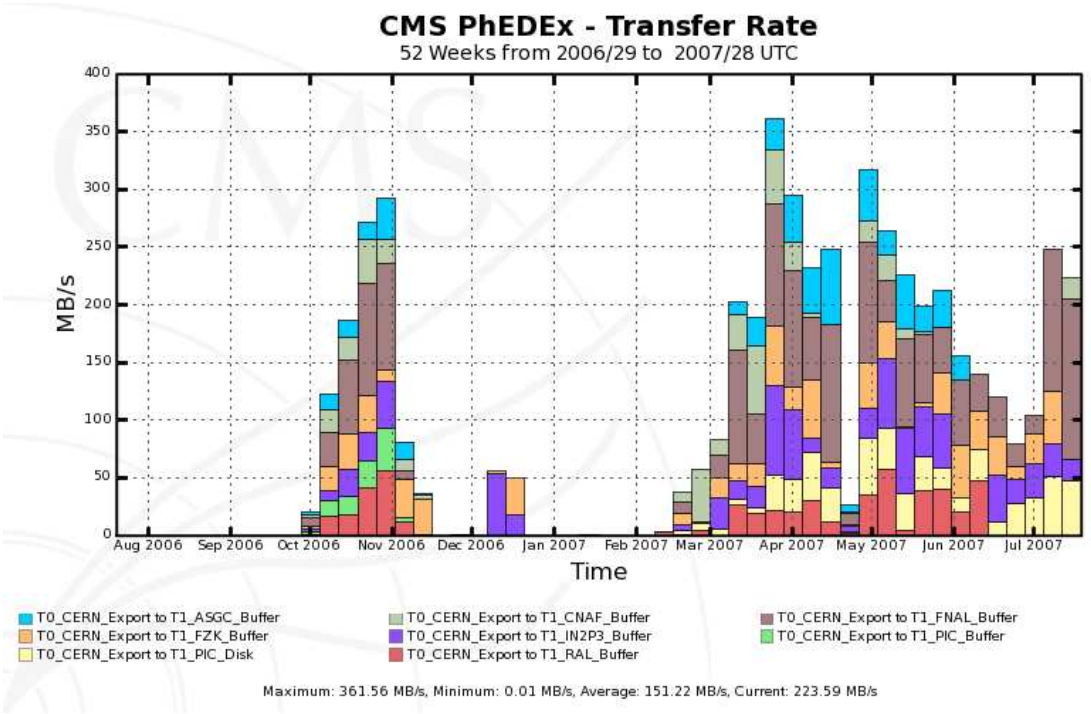


Figura 6.5: Tasa de transferencia de datos desde el CERN a los centros Tier-1 durante el último año.

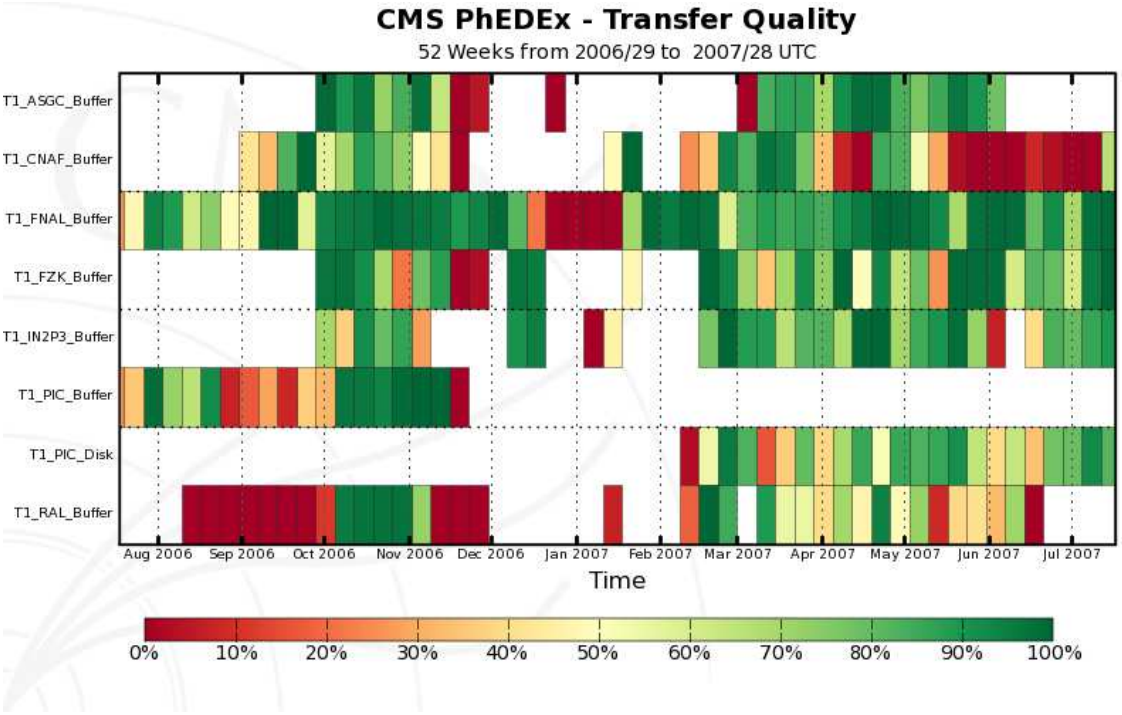


Figura 6.6: Calidad de las operaciones de transferencia de datos desde el CERN a los centros Tier-1 durante el último año.

6.1.2. Transferencias desde los centros Tier-1 a los centros Tier-2

En el caso de las transferencias de datos desde los centros Tier-1 a cada centro Tier-2, el modelo de computación especifica unas necesidades de 60 MB/s, aproximadamente. Se trata de muestras filtradas en los centros Tier-1 y enviadas a los centros Tier-2 para su análisis. Las figuras 6.7 y 6.8 muestran la tasa y calidad de estas operaciones de transferencia de datos durante el último año. El promedio semanal máximo que se ha alcanzado es superior a los 720 MB/s. Durante el mes y medio de duración del CSA06 el valor medio se situó alrededor de los 125 MB/s, y en los últimos meses se ha estabilizado en torno a los 500 MB/s. En general, la calidad de las operaciones es buena, especialmente cuando se realizan a un Tier-2 asociado al Tier-1 de origen pues son los enlaces mejor establecidos gracias a la cercanía regional (física y entre los administradores) de ambos centros. Sin embargo, durante las primeras tomas de datos, los análisis se llevarán a cabo sobre los RECO Data (repartidos entre los centros Tier-1). Por tanto, el rendimiento en las operaciones entre los centros Tier-1 y centros Tier-2 no asociados debe mejorar significativamente.

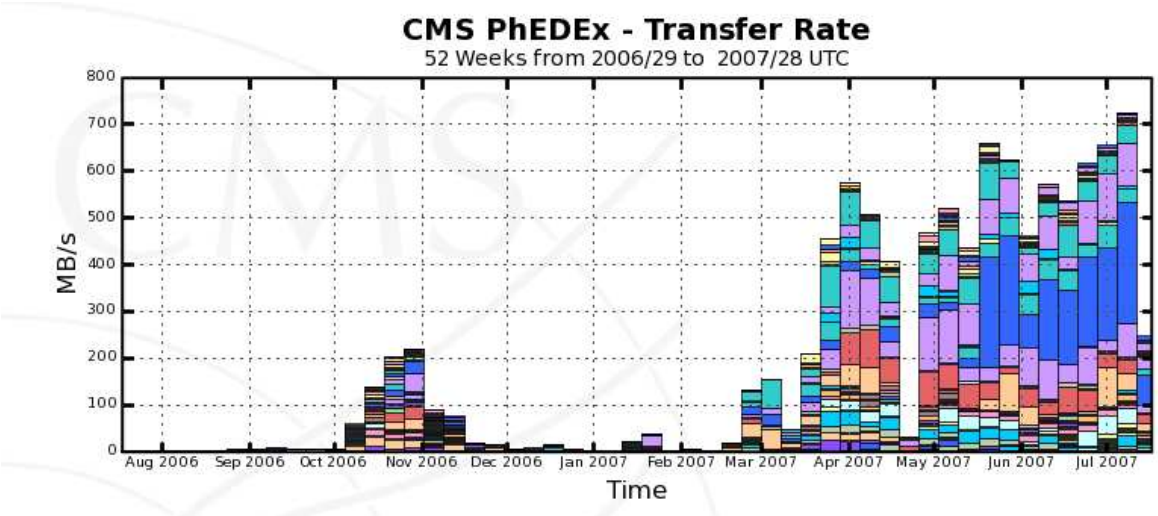


Figura 6.7: Tasa de transferencia de datos desde todos los centros Tier-1 a todos los centros Tier-2 durante el último año.

En el caso particular de los centros españoles, los resultados obtenidos durante el último año se pueden encontrar en los gráficos de la figura 6.9. Estas figuras muestran la velocidad de las transferencias, el volumen acumulado de datos movidos y la calidad de las operaciones, respectivamente.

En el caso del CIEMAT, el promedio semanal máximo alcanzado es de casi 35 MB/s. Durante las operaciones del CSA06 el valor promedio fue de 6-7 MB/s, con un promedio semanal máximo de 14 MB/s. En los últimos meses, las transferencias al CIEMAT se realizan con una tasa promedio de unos 15 MB/s. Hay que tener en cuenta que estos valores promedios corresponden a períodos semanales y, por tanto, la capacidad de transferencia instantánea es bastante mayor. De hecho, en las transferencias desde el PIC al CIEMAT se ha alcanzado rutinariamente una velocidad de 50 MB/s durante varios días. En total se han transferido algo más de 170 TB.

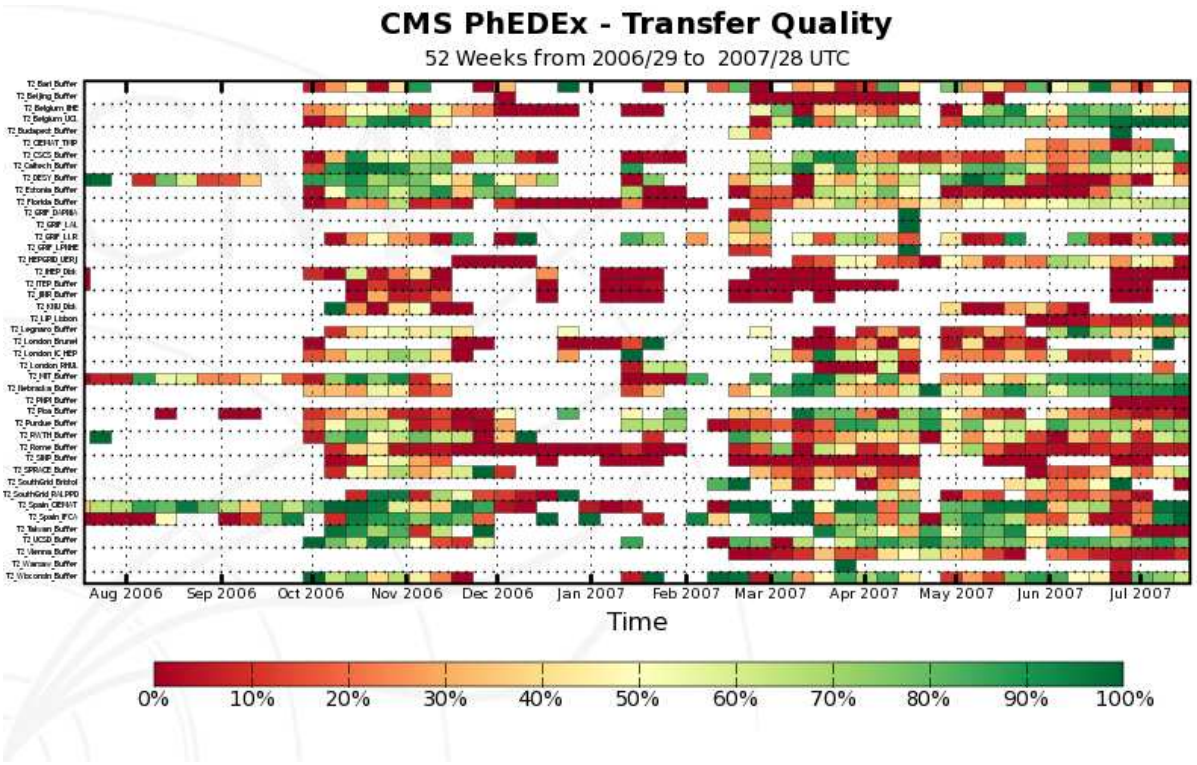


Figura 6.8: Calidad de las operaciones de transferencia de datos desde todos los centros Tier-1 a todos los centros Tier-2 durante el último año.

Para las transferencias hacia el IFCA, el valor promedio durante el último año ha sido de algo más de 6 MB/s (este valor también se ha duplicado a partir de marzo), con un promedio semanal máximo de 17 MB/s, aproximadamente. El volumen total de datos transferidos ha sido ligeramente superior a 100 TB.

En general, la calidad de estas operaciones es buena.

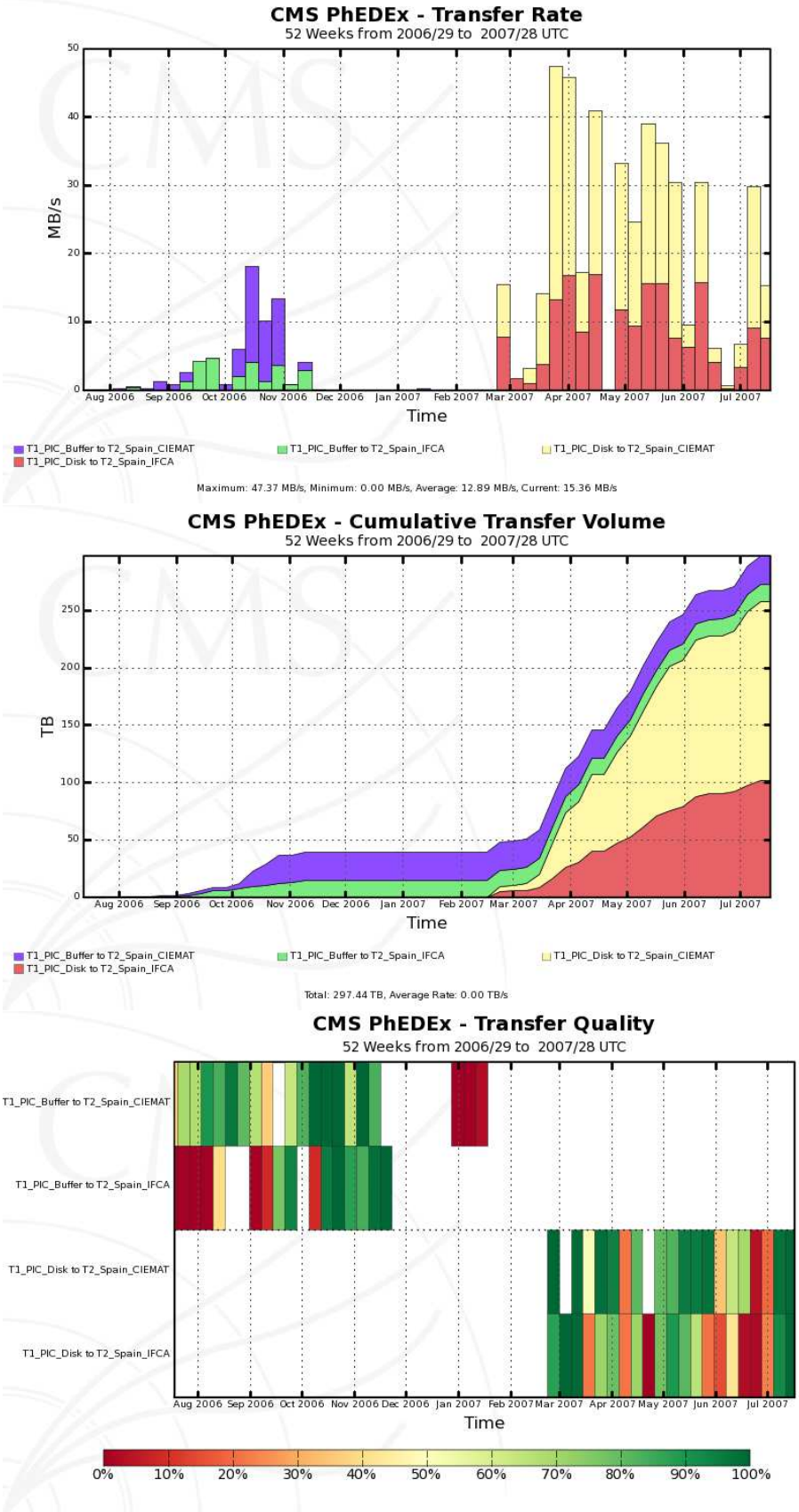


Figura 6.9: Tasa de transferencia de datos (arriba), volumen acumulado de datos transferidos (centro) y calidad de las operaciones de transferencia de datos (abajo) desde el PIC a los centros Tier-2 españoles durante el último año. T1_PIC_Buffer y Tier1_PIC_Disk son dos denominaciones para el mismo sistema de almacenamiento en el PIC que ha cambiado de nombre al introducir dCache junto con CASTOR.

6.1.3. Transferencias desde los centros Tier-2 a los Tier-1

Las transferencias desde los centros Tier-2 a los Tier-1 asociados también están contempladas en el modelo de computación de CMS, principalmente para copiar los resultados de las simulaciones de Monte Carlo. Los gráficos de la figura 6.10 muestran la tasa y calidad, respectivamente, de estas operaciones durante el último año. Corresponden a las transferencias ejecutadas desde todos los centros Tier-2 a los los Tier-1. El promedio semanal máximo alcanzado es ligeramente superior a 165 MB/s. Durante el CSA06 no se practicaron estas transferencias. Durante los 3 últimos meses, el valor promedio que se está alcanzado es de unos 100 MB/s, con un máximo superior a 225 MB/s. Se puede ver que las transferencias han sido especialmente delicadas en algunos casos (como el RAL), mientras que en el caso de otros T1 (como FNAL) la calidad ha sido sensiblemente mejor. Los resultados de la figura están principalmente dominados por las transferencias regionales, y FNAL y PIC son los centros más avanzados en este aspecto. Por el momento, las transferencias entre centros no asociados son marginales, y están en fase de depuración. El modelo computacional especifica que cada centro T1 debe estar preparado para recibir, aproximadamente, 1 TB de datos diario de cada centro T2 asociado. Se puede ver que el Tier-1 español cumple con esta especificación.

Los gráficos de la figura 6.11 muestran los resultados para el caso concreto del T2 español. El valor promedio en la tasa de transferencia desde los centros españoles es, aproximadamente, un 16 % con respecto al total. El valor promedio semanal máximo alcanzado ha sido de unos 19 MB/s, con picos diarios mucho más altos. La figura inferior muestra el volumen acumulado de datos transferidos desde el T2 español. En total, durante el último año, se han copiado casi 95 TB de datos.

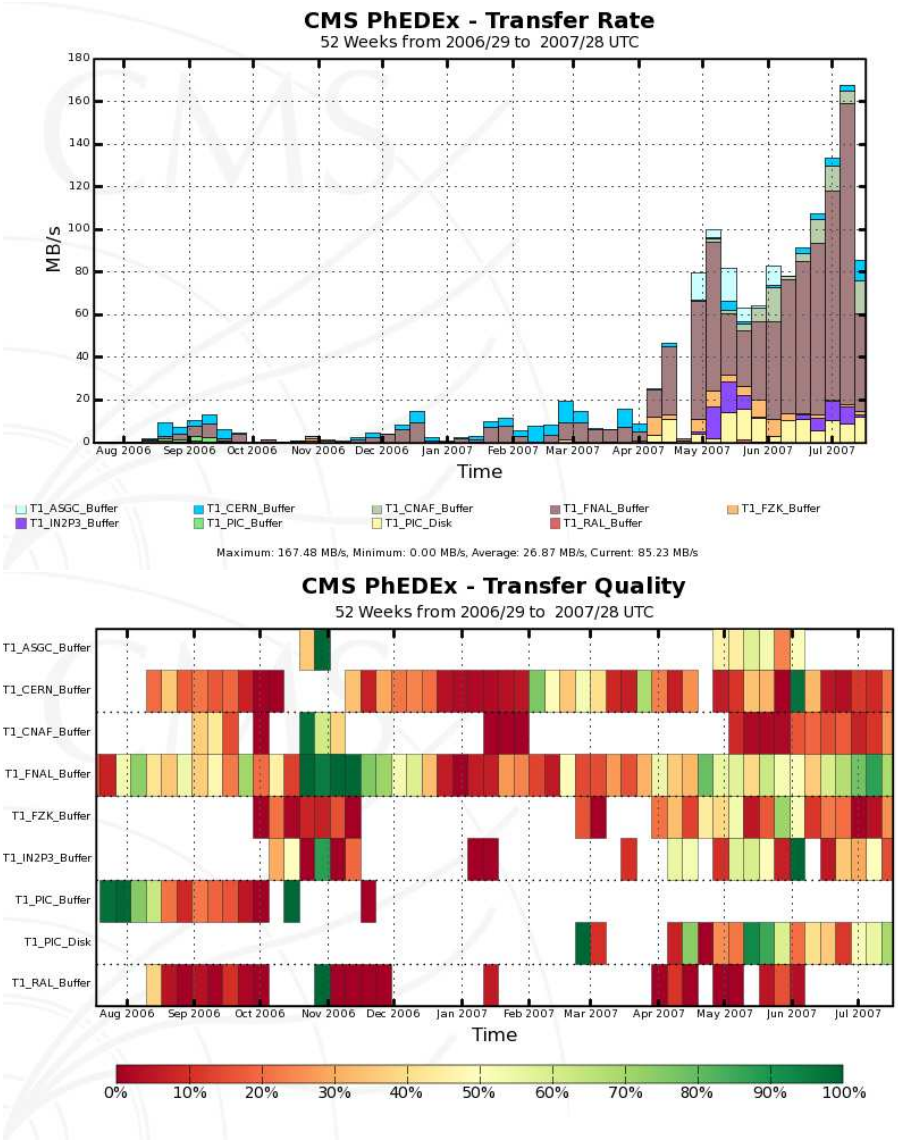


Figura 6.10: Tasa de transferencia de datos (arriba), y calidad de las operaciones (abajo), desde los centros Tier-2 a los Tier-1 durante el último año.

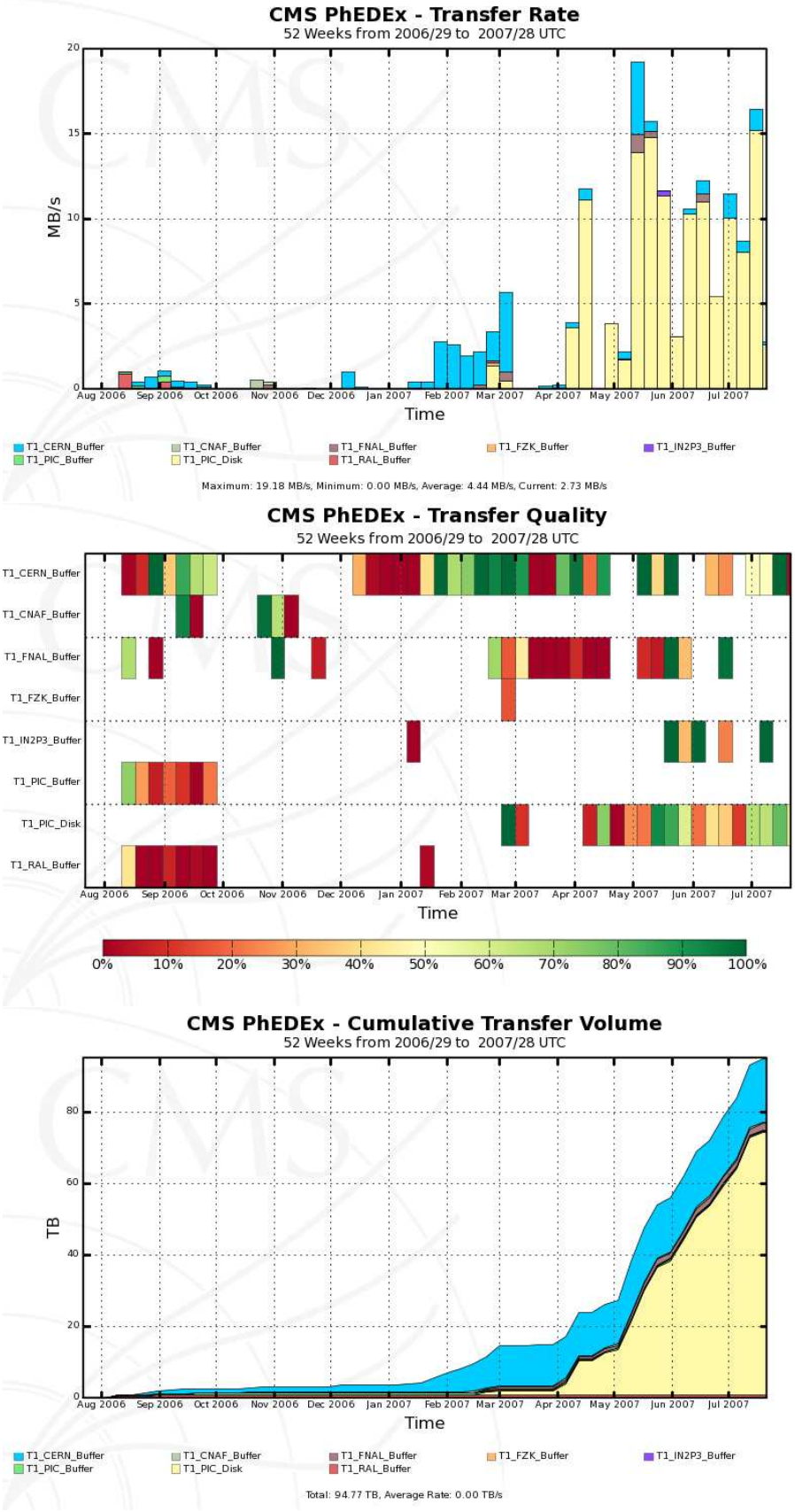


Figura 6.11: Tasa de transferencia de datos (arriba), calidad de estas operaciones de transferencia (centro) y volumen acumulado de datos transferidos (abajo) desde el centro Tier-2 español a los Tier-1 durante el último año.

6.1.4. Transferencias entre centros Tier-1

Finalmente, en el modelo de computación también se contemplan los movimientos de datos entre los distintos centros Tier-1. El objetivo principal es la distribución de los datos en formato AOD después de ser re-procesados en cada centro. Cada centro Tier-1 sólo tiene una fracción de los datos reconstruidos en formato RECO, mientras que cada centro mantiene una copia completa de los datos en formato AOD. Esto impone la necesidad de distribuir los datos cada vez que se ejecuta una nueva reconstrucción. Esto sucede unas pocas veces al año y estas transferencias son de naturaleza bursty, pues el objetivo es realizarlas a la mayor velocidad posible. La figura 6.12 muestra los resultados conseguidos con este tipo de operaciones durante el último año. El valor máximo alcanzado es de unos 36 MB/s. Estos valores están aún lejos de alcanzar las especificaciones del modelo de computación. Este tipo de operaciones se han empezado a ejercitar en los últimos meses, pero a baja velocidad, pues las transferencias desde el CERN y entre centros Tier-1 y Tier-2 han sido prioritarias.

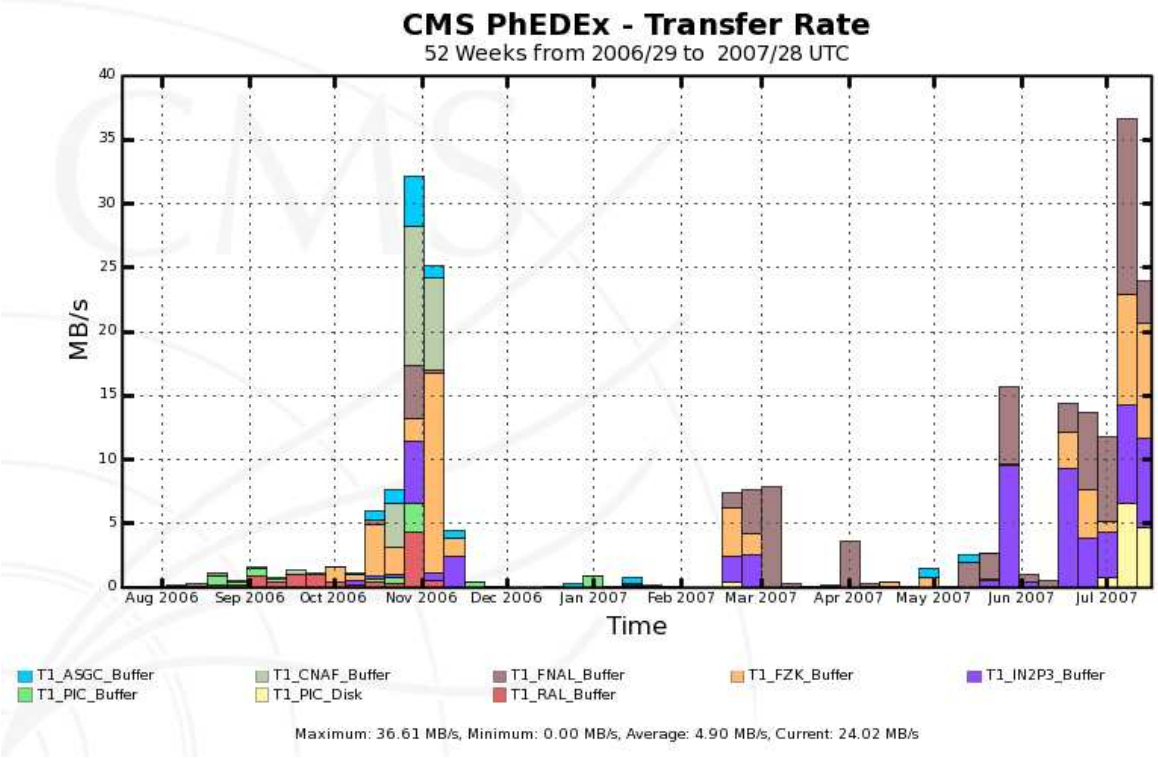


Figura 6.12: Tasa de transferencia de datos entre centros Tier-1 durante el último año. El tráfico está agrupado por el Tier-1 de destino, sumando las transferencias desde los demás centros.

6.1.5. Transferencias simultáneas para varias organizaciones virtuales

En la figura 6.13 se pueden comparar las operaciones de transferencia de datos realizadas desde el CERN durante el último año para varias organizaciones virtuales. La figura 6.14 muestra los mismos resultados para las transferencias entre el CERN y el PIC. En general, el volumen de datos transferidos por CMS es ligeramente superior a los demás, seguido muy de cerca por ATLAS. En el caso de las transferencias desde el CERN al PIC la VO con mayor actividad es CMS, al menos durante los últimos meses.

Una de las funciones del servidor FTS es garantizar un reparto del ancho de banda disponible en los diferentes enlaces. Cuando más de una VO ha estado transfiriendo datos suficientes se ha comprobado que el reparto ha sido el esperado. Esto garantiza que una VO individual no cope todo el ancho de banda

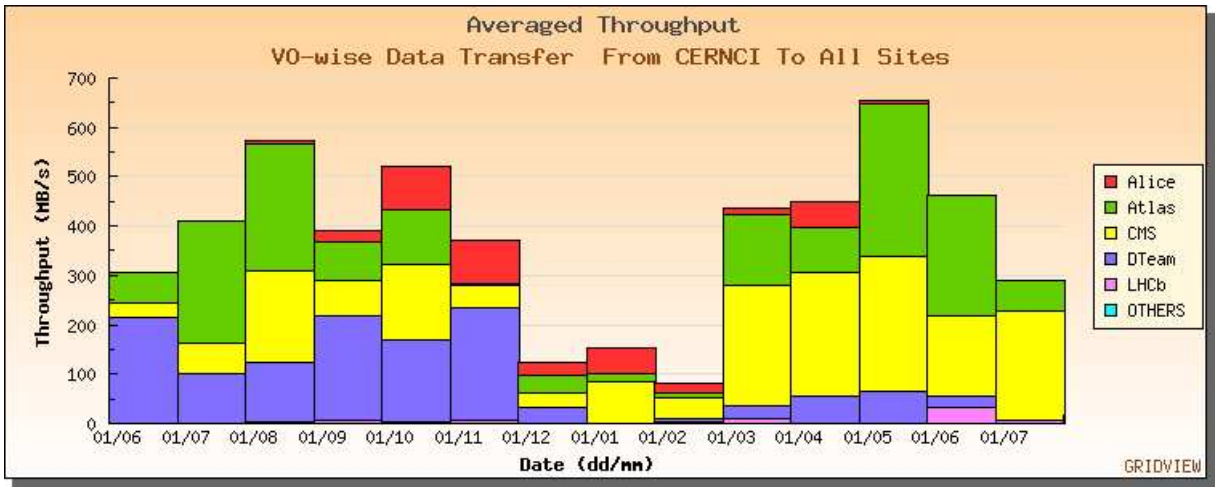


Figura 6.13: Tasa de transferencias de datos para distintas VOs, desde el CERN a todos los Tier-1, durante el último año.

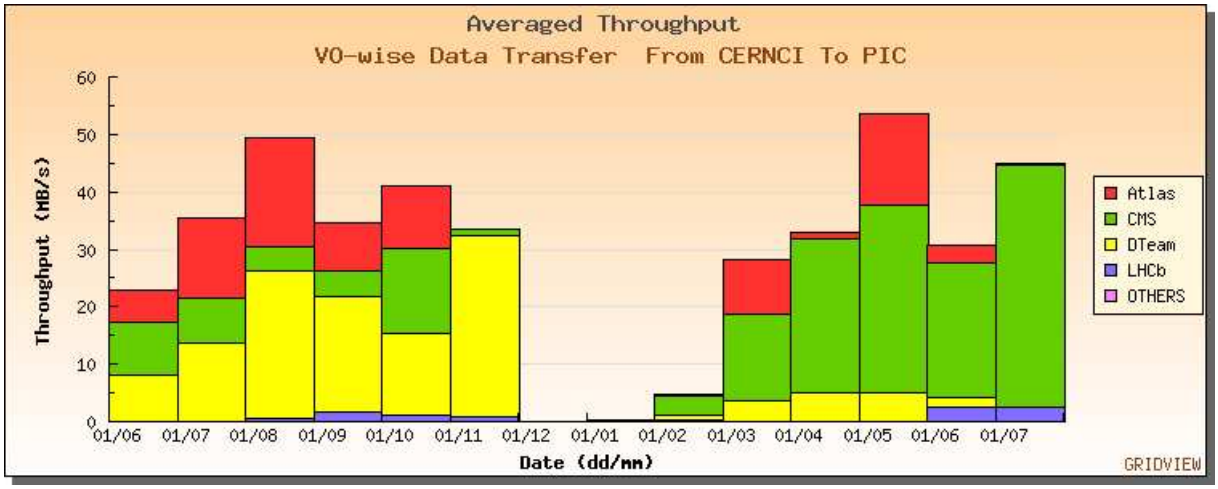


Figura 6.14: Tasa de transferencias de datos para distintas VOs, desde el CERN al PIC, durante el último año.

cuando otra desea transferir datos. Cuando solo una VO está transfiriendo datos ésta puede hacer uso de todo el ancho de banda disponible.

6.2. Gestión de trabajos

En esta sección se exponen los resultados globales alcanzados en la gestión de los distintos tipos de trabajos: reconstrucción en el Tier-0, producción Monte Carlo, reprocesamiento de datos, y operaciones de filtrado y análisis. Para la gestión de todas estas operaciones, con la excepción de los trabajos de análisis de los usuarios, se ha usado ProdAgent 4.2. La reconstrucción, el reprocesamiento y el filtrado de datos se han ejercitado durante los Data Challenges. Existe una actividad de producción continua, al igual que de análisis de los usuarios, aunque ésta última no es aún elevada. Para compensarlo se ejecutan análisis ficticios a través de JobRobot de forma continua.

Para el caso de la gestión de trabajos, las dos principales fuentes de información son la base de datos

de ProdAgent y el Dashboard. La primera recoge toda la información relativa a la producción Monte Carlo. En el segundo caso se incluyen también las actividades de análisis, los trabajos de test enviados por JobRobot, los trabajos de instalación de software, y registra las actividades de monitorización.

La figura 6.15 muestra, ordenados por actividades, el número de trabajos enviados al Grid, su estado, y el resultado de los que ya han acabado (con éxito, fallido o desconocido). Los resultados que se muestran corresponden al último año de operaciones. Las figuras 6.16 y 6.17 muestran la misma información para el CIEMAT y PIC, respectivamente. En total, Dashboard ha registrado actividad para 4 millones de trabajos de producción y algo más de 3 millones de trabajos de análisis. Estos 3 millones se dividen, a partes iguales prácticamente, entre los trabajos de análisis reales y los trabajos de análisis simulado a través de JobRobot. En el caso del CIEMAT se contabilizan unos 65000 trabajos de producción y casi 90000 de análisis. En ambos casos, el porcentaje de trabajos finalizados con éxito es bastante satisfactorio. En el caso del PIC las cifras son ligeramente menores, con unos 40000 trabajos de producción y algo más de 60000 trabajos de análisis.

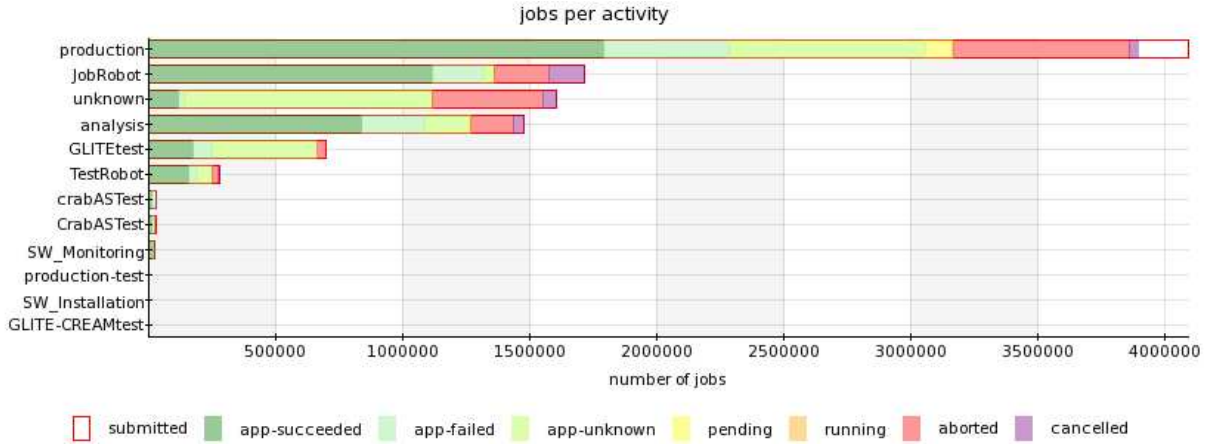


Figura 6.15: Número de trabajos, ordenados por actividades, enviados a todos los centros durante el último año.

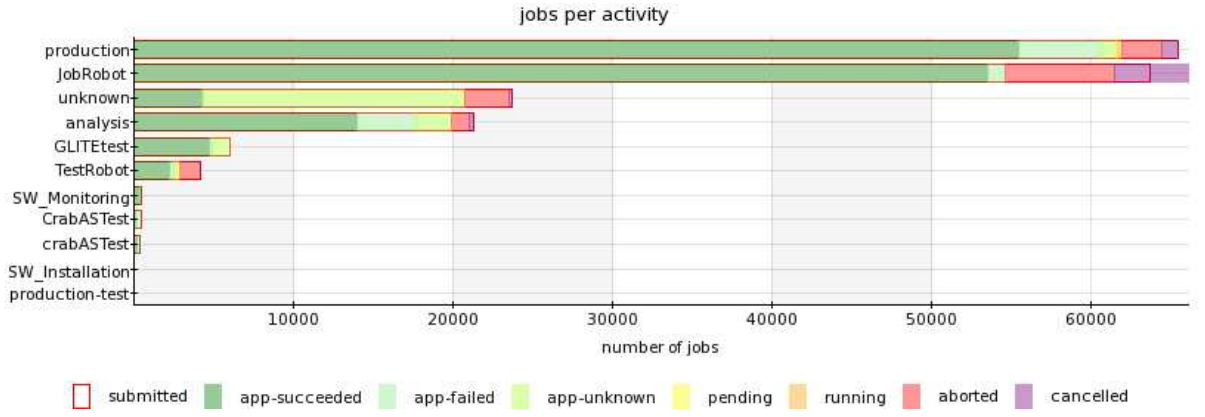


Figura 6.16: Número de trabajos, ordenados por actividades, enviados al CIEMAT durante el último año.

La figura 6.18 muestra la distribución de trabajos registrados en Dashboard durante el último año, ordenada por centros. FNAL y CERN son los centros que han recibido mayor número de trabajos, aproximadamente 1 millón cada uno de ellos. CIEMAT ocupa el octavo puesto de la lista.

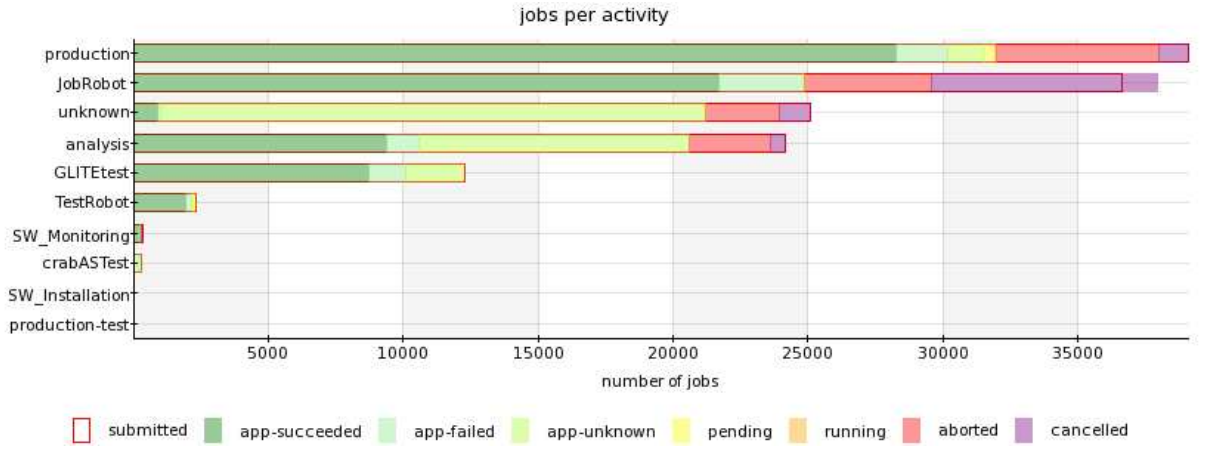


Figura 6.17: Número de trabajos, ordenados por actividades, enviados al PIC durante el último año.

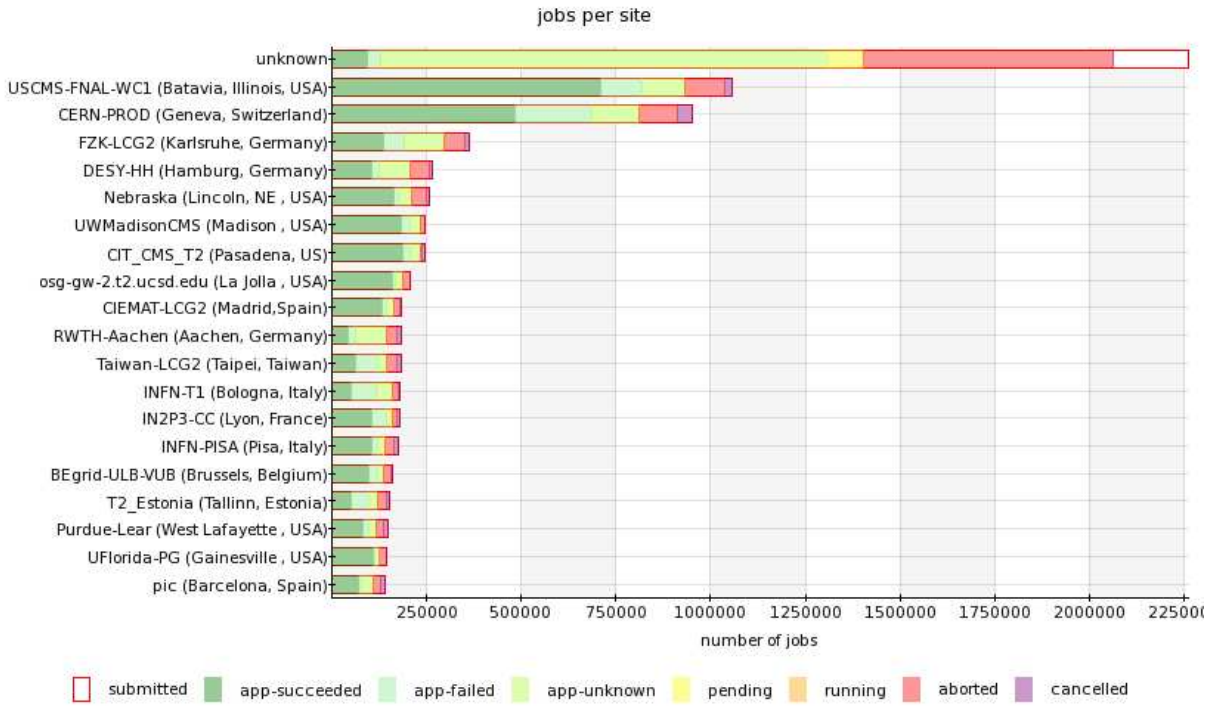


Figura 6.18: Distribución de trabajos enviados al Grid, ordenados por centros, durante el último año.

Finalmente, la figura 6.19 muestra los distintos tipos de trabajos enviados al Grid, para un período de casi un año de operaciones. Se pueden distinguir los trabajos de producción Monte Carlo, de análisis, de análisis ficticios gestionados con JobRobot para estresar los sistemas, trabajos de test, de instalación del software, etc. Las principales contribuciones son las actividades de análisis y del JobRobot durante las operaciones del CSA06, y las actividades de producción Monte Carlo a partir de la navidad del 2006. Se puede apreciar claramente una subida continuada en el volumen de operaciones, llegando a una valor actual de, aproximadamente, unos 100000 trabajos cada 6 días.

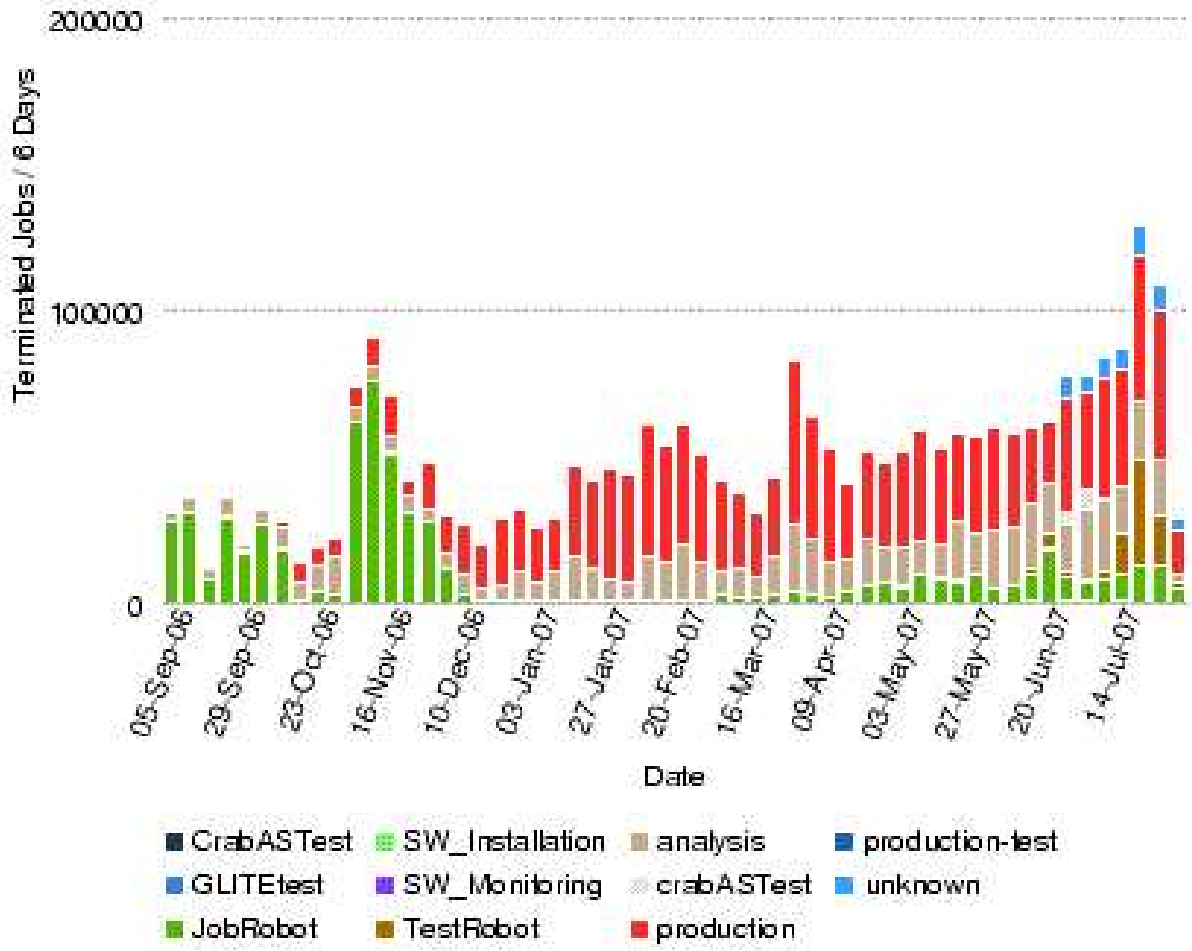


Figura 6.19: Distintos tipos de trabajos enviados en función del tiempo. Cada bin corresponde a un período de 6 días.

6.2.1. Trabajos de producción Monte Carlo

La información contenida en la base de datos global de ProdAgent corresponde a los últimos cuatro meses de operaciones. De nuevo, se puede apreciar un aumento considerable en las tareas de producción correspondiente a la pre-producción para el CSA07.

Se han producido 160 millones de sucesos durante los últimos 4 meses (ver figura 6.20), lo que correspondería a una media de 40 millones de sucesos simulados cada mes. Sin embargo, casi el 90 % de la producción se ha ejecutado en dos meses, por lo que la tasa real de simulación Monte Carlo de CMS ha sido de unos 70 millones de sucesos por mes. En la figura se puede ver también que el CIEMAT es el cuarto centro de la colaboración CMS en volumen de datos simulados, y el primero entre los centros Tier-2.

El CIEMAT ha realizado en torno al 9 % de la producción total y en torno al 20 % de la producción realizada en los centros Tier-2. Este resultado es muy superior al valor nominal del 5 % de los recursos aportados por un Tier-2 nominal. El PIC, a pesar de ser un Tier-1 y no tener, por tanto, responsabilidades de producción, ha contribuido con la simulación del 2.5 % de los sucesos. Esto ha sido gracias al hecho de que había recursos disponibles pues los procesamientos de datos típicos de un Tier-1 (filtrado y reprocesa-

meinto) sólo se han ejercitado en momentos puntuales (como los Computing Challenges). Finalmente, el IFCA ha contribuido con 0.3 % de los sucesos, y está en proceso de incorporar mayor potencia de cálculo.

Los dos primeros centros en nivel de operaciones son FNAL y CERN, como también se puede ver en la figura 6.21.

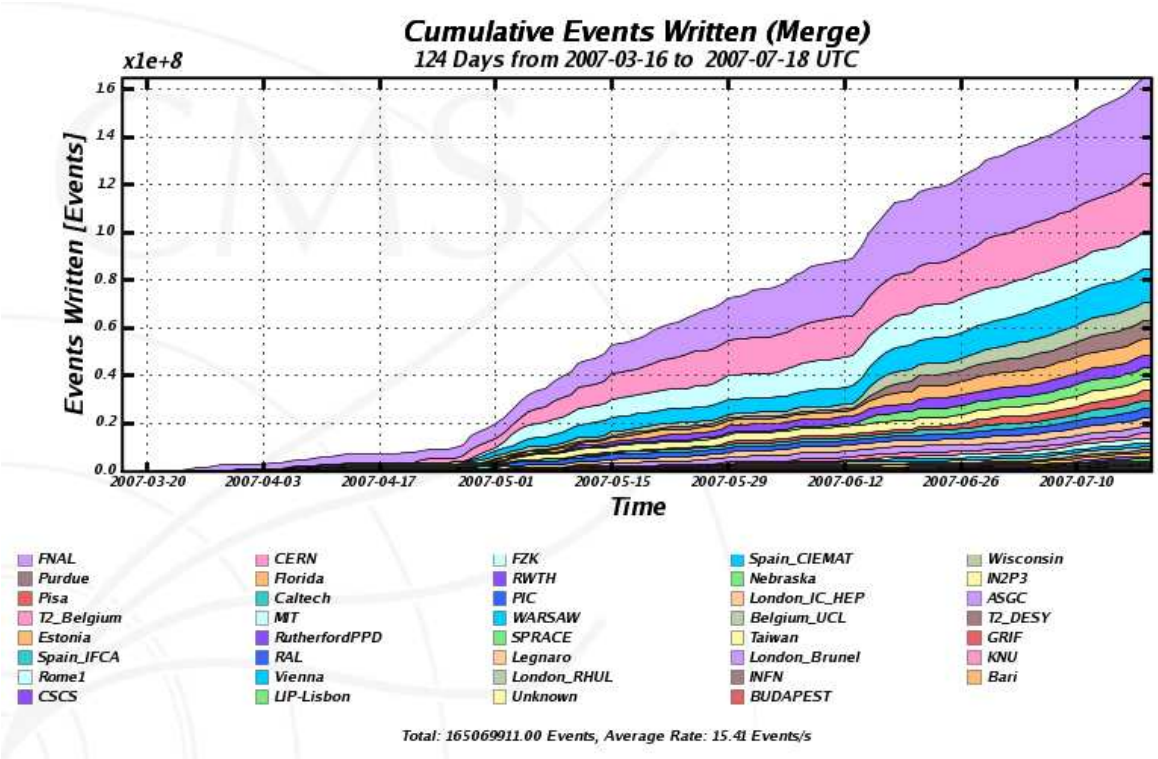


Figura 6.20: Número acumulado de sucesos Monte Carlo producidos durante los últimos 4 meses.

La figura 6.22 muestra el número de CPUs que están completamente ocupadas (las 24 horas del día) con trabajos de producción Monte Carlo. Para averiguar este valor se divide el número total de horas empleadas en producción entre 24, y el resultado da cuenta del número de nodos que, en promedio, se están usando de forma eficiente. Es destacable el valor máximo que se ha alcanzado, superior a los 6500 trabajos en un sólo día. Durante los 2 últimos meses el ritmo es de unos 5000 trabajos en ejecución por día. Estos datos demuestran que CMS ha sido capaz de desarrollar herramientas que le permiten hacer un uso cada vez más eficiente de todos los recursos Grid disponibles. Sin embargo, parece que, en el estado actual, es difícil superar este valor medio de 5000 trabajos diarios, y las actividades de producción siguen necesitando grandes cantidades de recursos humanos, por lo que se debe seguir trabajando en la automatización de las tareas.

La figura 6.23 muestra la distribución de horas de procesamiento por centros. En total, se han contabilizado más de 5 millones de horas de procesamiento (casi 600 años), siendo FNAL y CERN, de nuevo, los principales contribuyentes. El centro del CIEMAT es el sexto, con algo más de 240000 horas de trabajo.

La figura 6.24 muestra la tasa de éxito de los trabajos de producción en los últimos 4 meses para todos los centros que contribuyen en las tareas de producción. En la figura se pueden ver qué centros colaboran, sus respectivos períodos de actividad, y el porcentaje de éxitos. Se puede ver que el comportamiento de los centros españoles es muy bueno y que en general, salvo en centros concretos, la eficiencia es elevada.

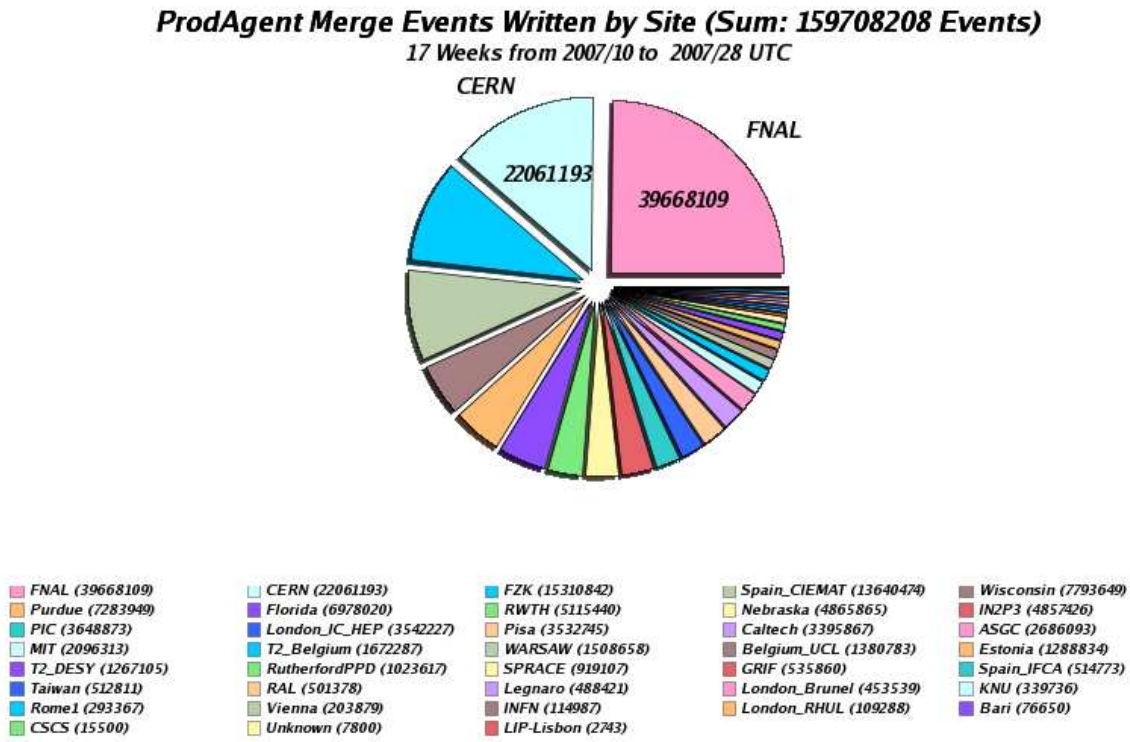


Figura 6.21: Distribución por centros de las operaciones de producción Monte Carlo durante los últimos 4 meses.

En la figura 6.25 se pueden ver los distintos códigos de retorno de los trabajos de producción Monte Carlo finalizados cada día durante los últimos meses. Aparte de la escala alcanzada (del orden del 20000 trabajos), se puede ver que, efectivamente, la mayoría de los trabajos finalizan correctamente (código 0) y que un porcentaje significativo de los trabajos fallidos devuelven código -1, que significa que la causa de fallo estuvo en los servicios Grid.

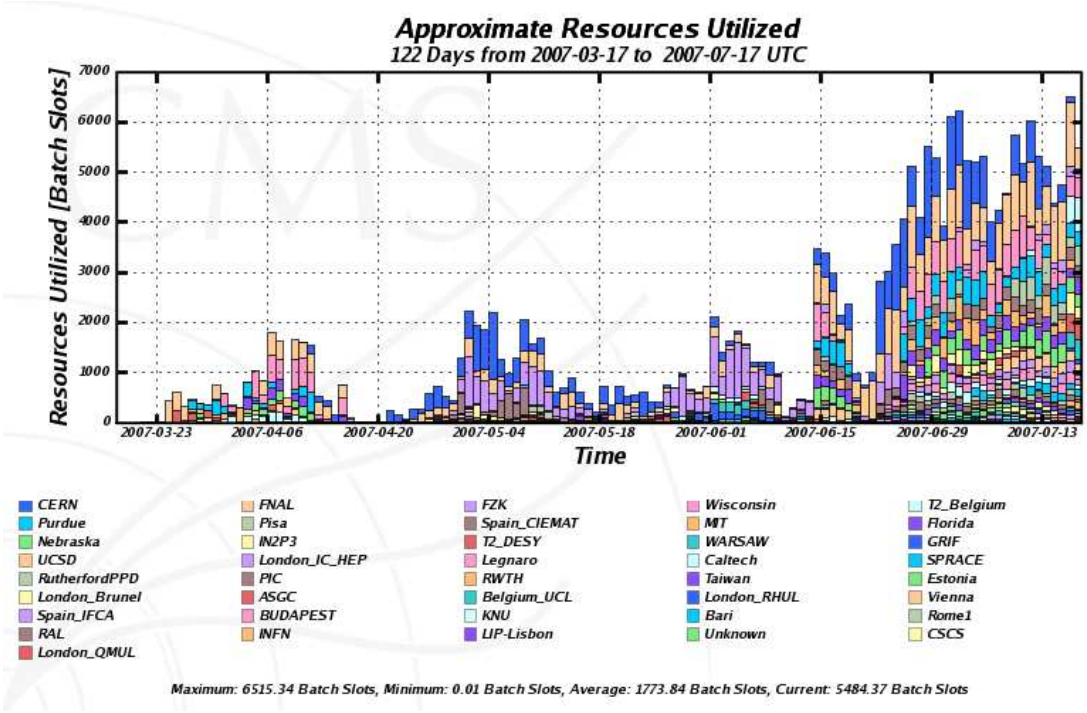


Figura 6.22: Número de CPUs ocupadas las 24 h con trabajos de producción Monte Carlo, durante los últimos 4 meses.

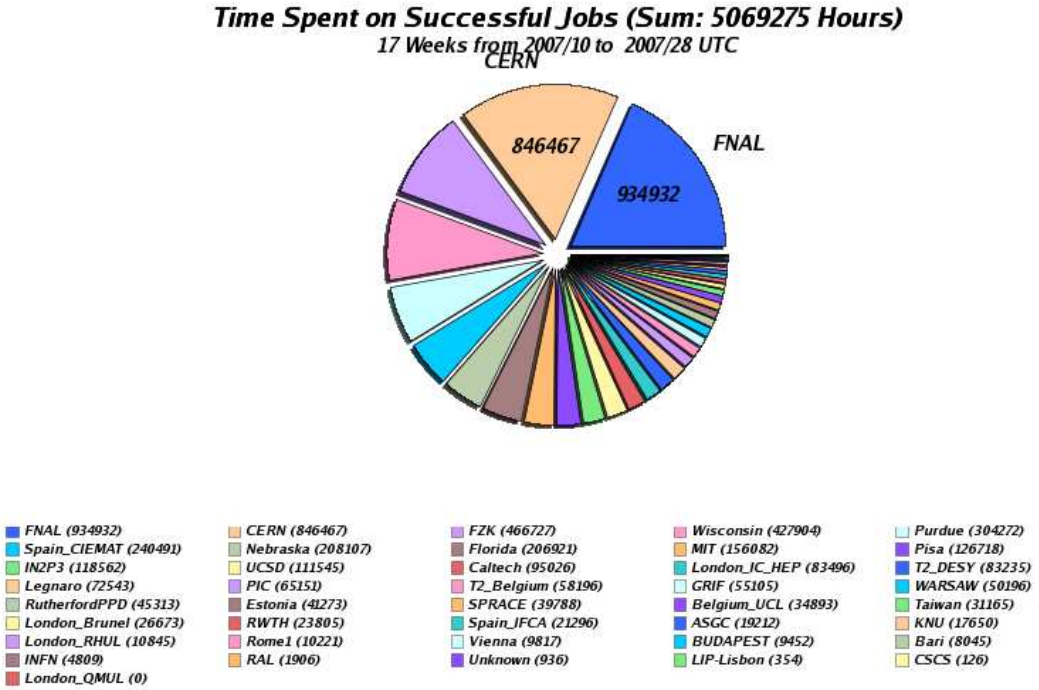


Figura 6.23: Distribución por centros del tiempo empleado en las operaciones de producción Monte Carlo durante los últimos 4 meses.

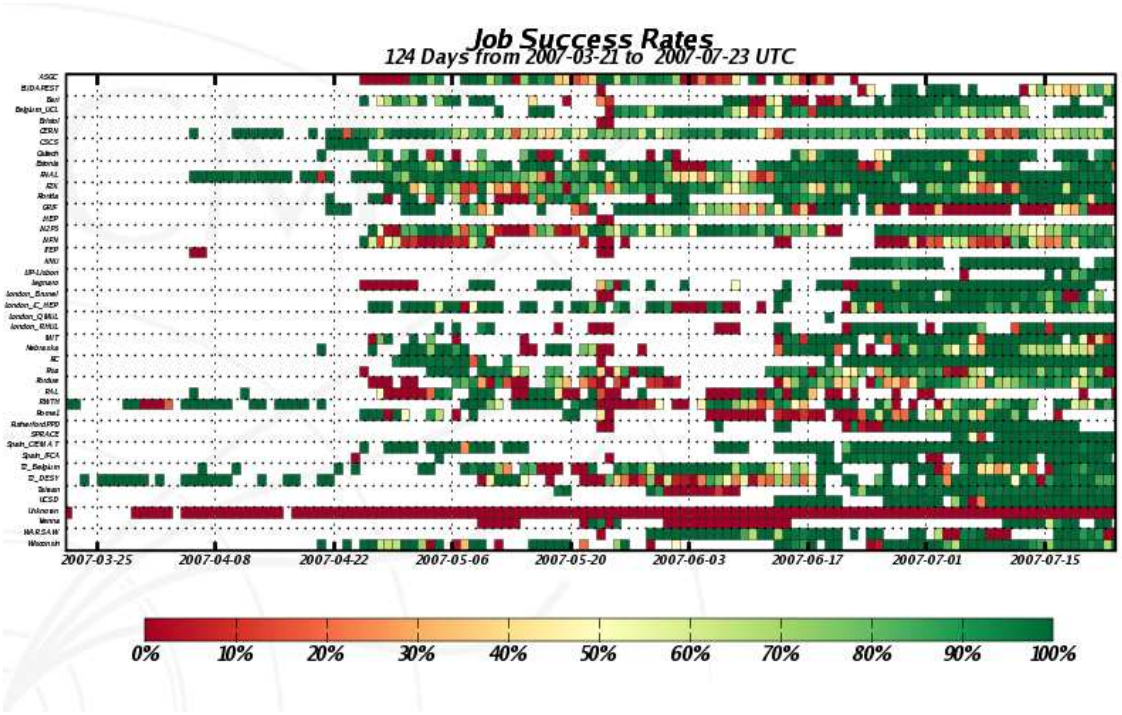


Figura 6.24: Tasa de éxito de los trabajos de producción durante los últimos 4 meses, para todos los centros que colaboran en las tareas de producción Monte Carlo.

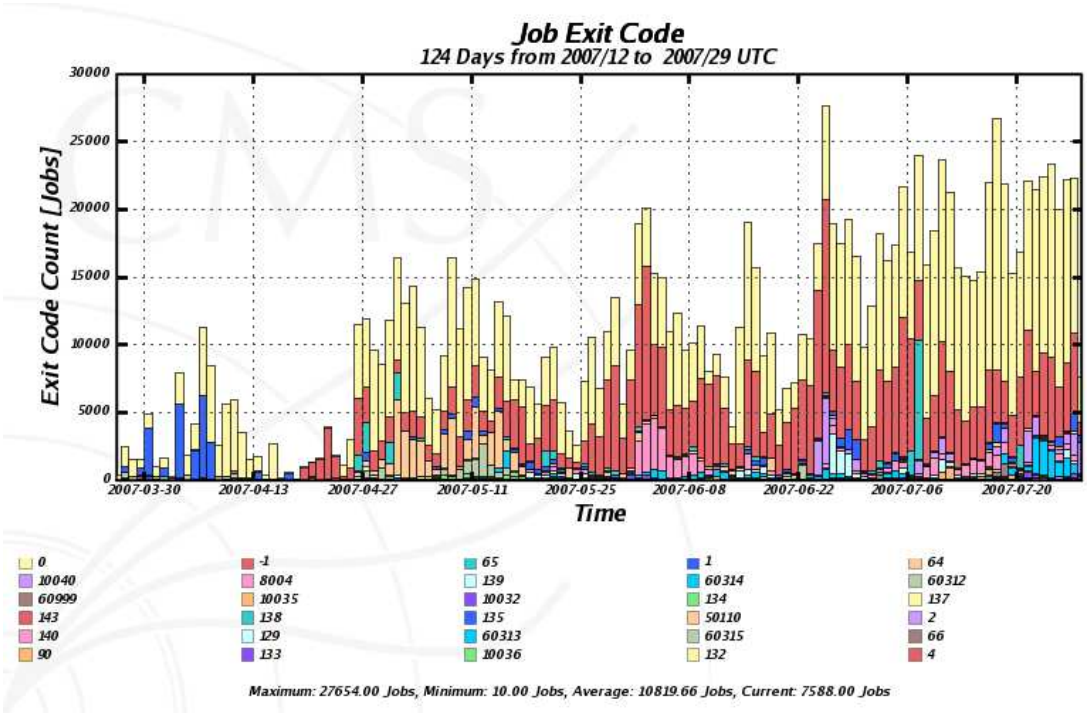


Figura 6.25: Trabajos de producción Monte Carlo finalizados cada día durante los últimos meses, distinguiendo el código de retorno.

Conclusiones

Para poder llevar a cabo los objetivos de física del nuevo acelerador del CERN, el LHC, será necesaria una cantidad sin precedentes de recursos de computación para recoger, transferir, almacenar, procesar y analizar la ingente cantidad de datos que se producirán anualmente. Esta cantidad de datos, del orden de varios Petabytes por año, es el resultado de la enorme luminosidad del acelerador (que llegará hasta un valor de 10^{34} $cm^{-2}s^{-2}$), un potente sistema de filtrado online en el experimento (capaz de analizar una tasa de colisiones de 1 GHz y de reducir en más de cinco órdenes de magnitud la tasa de sucesos registrados por el detector) y una gran capacidad de almacenamiento de sucesos relevantes para el análisis ($\sim$150 Hz, lo que equivale a unos 300 MB/s) incluyendo los datos en formato RAW y RECO. Los modelos de computación clásicos de los experimentos de física de altas energías no resultan adecuados para gestionar este volumen de datos, y en el LHC se ha optado por una solución basada en la distribución geográfica de los recursos y los servicios de computación: las tecnologías Grid. El buen funcionamiento de este modelo de computación será crucial para el éxito del análisis de datos en CMS. La complejidad del sistema de computación para los experimentos de LHC es tal que se le considera como un experimento más, dadas sus dimensiones y recursos humanos y técnicos empleados.

En el modelo de computación de CMS los recursos se distribuyen de manera jerárquica en una estructura de niveles (Tiers) donde cada nivel de computación tiene asociadas funcionalidades concretas. El Tier-0 (en el CERN) almacena los datos registrados por el detector y los reconstruye. Los 7 centros del tipo Tier-1 reciben una fracción de esos datos y ejecutan un procesamiento organizado sobre ellos (pases de filtrado seleccionados por los grupos de física, y de reprocesamiento, unas pocas veces al año, cuando hay disponibles versiones mejoradas de las constantes de calibración y alineamiento o del código de reconstrucción). En los aproximdamente 25 centros Tier-2 los físicos analizarán los datos y se producirán las muestras simuladas de datos. España posee un centro Tier-1 (en el PIC) y un centro Tier-2 distribuido (CIEMAT e IFCA). Se ha dotado a estos centros de los recursos y servicios Grid necesarios para complir con su cometido. El PIC tiene actualmente instaladas unas 200 CPUs (aproximadamente 600 kSI2k) y 210 TB de espacio en disco y en cinta. El CIEMAT posee 400 CPUs (que ofrecen una potencia de cálculo de unos 700 kSI2k) y 35 TB de espacio. Estos recursos irán escalando en los próximos años hasta alcanzar, en 2010, una cantidad de recursos un orden de magnitud mayor de los disponibles actualmente. Cada año se duplicará, aproximadamente, la cantidad de recursos disponibles en cada centro. Se ha contribuido de forma directa en la puesta a punto de estos centros para que puedan ejecutar las tareas de computación de CMS, y cumplir con la contribución establecida para los centros españoles.

Se ha participado en el desarrollo de aspectos claves del sistema de computación de CMS. Se ha contribuido en el desarrollo del sistema de procesamiento de datos y del sistema de transferencia de datos. En el área de procesamiento de datos se ha portado el sistema de producción de sucesos simulados de Monte Carlo al Grid LCG, de modo que grandes recursos de computación pudieron emplearse en la simulación. Además, con la experiencia adquirida, se contribuyó en el desarrollo de un sistema mejorado de producción que se ha extendido para llevar a cabo el resto de tareas de procesamiento organizado del experimento, la reconstrucción en el centro Tier-0, el filtrado y reprocesamiento en los centros Tier-1, y la producción eficiente de Monte Carlo en los centros Tier-2. El nuevo sistema consigue un mayor nivel de automatización, facilidad de mantenimiento, escalabilidad, monitorización y gestión más eficiente de los recursos. Permite operar en varios entornos Grid. Además, integra el nuevo modelo de datos de CMS, los nuevos servicios de gestión de datos y el nuevo entorno de procesamiento de datos.

Se ha participado en el desarrollo de un sistema robusto, fiable y eficiente de transferencia de datos. Este sistema permite ejecutar los flujos de datos del modelo de computación de CMS (del Tier-0 a los centros Tier-1, de los Tier-1 a los Tier-2 y viceversa, y entre centros Tier-1). Este sistema selecciona el camino de transmisión de datos más adecuado entre dos puntos de acuerdo a las condiciones dinámicas de los enlaces, implementa las prioridades y políticas de distribución de datos del experimento, interacciona eficientemente con los sistemas locales de almacenamiento de los centros, y reintenta las transferencias en caso de fallo.

El sistema Grid de análisis de datos del experimento se ha beneficiado de manera crucial de los desarrollos y optimizaciones de los sistemas de gestión y procesamiento de datos desarrollados. Se ha dotado del entorno de análisis adecuado a los centros españoles.

Se ha participado, de forma muy relevante, en la integración de los diferentes elementos del sistema de computación de CMS a través de los ejercicios de computación ('Computing Challenges') que periódicamente se ejecutan para poner a prueba el sistema de computación a escalas y complejidades crecientes. Se han identificado los puntos débiles en los sistemas de gestión y procesamiento de datos, y han sido subsanados. Asimismo, se han identificado elementos del sistema de computación que ha sido necesario rediseñar (como los servicios de gestión de datos, el entorno de procesamiento de datos, el modelo de datos, etc.) En el marco de estos computing challenges se han desarrollado componentes esenciales para el modelo, como el sistema de transferencia de datos. Se ha comprobado que los centros españoles han incorporado de forma satisfactoria los recursos y servicios Grid necesarios, y que aumentan de escala adecuadamente.

Finalmente, se ha contribuido de forma muy significativa a las operaciones de computación de CMS, es decir, a la ejecución de los flujos de distribución y procesamiento de datos. Esta contribución se ha llevado a cabo mediante la aportación de recursos y a través de la ejecución de parte de las operaciones. El CIEMAT es el centro Tier-2 con mayor contribución en la producción de datos simulados en CMS (15 millones de los 150 producidos en los últimos 4 meses). El experimento ha alcanzado en 2007 una escala muy significativa en distribución de datos (0.5 PB/semana), en producción de datos Monte Carlo (50 M/mes), procesamiento y análisis de datos (con 9 millones de trabajos en el último año, lo que equivale a 25000/día, de los cuáles 160000 se ejecutaron en el CIEMAT y 130000 en el PIC). Los números alcanzados corresponden a una escala equivalente al 50 % de los objetivos para el 2008.

Este trabajo ha contribuido a la incorporación de los centros Tier-1 y Tier-2 españoles en el sistema de computación de CMS y al desarrollo, integración y operación de elementos claves de dicho sistema de computación, fundamentales para que el análisis de los datos pueda llevarse a cabo en la escala requerida y para que CMS pueda cumplir sus objetivos de física. Un ejemplo significativo que muestra la contribución del nuevo sistema de computación distribuido para alcanzar estos objetivos de física del experimento es el análisis del canal $H \rightarrow ZZ^{(*)} \rightarrow 4\mu$, donde todas las muestras necesarias para su consecución fueron producidas y distribuidas usando las herramientas Grid [126, 127, 128, 129]. A partir de este análisis todas las operaciones de producción, distribución y análisis de datos en CMS se ejecutan usando las componentes de este nuevo sistema de computación Grid desarrollado e implementado.

Apéndices

Apéndice A

Simulación Monte Carlo en los experimentos de Física de Altas Energías

A.1. La simulación Monte Carlo

La simulación de Monte Carlo es una técnica que combina conceptos estadísticos (muestreo aleatorio) con la generación de números seudo-aleatorios y la automatización de los cálculos. La simulación de Monte Carlo está presente en todos aquellos ámbitos en los que el comportamiento aleatorio o estocástico desempeña un papel fundamental. Representa una tentativa de modelar la naturaleza con la simulación directa de la dinámica del sistema estudiado. En este sentido, el método de Monte Carlo es una herramienta que nos sirve para obtener una solución de un sistema macroscópico con la simulación de sus interacciones microscópicas. La simulación de Monte Carlo es una técnica cuantitativa que hace uso de la estadística y de los ordenadores para imitar, mediante modelos matemáticos, el comportamiento aleatorio de sistemas reales.

La base de las simulaciones es la viabilidad para obtener largas secuencias de números aleatorios tales que la aparición de cada número en la secuencia sea impredecible, y que la secuencia de números supere tests estadísticos para detectar desviaciones de la aleatoriedad. Habitualmente, las secuencias de números se obtienen de algún algoritmo y se denominan números seudo-aleatorios, reflejando así su origen determinístico.

El primer número usado para empezar la secuencia es el llamado "número semilla". La longitud del ciclo, que es la cantidad de números aleatorios generados antes de que empiecen las repeticiones, es del orden de la centena. Estos algoritmos han de generar números de una manera realmente estocástica si se desean simular correctamente procesos como, por ejemplo, los de interacción entre las partículas y la materia. Esto hace que los generadores deban cumplir una serie de características:

- Buena distribución. Se entiende que los números obtenidos estén uniformemente distribuidos en el intervalo en el que se obtienen (0, 1). Si tomamos un subintervalo cualquiera, la fracción de números aleatorios que aparece respecto del total tiene que ser la misma para todo subintervalo de la misma amplitud.
- Ausencia de correlaciones. Al ser generados mediante un algoritmo, siempre tienen un ciclo más o menos largo. En el caso de simulaciones en que se usa una gran cantidad de números aleatorios sería importante que éstos no se repitieran para evitar las correlaciones.
- Repetibilidad. Interesa que se pueda reproducir la sucesión de números usados. Si se repite la simulación en las mismas condiciones el resultado ha de ser el mismo.

Una vez fijadas las características del experimento que queremos estudiar, la simulación detallada de un gran número de simulaciones proporciona, esencialmente, la misma información que un experimento real. Proporciona además información complementaria que experimentalmente sería, o bien imposible de obtener, o bien con un gran alto coste, tanto material como temporal. Obviamente, el análisis será tanto más preciso cuanto mayor sea el número de experimentos (historias) simulados.

A.2. Simulación Monte Carlo en los experimentos de Física de Altas Energías

Todos los procesos que involucran colisiones o transporte de partículas tienen naturaleza estocástica. No se puede prever qué tipo de interacción se va a producir en cada momento y lugar sino que solamente se puede asignar una probabilidad a cada uno de los posibles sucesos. Sin embargo, las distribuciones de probabilidad que gobiernan los procesos que queremos estudiar son bien conocidas. El método de Monte Carlo hace uso de las distribuciones de probabilidad de las interacciones individuales en los materiales para simular la trayectoria errática de las partículas. Cuando una partícula con carga, un fotón o un neutrón, se hace incidir sobre la materia se producen una serie de interacciones con los átomos y núcleos que la forman. Todos estos fenómenos de absorción, dispersión y producción de partículas secundarias siguen un proceso aleatorio. Todos los datos físicos que van a determinar estos procesos físicos estarán implementados en el código de modo que, mediante secuencias de números aleatorios, se puede simular lo que realmente ocurre en la naturaleza.

El desarrollo y aplicación del método de Monte Carlo al estudio de la radiación ionizante en la materia se debió a tres razones fundamentalmente. En primer lugar, el desarrollo de la teoría cuántica permitió conocer exhaustivamente las distintas secciones eficaces de interacción de las partículas en los diversos materiales. En segundo lugar, el problema de la dispersión de la radiación no es fácilmente tratable si no se usan métodos estadísticos, debido al gran número de interacciones que se producen. Por último, el uso de ordenadores cada vez más rápidos y potentes ha supuesto un gran avance en este campo. Los números aleatorios usados se obtienen de una distribución de probabilidad que describe el comportamiento de la partícula. Al realizar un gran número de historias al azar aumentará la precisión del valor promedio (o de otras cantidades de interés).

En contraposición con los métodos analíticos, las simulaciones de Monte Carlo pueden usar secciones eficaces reales, modelos reales de haces, y descripciones de los detectores con geometrías complejas. El precio que se debe pagar al aumentar la complejidad es el aumento de los tiempos de cálculo.

La cadena de producción de datos simulados en los experimentos de Física de Altas Energías comprende diversas etapas, donde el output de cada una de ellas constituye el input de la siguiente:

- Durante la generación se simulan las colisiones, determinándose las partículas obtenidas así como las propiedades cinemáticas de los sucesos.
- En la simulación se calculan las interacciones de estas partículas creadas con el detector.
- La simulación de las señales emitidas por la electrónica del detector se conoce con el nombre de digitalización.
- Finalmente, la reconstrucción es la operación por la cual se calculan cantidades físicas de las partículas, como la energía y el momento.

Cada una de estas etapas se realiza por separado por diversas razones. Los paquetes de software necesarios son diferentes para cada paso. El tiempo necesario para procesar cada suceso difiere drásticamente entre etapas. La generación es un proceso considerablemente más rápido que la simulación, mientras que la reconstrucción y la digitalización tienen unos requisitos intermedios. Este desacoplamiento entre etapas permite, por ejemplo, repetir la reconstrucción para tener en cuenta mejoras en el software, nuevas calibraciones o diferentes algoritmos de reconstrucción, sin necesidad de rehacer la simulación.

A.2.1. Generación

El objetivo de la etapa de generación es la simulación de las colisiones, la determinación de las partículas que se obtienen como resultados de dichas colisiones, y las propiedades cinemáticas de los sucesos, de acuerdo a las distribuciones determinadas por la teoría (QED, QCD, etc.) Se han desarrollado varios paquetes de software para este propósito, como PYTHIA [130] o HERWIG [131]. Estos generadores proporcionan los sucesos de las colisiones como input para la simulación del detector, teniendo en cuenta los parámetros particulares para cada estudio determinado. Los sucesos pueden ser aceptados o rechazados de forma opcional en función de cierta información sobre los sucesos del generador para así enriquecer el contenido físico de los sucesos.

A.2.2. Simulación

A las partículas generadas en las colisiones simuladas se les sigue la pista a través del detector completo durante la fase de simulación. Se utiliza un software dedicado, que suele estar basado en el paquete GEANT4 [132]. Este software, desarollado en el CERN, proporciona un gran conjunto de procesos físicos que permiten la descripción detallada de las interacciones electromagnéticas y hadrónicas, y puede calcular todos los procesos de interacción de las partículas con la materia (tales como scattering Compton, procesos de bremstrahlung, múltiple scattering o procesos de producción de pares). Las herramientas de simulación deben incluir también una descripción detallada de la geometría de todas las componentes del detector así como de la forma de los campos electromagnéticos que lo rodea. A las señales producidas por el detector debido al paso de las partículas se las conoce con el nombre de *hits*.

Para conseguir una mayor precisión hay que tener en cuenta la gran cantidad de colisiones pp inelásticas y difractivas que se producirán en cada cruce de haces, y que también inducirán señal en el detector. Hay que considerar también las partículas procedentes de cruces anteriores y posteriores al actual, y que se mezclarán con el suceso nominal (el que dispara el trigger) dependiendo de la velocidad de la electrónica de lectura de los distintos componentes del detector. Todos estos fenómenos reciben el nombre común de *pile-up* (PU), y requieren un tratamiento especial. Durante los regímenes de baja luminosidad ($\mathfrak{L} = 2 \text{ x } 10^{33}$ cm^{-2}s^{-1}) y alta luminosidad ($\mathfrak{L} = 10^{34}$ cm^{-2}s^{-1}), en LHC se producirán, respectivamente, un promedio de 5 y 25 colisiones inelásticas o difractiva en cada cruce de haces. La figura A.1 muestra un suceso del tipo H$\rightarrow$ZZ$\rightarrow 4\mu$ (arriba) y cómo queda oculta entre las miles de trazas procedentes de todas las interacciones y procesos de PU (abajo).

Puesto que la adición de las colisiones de PU ocurre de forma mucho más rápida que la simulación del detector, y dado que depende de la luminosidad del LHC y de las condiciones de operación, los sucesos de PU suelen simularse por separado. Se mezclan después ambos outputs en una segunda etapa, usando la contribución del PU adecuada dependiendo del régimen de luminosidad que se desee simular. Las colisiones de señal se usan, por tanto, para producir muestras correspondientes a diferentes luminosidades.

La simulación de los sucesos de PU es totalmente equivalente a la simulación de la señal, se usan las mismas herramientas software y se obtienen hits con el mismo formato. Las colisiones PU que se van a mezclar con la señal se escogen de forma aleatoria a partir de una muestra generada con anterioridad. Para evitar posibles correlaciones entre subconjuntos de sucesos seleccionados que parcialmente pudiesen coincidir en la misma secuencia de sucesos de PU, las muestras de PU no se usan nunca dos veces en el mismo orden. Más aún, las colisiones PU simuladas que satisfacen las condiciones del sistema de trigger son filtradas para evitar un *bias* de baja estadística entre los muchos cruces de haces que podrían usar tales eventos.

El número promedio de sucesos, los números mínimo y máximo de cruces de haces antes y después del nominal, y el espacio entre cruces, son parámetros configurables. Más aún, es posible escoger diferentes opciones para seleccionar de forma aleatoria sucesos de PU a partir de la muestra pregenerada.

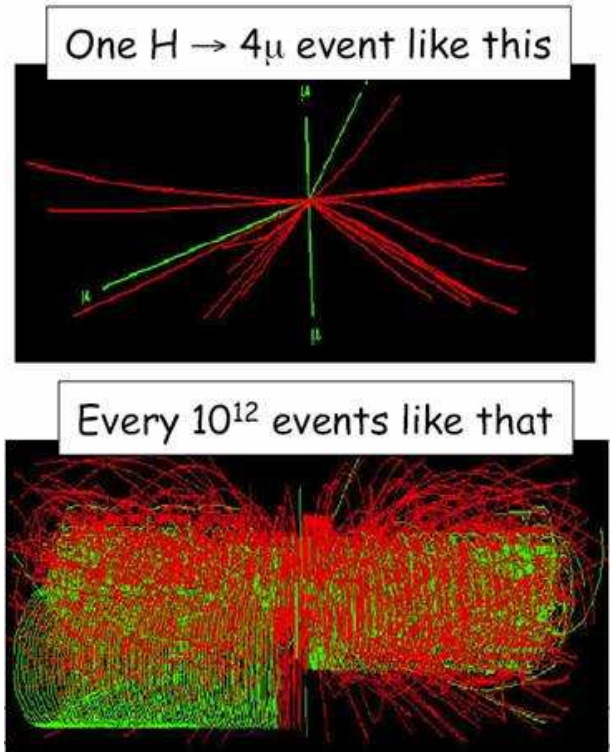


Figura A.1: Simulación de un suceso típico del LHC (arriba), y todas las trazas de partículas que interaccionan con el detector ocultando este suceso (abajo).

A.2.3. Digitalización

La etapa de digitalización, que sucede a la fase de creación de hits, constituye la simulación de la electrónica de lectura usada por el detector para la toma de datos. A partir de las posiciones de los hits y de las pérdidas de energía simuladas en el detector sensible produce un output que ha de ser lo más parecido posible a los datos reales procedentes del detector. La información procedente de la etapa de generación (como el tipo de partícula o su momento) se preserva durante la digitalización. Estas señales de la electrónica que simulan en esta fase se conocen con el nombre de *digis*.

A.2.4. Reconstrucción

Finalmente, el proceso termina con la fase de reconstrucción. Puede dividirse en tres etapas, correspondientes a la reconstrucción local en los módulos individuales de cada subdetector, la reconstrucción total de los subdetectores, la reconstrucción global en el detector completo y y la combinación de los objetos reconstruidos para producir objetos de alto nivel.

El input de la reconstrucción local son, o bien los datos reales procedentes del detector, o bien los datos simulados jugando el papel de datos reales (en cualquier caso reciben el nombre de digis).

Los algoritmos de reconstrucción global usan los objetos creados en la reconstrucción local dentro de un módulo individual del detector, combinándolos con los procedentes de los demás módulos del mismo subdetector, para producir los objetos que representan la mejor medida de ese subdetector. La información proveniente de varios subdetectores no se combina durante esta etapa.

Finalmente, la última etapa de la reconstrucción combina objetos creados durante la reconstrucción global

en cada subdetector, creando objetos de alto nivel basados en el detector completo, y que son apropiados para disparar el trigger de alto nivel y para los análisis físicos.

Los grupos de física solicitan la simulación de sucesos para un determinado proceso físico, especificando un conjunto bien definido de parámetros, versión del software, geometría, y muestra de PU.

Bibliografía

[1] S. L. Glashow, Nucl. Phys. B22 (1961) 579.
S. Weinberg, Phys. Rev. Lett. 19 (1967) 1264.
A. Salam in *Elementary Particle Theory,* ed N. Svartholm
(Almquist and Wiksells, Stockholm, 1969) p. 367.

[2] S. L. Glashow, I. Iliopoulos, L. Maiani, Phys. Rev. D2 (1970) 1285.

[3] N. Cabbibo, Phys. Rev. Lett. (1963) 531.
M. Kobayashi, K. Maskawa, Prog. Theor. Phys. 49 (1973) 652.

[4] D. I. Kazakov. Beyond the Standard Model. `http://arxiv.org/abs/hep-ph/0611279`, Noviembre 2006. Charla plenaria en XXXIII ICHEP, Moscú.

[5] J. Ellis. Beyond the Standard Model for Hillwalkers. *CERN-TH/98-329*, 1998. Charlas presentadas en la European School of High-Energy Physics, St. Andrews, Scotland, UK.

[6] J. Ellis. Limits of the Standard Model. *CERN-TH/2002-320*, Agosto 2002. Charlas presentadas en la PSI Summer School, Zuoz, Suiza.

[7] J. Ellis. Supersymmetry for Alp Hikers. *CERN-TH/2002-052*, Agosto-Septiembre 2002. Charlas presentadas en la Europena School of High-Energy Physcis, Beatenberg, Suiza.

[8] P. W. Higgs, Phys. Lett. 12 (1964) 132.
P. W. Higgs, Phys. Rev. Lett. 13 (1964) 508.
P. W. Higgs, Phys. Rev. 145 (1966) 1156.
F. Englert and R. Brout, Phys. Rev. Lett. 13 (1964) 312.
G. S. Guralnik, C. R. Hagen and T. W. B. Kibble, Phys. Rev. Lett. 13 (1964) 585.

[9] LHC Home Page. `http://lhc-homepage.web.cern.ch/lhc-new-homepage`.

[10] CERN – The world's lagest particle physics laboratory. `http://www.cern.ch`.

[11] LEP design report, Vol. 1. The LEP injector chain. CERN-LEP-84-01.

[12] LHCb Collaboration. A Large Hadron Collider Beauty Experiment for Precision Measurements of CP Violation and Rare Decays. *CERN/LHCC 98-4*, Febrero 1998.

[13] ALICE Collaboration. ALICE - Technical Proposal for a Large Ion Collider Experiment at the CERN LHC. *CERN/LHCC 95-71*, Diciembre 1995.

[14] CMS Collaboration. The Compact Muon Solenoid Technical Proposal. *CERN/LHCC 94-38*, Diciembre 1994.

[15] ATLAS Collaboration. ATLAS Technical Proposal. *CERN/LHCC 94-43*, Diciembre 1994.

[16] TOTEM Collaboration. TOTEM: Total Cross Section, Elastic Scattering and Diffraction Dissociation at the LHC at CERN. Technical Design Report. *CERN/LHCC 2004-002*, Enero 2004.

[17] CMS collaboration. CMS Technical Design Report: Detector performance and software. *CERN-LHCC-2006-001*, Febrero 2006.

[18] CMS Collaboration. CMS Tracker Project Technical Design Report. *CERN/LHCC 98-6*, Abril 1998.

[19] CMS Collaboration. CMS ECAL Technical Design Report. *CERN/LHCC 97-33*, Diciembre 1997.

[20] CMS Collaboration. CMS HCAL Technical Design Report. *CERN/LHCC 97-31*, Junio 1997.

[21] CMS Collaboration. CMS Muon Technical Design Report. *CERN/LHCC 97-32*, Diciembre 1997.

[22] CMS Collaboration. CMS Magnet Project - Technical Design Report. *CERN/LHCC 97-10*, Mayo 1997.

[23] CMS Collaboration. The TriDAS Project: Technical Design Report, Volume 1: The Level-1 Trigger. *CERN/LHCC 2000-038*, Diciembre 2000.

[24] CMS Collaboration. The TriDAS Project: Technical Design Report, Volume 2: Data Adquisition & High Level Trigger. *CERN/LHCC 2002-026*, Diciembre 2002.

[25] IEEE 802.3z-1998, Julio de 1998. IEEE Standards Board meeting.

[26] Ian Foster, Carl Kesselman, and Steven Tuecke. The anatomy of the Grid: Enabling scalable virtual organizations. *Lecture Notes in Computer Science*, 2150, 2001.

[27] Standard Performance Evaluation Corporation.
`http://www.spec.org`.

[28] LHC Computing Grid.
`http://cern.ch/lcg/`.

[29] gLite, Lightweight Middleware for Grid Computing.
`http://cern.ch/glite/`.

[30] EGEE, Enabling Grids for E-sciencE.
`http://eu-egee.org/`.

[31] The Data Grid Project.
`http://eu-datagrid.web.cern.ch/eu-datagrid/`.

[32] The Globus Alliance.
`http://www.globus.org/`.

[33] James Frey, Todd Tannenbaum, Ian Foster, Miron Livny, and Steven Tuecke. Condor-g: A computation management agent for multi-institutional grids. *Proceedings of the Tenth IEEE Symposium on High Performance Distributed Computing (HPDC10) San Francisco, California*, Agosto 2001.
`http://www.cs.wisc.edu/condor/condorg/`.

[34] Housley et. al. Rfc 3280 - internet x.509 public key infrastructure certificate and certificate revocation list (crl) profile. *The Internet Society*, Abril 2002.

[35] Fabrizio Pacini. Job Description Language Attributes Specification for the gLite middleware (submission through Network Server). EGEE-JRA1-TEC-555796-JDL-. CERN Engineering Data Management Service. `https://edms.cern.ch/document/555796/1/`.

[36] Min Cai, Ann Chervenak, and Martin Frank. A Peer-to-Peer Replica Location Service Based on A Distributed Hash Table. In *SC2004 Conference*, Noviembre 2004. Pittsburgh, PA.

[37] Grid File Access Library.
`http://grid-deployment.web.cern.ch/grid-deployment/documentation/LFC_DPM/gfal/html/`.

[38] S. Shepler. Rfcc - 2624. nfs version 4 design considerations. *Sun Microsystems, Inc.*, Junio 1999.

[39] The Andrew File System.
`http://www.psc.edu/general/filesys/afs/afs.html`.

[40] Platform Computing.
`http://www.platform.com/`.

[41] Perfectly Normal File System.
`http://www-pnfs.desy.de`.

[42] The GLUE Schema.
`http://glueschema.forge.cnaf.infn.it`.

[43] LDAP Services User Guide.
`http://hepunx.rl.ac.uk/edg/wp3/documentation/wp3-ldap_user_guide.html`.

[44] `http://www.r-gma.org/index.html`.

[45] The Open Grid Forum.
`http://www.ogf.org/`.

[46] GridICE: a monitoring service for the Grid.
`http://gridice.forge.cnaf.infn.it`.

[47] Service Availability Monitoring.
`http://wiki.grid.cyfronet.pl/SAM`.

[48] Grid Operation Centre DataBase.
`http://goc.grid-support.ac.uk/gridsite/gocdb/`.

[49] GridView: Monitoring and Visualization Tool for LCG.
`http://gridview.cern.ch`.

[50] `http://arda-dashboard.cern.ch/cms/`.

[51] Monitoring Agents Using a Large Integrated Service Architecture, `http://monalisa.cern.ch/monalisa.html/`
I. Legrand et al., MonALISA : A Distributed Service for Monitoring, Control and Global Optimization. Proceedings of the International Conference of Computing in High Energy and Nuclear Physics, Febrero 2006, Mumbai, India.

[52] CMS collaboration. CMS Computing Project: Technical design report. *CERN-LHCC-2005-023*, Junio 2005.

[53] Physics Experiment Data Export, `http://cern.ch/cms-project-phedex/`.
L. Tuura et al., PhEDEx high-throughput data transfer management system. Proceedings of the International Conference of Computing in High Energy and Nuclear Physics, Febrero 2006, Mumbai, India.

[54] CMS Dataset Bookkeeping System.
`https://twiki.cern.ch/twiki/bin/view/CMS/DBS-TDR`.

[55] CMS Data Location Service.
`https://twiki.cern.ch/twiki/bin/view/CMS/DLS`.

[56] CMS Trivial File Catalogue.
`https://twiki.cern.ch/twiki/bin/view/CMS/SWIntTrivial`.

[57] `http://lynx.fnal.gov/ntier-wiki`.

[58] http://www.squid-cache.org.

[59] C.Grandi and A. Renzi. Object Based System for Batch Jobs submission and Monitoring. CMS NOTE-2003/005. http://cmsdoc.cern.ch/documents/03/note03_005.pdf
C. Grandi et al. Evolution of BOSS, a tool for job submission and tracking. Proceedings of the International Conference of Computing in High Energy and Nuclear Physics, Febrero 2006, Mumbai, India.

[60] http://t2.unl.edu:8084/xml/.

[61] Puerto de Información Científica.
http://www.ifae.es/pic/.

[62] Centro de Investigaciones Energéticas, Medio Ambientales y Energéticas.
http://www.ciemat.es/.

[63] Instituto de Física de Cantabria.
http://www.ifca.unican.es/.

[64] Universidad Autónoma de Barcelona.
http://www.uab.es/.

[65] Instituto de Física de Altas Energías.
http://www.ifae.es/.

[66] Experimento Magic.
https://www.ipp.phys.ethz.ch/research/?file=experiments_magic.

[67] Long-baseline Neutrino Oscillation Experiment.
http://neutrino.kek.jp/.

[68] Intel Corporation.
http://www.intel.com.

[69] DELL.
http://www.dell.com.

[70] Sun Fire X4500 Server.
http://www.sun.com/servers/x64/x4500/.

[71] SUN StorageTek.
http://www.sun.com/storagetek.

[72] Linear Grabar-Abrirte technology.
http://www.lto.org/.

[73] http://www.clusterresources.com/pages/products/torque-resource-manager.php
http://www.clusterresources.com/pages/products/maui-cluster-scheduler.php.

[74] Enstore, the Fermilab Mass Storage System.
http://www-isd.fnal.gov/enstore/.

[75] Kickstart Installations.
http://www.redhat.com/docs/manuals/linux/RHL-7.3-Manual/custom-guide/ch-kickstart2.html.

[76] Red Hat Linux.
http://www.redhat.com.

[77] Robert Harakaly. EGEE II INFSO-RI-031688. 2006, Ginebra, Suiza.

[78] Ganglia Monitoring System.
`http://ganglia.sourceforge.net/`.

[79] Nagios.
`http://www.nagios.org/`.

[80] Roundup Issue Tracker.
`http://roundup.sourceforge.net/`.

[81] Advanced Micro Devices, AMD.
`http://www.amd.com`.

[82] IBM z/VM publications. http://www.vm.ibm.com/pubs/.

[83] The Hyper-Threading Technology.
`http://www.intel.com/technology/platform-technology/hyper-threading/index.htm`.

[84] SUPERMICRO Computer, Inc.
`http://www.supermicro.com`.

[85] Applied Micro Circuits Corporation.
`http://www.3ware.com`.

[86] Sun Grid Engine.
`http://www.sun.com/software/gridware/`.

[87] Jim Basney, Miron Livny, and Todd Tannenbaum. High throughput computing with condor. *HPCU new, Volume 1(2)*, Junio 1997.
`http://www.cs.wisc.edu/condor/`.

[88] Yum.
`http://fedoraproject.org/wiki/Tools/yum`.

[89] gstat.
`http://goc.grid.sinica.edu.tw/gstat/`.

[90] J. Caballero, The CIEMAT Batch Queue Monitor.
`http://pcae30.ciemat.es/pbs_monitor.html`.

[91] LCG Optical Private Network.
`http://lcg.web.cern.ch/lcg/activities/networking/nw-grp.html`.

[92] LHCNet: Transatlantic Networking for the LHC and the U.S. HEP Community.
`http://lhcnet.caltech.edu`.

[93] Red GEANT.
`http://www.geant.net`.

[94] RedIris.
`http://www.rediris.es`.

[95] Anella Cientifica.
`http://www.cesca.es/comunicacions/anella.html`.

[96] P. Garcia-Abia et al. Implementation of Monte Carlo Production in LCG-2. *CMS NOTE-2005/019*, Octubre 2005.

[97] CMS Monte Carlo Production in the Open Science and LHC Computing Grid, J.Caballero, P.Garcia-Abia and J.M.Hernandez, Proceedings of the International Conference of Computing in High Energy and Nuclear Physics, Febrero 2006, Mumbai, India.

[98] J.Caballero, P.Garcia-Abia, and J.M.Hernandez. Integration and operational experience in CMS Monte Carlo production in LCG. *CMS NOTE-2007/016*, Julio 2007.

[99] T.Barras et al., Techniques for High-Throughput, Reliable Transfer Systems: Break-Down of PhEDEx Design. Proceedings of the International Conference of Computing in High Energy and Nuclear Physics, Febrero 2006, Mumbai, India.

[100] Runjob Project.
`http://projects.fnal.gv/runjob/`.

[101] Véronique Lefébure and Julia Andreeva. RefDB: The Reference Database for CMS Monte Carlo Production. *Proceedings of the CHEP03, La Jolla, California*, Marzo 2003.

[102] ORCA, Object-oriented Reconstruccion for CMS Analysis.
`http://cmsdoc.cern.ch/orca/`.

[103] OSCAR, Object oriented Simulation for CMS Analysis and Reconstruction.
`http://cmsdoc.cern.ch/oscar/`.

[104] COBRA, Coherent Object-oriented Base for Reconstruccion, Analysis and Simulation.
`http://cobra.web.cern.ch/cobra/`.

[105] CMS Software.
`http://cmsdoc.cern.ch/cms/cpt/Software/html/General/`.

[106] POOL, Pool of Persistent Objects for LHC.
`http://pool.cern.ch/`.

[107] M.A.Afaq, CMS Publish Service CMSGLIDE.
`http://lynx.fnal.gov/runjob/Setup_20Publish_20Service`.

[108] LFC, LCG File Catalogue.
`https://uimon.cern.ch/twiki/bin/view/LCG/LfcAdminGuide`.

[109] M.Abbrescia, W. Bacchi, J.Caballero et al. CMS Monte Carlo production in the WLCG Computing Grid. Proceedings of the International Conference of Computing in High Energy and Nuclear Physics, Septiembre 2007, Victoria, Canadá.

[110] M.Abbrescia, W. Bacchi, J.Caballero et al. CMS Monte Carlo production operations in a distributed computing environment. Poster presented at the Hadron Collider Physics Symposium (HCP), Mayo 2007, La Biodola, Isola d'Elba, Italia.

[111] M.Aldaya, J.Caballero et al. Producción y distribución de datos para el experimento CMS. Proceedings de la XXX reunión bienal de la Real Sociedad Española de Física, Septiembre 2005, Orense.

[112] Deutsches Elektronen-Synchrotron, DESY.
`http://www.desy.de`.

[113] MySQL AB.
`http://www.mysql.com`.

[114] J. Caballero and J.M. Hernandez, A Lightweight Production Monitoring.
`https://twiki.cern.ch/twiki/pub/CMS/LCGProdOperationsSteps/PAmonBOSSDB-HowTo.pdf`.

[115] D. D. Corkill. Collaborating software: Blackboard and multi-agent systems and the future. *in proceedings of the International Lisp Conference*, New York, 2003.

[116] Oracle.
`http://www.oracle.com`.

[117] V. Cerf and R. Kahn. A protocol for packet network intercommunication. *IEEE Transactions on Communications, Vol. COM-22, No. 5, pp 637-648*, Mayo 1974.

[118] Cormen, Leiserson, Rivest, and Stein. Introduction to algorithms. *MIT Press and McGraw-Hill, Second Edition*, 2001.

[119] P.Garcia-Abia et al. CMS Grid Computing at the Spanish Tier-1 and Tier-2 sites. Poster presented at the International Conference of Computing in High Energy and Nuclear Physics, Febrero 2006, Mumbai, India.

[120] J.Caballero et al. Exercising CMS dataflows and workflows in computing challenges at the Spanish Tier-1 and Tier-2 sites. Poster presented at the International Conference of Computing in High Energy and Nuclear Physics, Septiembre 2007, Victoria, Canadá.

[121] J.M. Hernández et al. CMS DC04 Data Challenge at PIC Tier-1 and CIEMAT Tier-2. *CMS NOTE-2004/030*, 2004.

[122] J.M. Hernández et al. LCG Service Challenge 3 at the Spanish Tier-1 and Tier-2 sites. *CMS NOTE-2006/087*, 2006.

[123] J.Alcaraz, J.Caballero et al. CMS CSA06 Computing, Software and Analysis challenge at the Spanish Tier-1 and Tier-2 sites. *CMS NOTE-2007/022*, Septiembre 2007.

[124] CMS Collaboration. CMS Computing, Software and Analysis Challenge in 2006. *CERN/LHCC 2007-010*, Marzo 2007.

[125] An Object-oriented Data Analysis Framework.
`http://root.cern.ch`.

[126] M.Aldaya, P.Arce, J.Caballero et al. Search for the Standard Model Higgs boson in the H→ZZ$^{(*)}$ →4μ decay channel using a mass-independent analysis. *CMS NOTE-2006/106*, Junio 2006.

[127] CMS collaboration. CMS Technical Design Report: Physics performance. *CERN-LHCC-2006-021 and Journal of Physics. G: Nuclear and Particle Physics. 34 995-1579*, Junio 2007.

[128] M.Aldaya, P.Arce, J.Caballero et al. Búsqueda del bosón de Higgs en el canal H→ZZ$^{(*)}$ →4μ en el experimento CMS. Proceedings de la XXX reunión bienal de la Real Sociedad Española de Física, Septiembre 2005, Orense.

[129] S. Abdullin et al. CMS Detector Sensitivity to the Standard Model Higgs Boson in $H \rightarrow ZZ \rightarrow 4\mu$ Decay Channel Proceedings of the American Physical Society April Meeting, Abril 2006. Dallas, USA.

[130] PYTHIA.
`http://www.thep.lu.se/~torbjorn/Pythia.html`.

[131] HERWIG.
`http://hepwww.rl.ac.uk/theory/seymour/herwig/`.

[132] GEANT.
`http://geant4.web.cern.ch`.

Acrónimos

A continuación se listan algunos de los acrónimos de carácter técnico más importantes usados en el texto.

AOD	Analysis Object Data
BDII	Berkeley Database Information Index
BOSS	Batch Object Submission System
CA	Certification Authority
CAF	CERN Analysis Facility
CASTOR	CERN Advanced STORage manager
CE	Computing Element
CRAB	CMS Remote Analysis Builder
CSA	Computing, Software and Analysis challenge
DBS	Data Bookkeeping System
DC	Data Challenge
DCAP	D-Cache Access Protocol
DIT	Directory Information Tree
DLS	Data Location Service
DMS	Data Management System
DN	Distinguished Name
DPM	Disk Pool Manager
EDG	European DataGrid
EDM	Event Data Model
EGEE	Enabling Grids for E-sciencE
EVD	EVent Data
FEVT	Full EVenT
FTS	File Transfer Service
GFAL	Grid File Acccess Library
GG	Grid Gate
GGF	Global Grid Forum
GIIS	Grid Index Information Server
GMA	Grid Monitoring Architecture
GOCDB	Grid Operations Centre DataBase
GRIS	Grid Resource Information Server
GUID	Grid Unique Identifier
HSM	Hierarchical Storage Managers
IS	Information System
JDL	Job Description Language

LAN	Local Area Network
LCG	LHC Computing Grid
LDAP	Lightweigh Directory Access Protocol
LFC	LCG File Catalogue
LFN	Logical File Name
LRC	Location Replica Catalogue
LRMS	Local Resource Management System
LSF	Load Sharing Facility
MC	Monte Carlo
MDS	Monitoring and Discovery Service
MSS	Massive Storage System
NFS	Network File System
NS	Name Server
OPN	Optical Private Network
OSG	Open Science Grid
PA	ProdAgent
PBS	Portable Batch System
PFN	Physical File Name
PNFS	Perfectly Normal File System
POOL	Pool of Persistent Objects for LHC
POSIX	Portable Operating System Interface
PSU	Pool Selection Unit
PU	Pile Up
RB	Resource Broker
RFIO	Remote File Input/Output protocol
R-GMA	Relational Grid Monitoring Architecture
RLS	Replica Location Service
RMC	Replica Metadata Catalogue
SAM	Service Availability Monitoring
SC	Service Challenge
SE	Storage Element
SRM	Storage Resource Manager
SURL	Storage URL
TCP	Transmission Control Protocol
TFC	Trivial File Catalogue
TMDB	Transfer Management Database and Bookkeeping
TURL	Transport URL
UI	User Interface
URI	Uniform Resource Identifier
URL	Uniform Resource Locator
VDQM	Volume Drive Queue Manager
VM	Virtual Machine
VO	Virtual Organization
VOMS	Virtual Organization Membership Service
WAN	Wide Area Network
WLCG	Worldwide LCG
WM	Workload Manager
WMS	Workload Management System
WN	Worker Node

Índice de tablas

Índice de figuras