ALICE Grid-Aktivitäten

K. Schwarz

DVEE, Gesellschaft für Schwerionenforschung mbH
D-64291 Darmstadt

Zusammenfassung: ALICE ist eines der vier Experimente des Large Hadron Colliders LHC, die derzeit am CERN in Genf gebaut werden. Wenn das Experiment seinen Betrieb aufnimmt, wird es Daten mit einer Rate von bis zu 2 Petabyte pro Jahr aufzeichnen. Die hieraus resultierenden Anforderungen an das LHC Computing Modell führten zur Ausbildung mehrerer Grid-Projekte, u.a. dem gemeinsamen Grid-Projekt aller LHC-Experimente, LCG, sowie der ALICE Produktionsumgebung AliEn. Diese Grid-Implementation ermöglicht es ALICE in einer global verteilten Computing-Infrastruktur Daten aus dem Experiment zu simulieren, rekonstruieren und zu analysieren. In sogenannten „Data Challenges“ wird getestet, ob das vorhandene Framework mit den anfallenden Datenmengen zurechtkommt und somit validiert werden kann. Im Rahmen des „Physics Data Challenge 2004“ sollen die global erzeugten Daten schließlich interaktiv und parallel mit der Parallel ROOT Facility PROOF analysiert werden.

1 Überblick

1.1 Das ALICE Experiment am LHC

ALICE (A Large Ion Collider Experiment [ALICE]) ist eines der vier Experimente des Large Hadron Colliders (LHC [LHC]) mit einem Umfang von 27 km, der derzeit am CERN [CERN] in Genf gebaut wird. Mit Hilfe von Proton-Proton-Kollisionen mit 7 TeV/Nukleon sowie Schwerionenkollisionen mit einer Schwerpunktenergie von 5,5 TeV/Nukleon soll Physik im Hochenergiebereich betrieben werden. Mit dem Universalschwerionenexperiment ALICE lassen sich fast alle bekannten Observablen der Teilchenkollisionen messen. Insbesondere kann mit ALICE aber das sogenannte Quark-Gluon-Plasma (QGP) untersucht werden, einem Materiezustand, in dem sich das Universum ca. 10^{-5} Sekunden nach dem Urknall befunden hat (Abbildung 1).

Zu diesem Zeitpunkt hatte das Universum eine Temperatur von 10 Milliarden Grad und eine Ausdehnung von lediglich einem Kilometer. Erst nach weiterer Abkühlung fanden sich Quarks und Gluonen zu hadronischer Materie zusammen und bildeten aus der heute bekannten Kernmaterie komplexere Gebilde wie höhere Elemente, Sterne und schließlich Galaxien. In den Schwerionenstößen des LHC wird die Kernmaterie nun so stark erhitzt und verdichtet, dass im Kollisionszentrum das QGP im Labor nachgebildet werden kann. Die Lebensdauer dieses künstlich erzeugten Zustandes des frühen Universums sollte bei LHC-Energien lang genug sein, dass dieser mit Hilfe der dem ALICE-Detektor zugänglichen Observablen (wie z.B. Hadronen, Elektronen, Myonen und Photonen) studiert werden kann. Insbesondere die Übergangsphase vom QGP in den hadronischen Zustand ist physikalisch interessant.

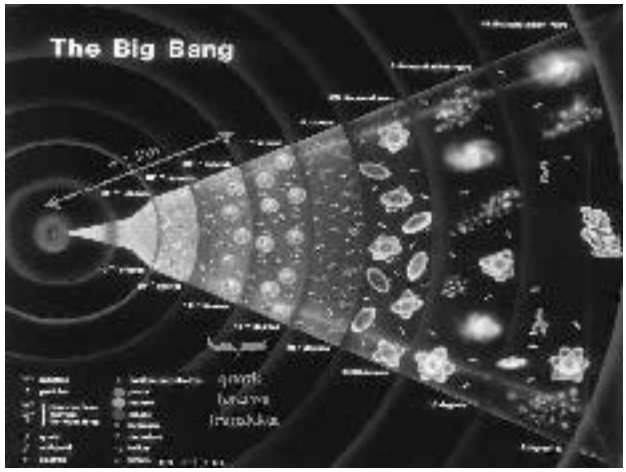

Abbildung 1: Etwa 10^{-5}s nach dem Urknall befand sich das Universum im Zustand des QGPs. Die Konstituenten der heutigen Kernmaterie, die Quarks, hatten sich noch nicht zu z.B. Protonen und Neutronen zusammengefunden.

Der ALICE-Detektor [ALICE-TDR] selbst ist ein komplexes Gebilde mit einer Gesamtlänge von 25 Metern und einem Gewicht von 16 Tonnen (Abbildung 2). Er besteht aus mehreren Einzeldetektoren, die in verschiedenen Teilen der Welt hergestellt werden. Erst am CERN entsteht hieraus das entgültige Experiment, wobei natürlich Millimeterarbeit geleistet werden muss.

Abbildung 2: Der ALICE-Detektor am Point 2 des LHC-Beschleunigers, 90m unter der Erdoberfläche. Er besteht aus mehreren Einzeldetektoren, die alle ihre spezielle Aufgabe haben. Insgesamt ist der Detektor 25 m lang und 16 t schwer.

1.2 Computing-Herausforderungen

Die vom Experiment kommenden Rohdaten müssen unter Berücksichtigung der Wechselwirkung mit dem Detektormaterial in einen Zustand überführt werden, auf den dann in der nachfolgenden Analyse physikalische Modelle und Prozesse angewendet werden können, um die zugrunde liegende Physik zu extrahieren bzw. zu verstehen. Die der Analyse vorausgehenden Schritte nennt man Rekonstruktion der Daten. Hierzu werden wichtige Schritte, wie Detektoreichung, Mustererkennung, Erkennung von Teilchenspuren und Teilchenidentifikation gezählt. Bei der Simulation wird der umgekehrte Weg beschritten. Ausgehend von bekannten Physikprozessen werden Rohdaten konstruiert, wie sie aus dem Detektor kommen würden.

Im ALICE-Experiment fallen pro Sekunde 160 GB an Daten an. Mit der Hilfe einer Serie von Triggern wird aus der Fülle der ankommenden Daten das für Physiker Interessante ausgewählt und auf eine Rate von 1,25 GB/s reduziert (Abbildung 3). Damit wird es 20 Jahre lang bis zu 2 PetaByte pro Jahr aufzeichnen. Diese Datenmengen wollen zudem analysiert werden. Noch bevor der LHC fertig aufgebaut ist, werden Physiker schon in großem Stil Teilchenkollisionen simulieren, um ein korrektes Design des Detektors gewährleisten zu können. All dies stellt derartige Anforderungen an das Computing-Modell von ALICE, dass diese an einem einzelnen Zentrum (z.B. CERN) kaum noch erfüllt werden können.

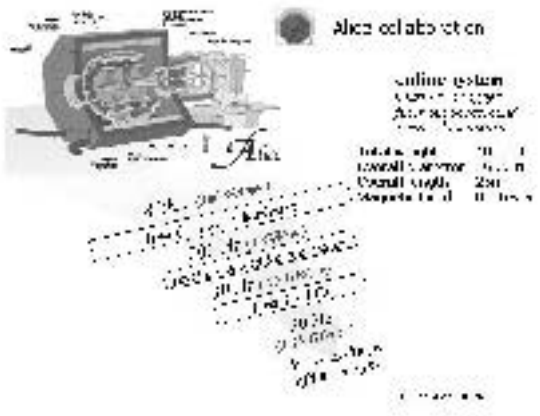

Abbildung 3: Die beim ALICE-Experiment ursprünglich anfallende Datenmenge von 160 GB/s wird durch eine Reihe von Triggern, mit denen die für Physiker interessanten Daten ausgewählt werden, auf 1,25 GB/s reduziert.

Weil ALICE selbst schon eine Kollaboration ist, die aus ungefähr 1000 Wissenschaftlern in der ganzen Welt besteht, liegt der Gedanke nahe, auch die Computing-Infrastruktur über den Globus zu verteilen. Als eine allgemein hierfür anerkannte Lösung bietet sich das Grid an. Ein Grid erlaubt es Forschungs- und/oder Computerzentren, die sich zu einer sogenannten „virtuellen Organisation“(z.B. Alice) zusammen geschlossen haben, sich als

eine Einheit zu sehen. Sie bilden auf diese Weise sozusagen ein großes virtuelles Computerzentrum. Jeder erreichbare Rechenknoten vermag einlaufende Programme auszuführen und die Benutzer können verteilte Datensätze als logische Dateien ohne Kenntnis des Speicherorts ansprechen. Natürlich können vorhandene Ressourcen auch von mehreren virtuellen Organisationen (VOs) gemeinsam genutzt werden. Allgemeine Definitionen zum Grid können in [TheGrid] gefunden werden. Die Vision ist, dass Rechenleistung, Information und andere Computer-Dienstleistungen für jedermann erhältlich sein sollen wie der Strom aus der Steckdose. Hieraus leitet sich auch der Name ab – von dem englischen Wort für das Stromnetz: Power Grid.

Auch die anderen Experimente am LHC stehen vor ähnlichen Herausforderungen. Daher initiierte CERN EU-weite Grid-Projekte, zunächst das European Datagrid (EDG [EDG]) mit 21 Partner-Instituten. Nach diesem erfolgreichen Grid-Middleware-Projekt läuft seit April 2004 das EGEE-Projekt (Enabling Grids for E-Science in Europe [EGEE]). Der Schwerpunkt dieses Projekts ist nicht mehr Softwareentwicklung, sondern der Aufbau eines funktionierenden europaweiten Grids. Auch die LHC-Experimente haben früh damit begonnen, die Entwicklung der Computing-Umgebung zu koordinieren. Das gemeinsame LCG-Projekt (LHC Computing Grid, [LCG]) besteht aus zwei Phasen. In der 1. Phase bis 2005 soll der Prototyp eines weltweit verteilten Grids aufgebaut werden, der ca. 50 % der Komplexität, die für ein LHC-Experiment benötigt wird, ausmacht. Die dabei gewonnen Erfahrungen fließen in der nachfolgenden 2. Phase (bis 2008) in den Aufbau des Produktionsgrids ein. Ein Grid ermöglicht es den LHC-Experimenten Simulation, Rekonstruktion und Analyse von Daten aus dem Experiment in einer global verteilten Computing-Infrastruktur durchzuführen.

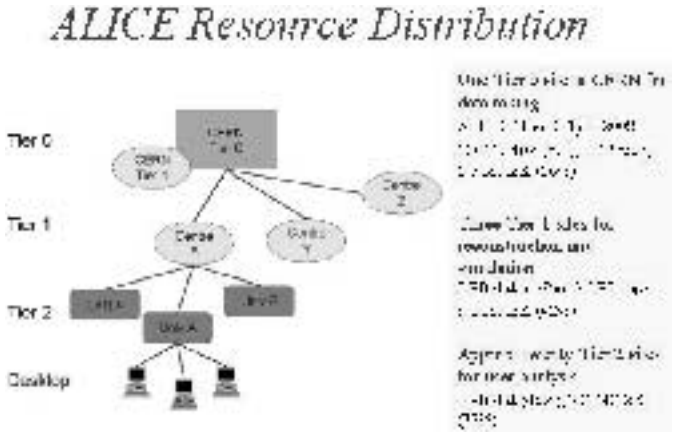

Abbildung 4: Das LHC-Computing-Modell hat eine hierarchische Struktur. Am CERN werden die Daten gespeichert. An den großen Tier1-Zentren findet die Rekonstruktion der Daten sowie Simulation statt. Auch Massenspeicher in PB-Größe sollten hier vorhanden sein. Die oft den Tier1-Zentren zugeordneten Tier2-Zentren dienen zur Simulation und Auswertung. Die Tier4-Ebene schließlich bilden die Desktop-Rechner der Wissenschaftler.

Das auf dem Grid basierende Computing-Modell der LHC-Experimente für verteiltes Rechnen hat eine hierarchische Struktur (Abbildung 4). Das ALICE-Modell ist dem allgemeinen Modell sehr ähnlich und wird im folgenden genauer beschrieben. Am oberen Ende der Hierarchie steht hierbei das CERN (Tier 0), an dem die Rohdaten aufgenommen und gespeichert werden. Die Tier1-Zentren, von denen eines am CERN und je ein weiteres in den größeren der teilnehmenden Länder steht, übernehmen den größten Teil der Rekonstruktion der Daten sowie Teile der Analyse. Hier sollten sich auch große Massenspeicher befinden, in denen die erzeugten Daten sowie insgesamt eine Kopie der Rohdaten gelagert werden. An den Tier2-Zentren, kleineren Zentren auf nationaler oder subnationaler Ebene, die einem Tier1-Zentrum zugeordnet sein können, sollte sich Simulation und Analyse die Waage halten. Tier3-Zentren schließlich entsprechen den Physikinstituten, in denen die auswertenden Wissenschaftler an ihren Rechnern, den T4-„Zentren“ sitzen.

1.3 Das ALICE-Framework

Wegen der relativ kleinen Computing-Gruppe des ALICE-Experiments wurde früh beschlossen, nicht zwei Frameworks zu entwickeln – eins für die Detektorkonstruktion und eins für die Experimentauswertung. Von Anfang an wurde ein beiden Zwecken dienliches Framework (AliRoot [AliRoot]) verwendet, welches in C++ konstruiert wurde und mit agilen Methoden ständig weiter entwickelt wird. Schon die Technical Design Reports für die unterschiedlichen Detektoren wurden mit AliRoot gerechnet. Daher bestand bereits vor der Produktionsreife der Software der EU-Grid-Projekte das Bedürfnis, große Produktionen durchzuführen und ALICE wollte die in den in Kapitel 1.2 genannten Grid-Projekten aufgebauten Ressourcen so bald wie möglich für die Experimentvorbereitung nutzen. Zu diesem Zweck wurde ein eigenes leichtgewichtiges Grid-Paket, AliEn (ALICE Environment [AliEn]) entwickelt, welches die für die verteilte Simulation, Rekonstruktion und Analyse von Daten notwendigen Grid-Komponenten zur Verfügung stellt. Insbesondere bietet AliEn ein global verteiltes Filesystem und die Möglichkeit, Jobs in einer verteilten Umgebung auszuführen. Das Zusammenspiel der einzelnen Komponenten des ALICE-Frameworks wird in Abbildung 5 verdeutlicht.

AliRoot basiert in direkter Weise auf dem ROOT-Framework [ROOT], welches die benötigte Leistung und Funktionalität in einer einfachen und benutzerfreundlichen Weise zur Verfügung stellt. Neben den bereits beschriebenen Komponenten sind verschiedene Simulationsframeworks, wie Geant3 [Geant3], Geant4 [Geant4] und Fluka [Fluka], die alle mit dem gleichen User-Code über VirtualMonteCarlo [VirtualMC] angesteuert werden können, zu sehen. Schließlich werden für die Simulation noch Eventgeneratoren, wie z.B. Hijing [Hijing] oder Pythia [Pythia], benötigt.

ROOT fußt auf einem C++-Interpreter und ist um eigene Klassen und vorkompilierte Bibliotheken gut erweiterbar. Um ROOT AliEn-tauglich zu machen, wurden entsprechende C++-Klassen hinzugefügt: die „TAliEn“-Klasse – sie basiert auf der abstrakten „TGrid“-ROOT-Klasse und benutzt das AliEn-C++-API – implementiert die Basismethoden. So können sich ALICE-Benutzer entweder von der ROOT-Kommandozeile oder über selbst geschriebene ROOT-Routinen mit dem AliEn-Grid verbinden und auf verteilte Datensätze zugreifen genau so wie wenn sie lokal vorliegen würden.

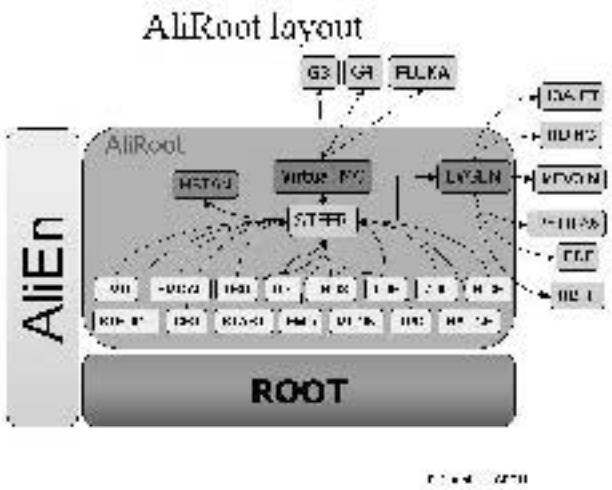

Abbildung 5: Das Zusammenspiel der einzelnen Komponenten des ALICE-Frameworks wird verdeutlicht. Zu sehen sind die drei Hauptkomponenten ROOT, AliRoot und AliEn, sowie externe Detektorsimulationspakete und Eventgeneratoren.

2 Die ALICE Grid-Implementation AliEn

Die Grid-Implementation des ALICE-Experiments, AliEn , begann für ALICE-Benutzer bereits Ende 2001 mit der Arbeit. Gegenwärtig finden mit Hilfe von AliEn vor allem verteilte Detektorsimulationen, sowie Rekonstruktion und Analyse der erzeugten Daten an über 40 Instituten auf vier Kontinenten statt. Bis Ende des Jahres 2003 liefen unter AliEn mehr als 26000 ALICE-Jobs, was 40 CPU-Jahren entspricht und mehr als 30 Tera-Byte Daten produziert hat. Bei der Entwicklung einer eigenen Gridumgebung stellte sich die Frage, ob es möglich ist, ein Grid aufzubauen, welches auf verfügbarer freier Software und offenen Standards basiert und die Bedienung sich auch dann nicht ändert, wenn zugrunde liegende Technologien ausgetauscht werden müssen. Aus diesem Grund wurde in ausgeprägtem Maße auf Modularität und Erweiterbarkeit des Systems geachtet. Auch setzte AliEn von Anfang an auf Web-Services. Auf diese Weise werden zukünftige Standards wie OGSA [OGSA] und WSRF [WSRF] reibungslos implementierbar. Von den drei Millionen Zeilen Code, aus denen AliEn derzeit besteht, basieren mehr als 95 % auf standard Open-Source-Kompomenten, die meist ohne Modifikation eingebaut wurden. Als Bindeglied zwischen den einzelnen Modulen kommt die Skriptsprache Perl [Perl] zum Einsatz, die u.a. auch gute Open-Source-Versionen von Kryptographiemodulen, gute Datenbankanbindung und ein gutes SOAP [SOAP]-Interface zur Verfügung stellt. In Abbildung 6 werden die AliEn-Komponenten auf einen Blick zusammengefasst.

2.1 zentrale Komponenten

Herzstück von AliEn ist der Filekatalog (Abbildung 7). Er bildet physikalische Filenamen (PFN), also den Filenamen inklusive der Information über den Speicherort der Datei, auf

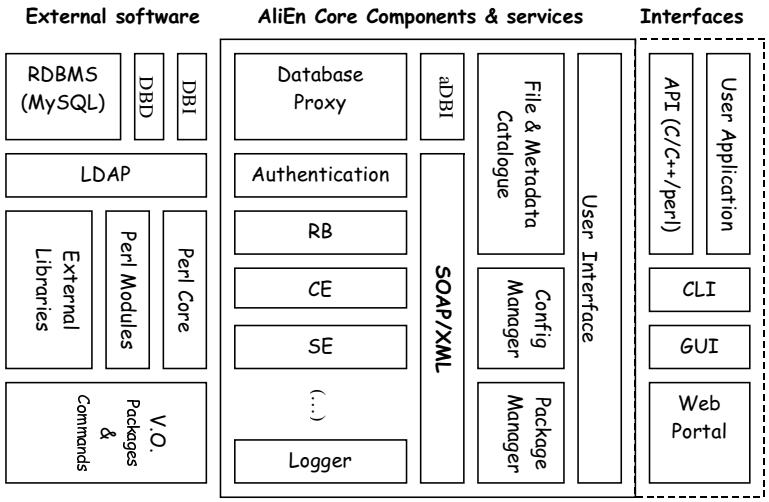

Abbildung 6: Alle AliEn-Komponenten auf einen Blick. Die Module sind nach funktionellen Gesichtspunkten geordnet und behalten jeweils ihre vollständige Eigenständigkeit.

logische Filenamen (LFN) ab, unter denen der Benutzer die Dateien im Betrieb sieht. Der Filekatalog hilft auch dabei, externe Dateien im System zu registrieren, wieder abzurufen und zu replizieren. Die technische Implementation basiert auf der relationalen Datenbank MySQL [MySQL]. Der Filekatalog sollte sich aber auch mit anderen relationalen Datenbanken verstehen. Auch darf der Administrator Teile des Verzeichnisbaums mit extra Datenbanken bedienen, die zudem auf mehreren Maschinen laufen können. Auf diese Weise skaliert das System recht gut, so dass ALICE ohne Performanceverluste schon mehr als drei Millionen Dateien registrieren konnte. Im Interface ähnelt der Katalog einem Unix-Filesystem. So kann jeder Benutzer exklusive Lese- und Schreibrechte für seine LFNs vergeben. Dateibeschreibungen sind als Metadaten speicherbar.

Die statische Konfiguration einer virtuellen Organisation in einem AliEn-Grid wird zur Laufzeit von einem LDAP-Konfigurationsserver [OpenLDAP] ausgelesen, was die Beschreibung der Benutzer und ihrer Rollen sowie von vorhandenen Softwarepaketen, Instituten und Grid-Dienste beinhaltet. Hierein spielt auch eine weitere Zentralkomponente von AliEn, der Package Manager, der alle Softwarepakete, die von VOs beigesteuert werden, automatisch verwaltet. Auf diese Weise wird es einfach, das System um Software zu erweitern, nach der eine VO verlangt. Die Pakete wissen um ihre Anforderungen an das System und kennen Abhängigkeiten von weiteren Softwarepaketen und Versionsnummern.

2.2 Pull-Architektur

Nach Installation, Setup und Start beginnen die AliEn-Services miteinander zu kommunizieren. Die zwischen den Komponenten (Diensten) ausgetauschten Nachrichten benutzen XML und SOAP. Vorteil dieses Konzepts ist, dass die Kommunikation allgemein gehalten werden kann und dass der Client keine Kenntnis über die Technik des Servers haben muss.

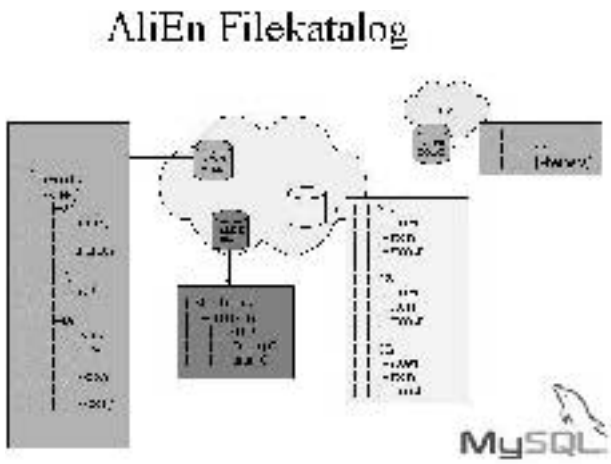

Abbildung 7: Der AliEn-Filekatalog bildet physikalische Filenamen auf ihre logischen Pendants ab. Die Übersetzung kann mit der geographischen Lage und den Fähigkeiten des betreffenden Clients variieren.

In der Praxis sitzt zwischen Client und Serverapplikation meist ein vermittelnder Proxy-Service.

Der Proxy-Service arbeitet auch für den Authentication Service. Der Authentifizierungsdienst prüft per SASL-Protokoll [SASL] die Credentials (englisch etwa für Ausweispapiere) der Benutzer. AliEn kennt die SASL-Mechanismen GSSAPI mit Globus/GSI und X509-Zertifikat, AFS-Passwort, SSH-Key und AliEn-Token).

Der AliEn Resource Broker (RB) benutzt eine Pull-Architektur, im Gegensatz zur Push-Architektur der auf Globus [Globus] basierenden Grid-Systeme. Beim „Pull“ fordert ein eingebundenes Computing-Element (siehe unten) neue Aufgaben vom Server an, sobald es wieder freie Kapazitäten hat. In dieser Konstellation muss der RB nicht den Status aller Systemressourcen kennen. Die Pull-Architektur erlaubt ein robustes System, das nicht auf die ständige Gegenwart aller Ressourcen angewiesen ist. Die lose Kopplung zwischen Ressourcen und den Resource Brokern, die die auszuführenden Programme an die Rechenressourcen verteilen, erlaubt es, fremde Grid-Systeme als AliEn-Rechen- und Speicherelemente im AliEn-Grid abzubilden (Abbildung 8). Nach diesem Prinzip entstanden Schnittstellen zum European Data Grid und zum LCG-Projekt.

2.3 Lokale Dienste

Üblicherweise läuft in einem AliEn-Client-Zentrum eine Reihe von Diensten lokal. Ein typisches Beispiel ist der Cluster-Monitor. Er fungiert als Gatekeeper, bietet also eine einheitliche Schnittstelle für alle einlaufenden Anfragen. Außerdem ist er der Proxy für die Dienste hinter dem Firewall. Ein anderer lokaler Dienst sind die Computing-Elemente. Sie bilden die Schnittstelle zu lokalen Batchsystemen. Die Storage- oder Speicherelemente

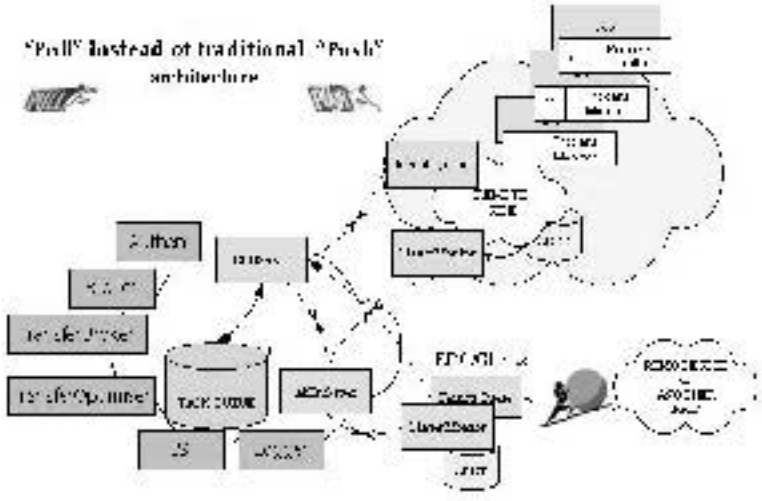

Abbildung 8: An einen AliEn-Server kann man ein AliEn-Rechenzentrum, aber auch ein anderes Grid anbinden. Zu sehen sind ein AliEn-Server und ein AliEn-Client mit den entsprechenden Services, sowie ein potentiell anbindbares weiteres Grid-System.

übernehmen das Abrufen und Speichern von Daten auf und von den lokalen Massenspeichern.

Über den Process-Monitor-Service, der als Interface zu jedem AliEn-Job fungiert, steuert der Benutzer seine laufenden Jobs. Der File-Transfer-Dienst, sein Name ist selbsterklärend, läuft auf dem selben Rechner, wie das Storage Element. Die Filetransfer-Dienste wurden als Daemons implementiert. Sie authentifizieren sich gegenseitig mit den von der AliEn-CA (Zertifizierungsstelle) ausgestellten Zertifikaten und führen Datenübertragungen im Namen des Benutzers durch.

Als Teil des AliEn-Monitoring-Moduls installiert sich das Mona-Lisa-System. Es sammelt Statusinformationen aus dem Grid ein und veröffentlicht sie über Webdienste. Nutznießer sind AliEn-Optimierprogramme und Visualisierungen (Abbildung 9).

2.4 Benutzerschnittstellen

Die Benutzer bedienen AliEn über einen User Interface Layer. Die Kommandozeilenversion hat die wichtigsten Unix-Filesystemkommandos eingebaut. Daneben gibt es ein grafisches Benutzerinterface und ein generisches Webportal. Mit ihnen kann der Benutzer große Mengen zu verteilender Jobs abschicken, inspizieren und manipulieren.

Um auch Anwendungsprogrammen Zugang zu AliEn zu verschaffen, bedarf es passender Schnittstellen (APIs). Neben einem dafür geeigneten Perl-Modul stellt AliEn Programmierern ein C- und ein C++-API zur Verfügung. So wurde mit dem C++-API – es ist Thread-safe – eine Variante des Open-Source-Projekts LUFS [LUFS] implementiert, die den AliEn-Filekatalog einklinkt. Benutzer des Konstrukts führen ein entsprechendes

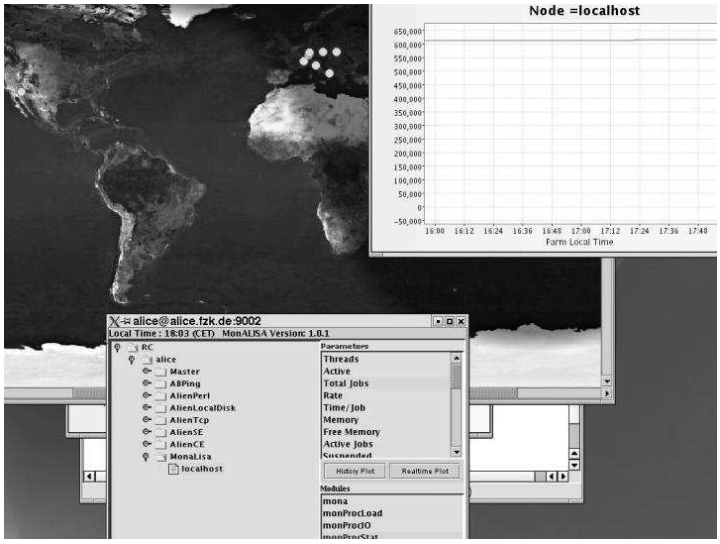

Abbildung 9: Das Mona-Lisa-Monitoring-System zeigt angeschlossene Rechenzentren (gelbe Kreise). Zudem sind rechnende AliEn-Jobs in Karlsruhe zu sehen.

„mount"-Kommando aus, melden sich beim Grid an – und haben Zugriff auf das AliEn-Filesystem.

3 ALICE-Data-Challenges

Um das ALICE-Computing-Modell testen zu können, wurden sogenannte Data Challenges eingeführt. Es gibt die Computing Data Challenges, die 1997 damit anfingen, die ALICE-Anforderungen vom Datenaufnahmesystem (DAQ) bis zum lokalen Massenspeicher zu testen. Hardware- und Softwarekomponenten werden unter realistischen Bedingungen überprüft, um eine frühe Integration in das Gesamtkonzept der ALICE-Computing-Infrastruktur zu ermöglichen.

Desweiteren gibt es die sogenannten Physics Data Challenges, mit denen primär die Auswertesoftware und das verteilte Computing-Modell getestet werden soll.

3.1 Motivation

Ziel eines ALICE Physics Data Challenge ist die Validierung des verteilten Computing Modells (vergl. Abbildung 4) sowie zu testen, ob das Offline Framework in der Lage ist, die anfallenden Datenmengen korrekt zu verarbeiten. So fand im Jahre 2001 der 1%-Data Challenge statt, mit dem überprüft werden sollte, ob das System mit einem Prozent der Daten zurecht kommt, die während eines Standardbetriebsjahres ab Experimentstart im Jahr 2008 durch Simulation anfallen. 2002 fand dann der 5 %-Data Challenge statt und seit Februar 2004 läuft der ALICE-Physics Data Challenge 03, der 10% der finalen Daten-

menge bewältigen soll (vergl. Abbildung 10). Aufgrund von technischen Schwierigkeiten hat sich PDC03 um einen Monat gegenüber der ursprünglichen Planung verspätet.

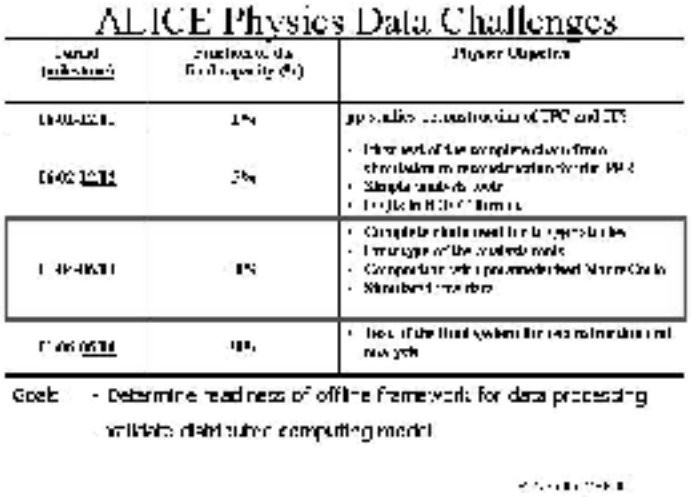

Abbildung 10: Die „Milestones“ der ALICE Physics Data Challenges. PDC01 sollte mit 1% der Datenmenge zurechtkommen, die während eines standard ALICE-Betriebsjahres anfallen. PDC02 mit 5% und PDC03 mit 10% der finalen Datenmengen.

Eine weitere wichtige Frage, die im Laufe des aktuellen Data Challenge PDC03 geklärt werden soll, ist, wieviel Computing-Kapazität am Tier0/1-Zentrum am CERN zur Verfügung steht und welcher Prozentsatz der Datenrekonstruktion am CERN gerechnet werden soll. In einem ursprünglichen Computing-Modell war zunächst geplant, annähernd die gesamte erste Rekonstruktionsphase am CERN zu rechnen. Inzwischen deutet sich an, dass diese Aufgabe wohl eher von allen Tier1-Zentren gemeinsam unternommen werden wird.

3.2 ALICE Physics Data Challenge III

Weil das sogenannte „Jet Quenching“ als starkes Indiz für das Erzeugen des QGP gilt, hat sich der Interessenschwerpunkt in der Schwerionenphysik deutlich in diese Richtung bewegt. Einer der normalerweise paarweise erzeugten Jets (Teilchenstrahlbündel) wird hierbei stark unterdrückt, wenn er den QGP-Feuerball durchqueren muss. Abschätzungen zeigen, dass ungefähr 10^5 Blei-Blei-Kollisionen mit dem HIJING-Eventgenerator erzeugt werden müssen, um Jet-Quenching im 20 und 200 GeV-Bereich studieren zu können. Da allein die Simulation eines einzigen Pb-Pb-Ereignisses 18000 kSI2k x s benötigt und 600 MB Plattenplatz belegt, folgt hieraus, dass große Mengen an Computing Ressourcen zum Einsatz kommen müssen. Die benötigte Rechenzeit kann jedoch dadurch ein wenig reduziert werden, dass zunächst eine sinnvolle Zahl von Pb-Pb-Untergrundereignissen simuliert wird, die dann in der Rekonstruktionsphase mit simulierten Ereignisevents kombi-

niert werden können. Hierbei ist wichtig, dass ein Untergrundereignis bei den angestrebten Untersuchungen bis zu 50 mal wiederverwendet werden kann.

Die Produktion wird unter Benutzung von AliEn durchgeführt. LCG wird als eins von vielen Computing Elements in die AliEn-Gridumgebung eingebunden. Die Produktionsjobs werden also an das AliEn-System abgeschickt. Dann können sie entweder den Weg über AliEn Computing Elements an native AliEnzentren nehmen oder aber über ein AliEn-LCG-Interface, welches auf einigen Schnittstellenzentren installiert ist, vom LCG-Resource Broker an ein LCG-Zentrum weiterverwiesen werden. An einigen wenigen Zentren, so z.B. GridKa [gridka] und CNAF, können die selben Rechenknoten sowohl via AliEn als auch über LCG angesteuert werden (Abbildung 11).

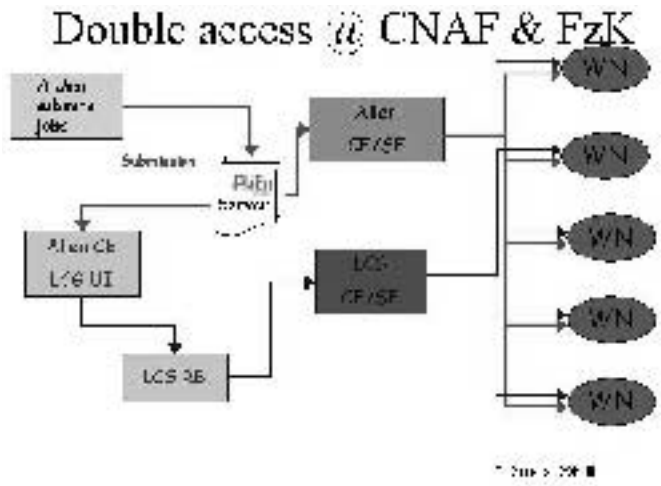

Abbildung 11: Die an das AliEn-Grid abgeschickten Jobs können entweder auf nativen AliEn-Sites oder an Zentren gerechnet werden, auf denen LCG installiert ist. An einigen Zentren konnten die selben Rechenknoten mit beiden Grid-Systemen angesprochen werden.

Der PDC03 erstreckt sich über die ersten beiden Quartale des Jahres 2004 und kann in drei Abschnitte eingeteilt werden. Im ersten Abschnitt werden Ereignisse an allen weltweit verfügbaren Zentren produziert, was bei der an den Tier1-Zentren verfügbaren Computerkapazität auf insgesamt etwa 80 MByte an simulierten Daten pro Sekunde hinausläuft. Anschließend werden alle oder ein Großteil der Daten zum CERN übertragen. Dies bedeutet, dass in dieser Phase 90 TB innerhalb von 2 Monaten zum CERN transferiert werden, welches eine durchschnittliche Bandbreite am CERN von 160 Mb/s benötigt. Die Idee hierbei ist, dass in der Rekonstruktionsphase im zweiten Abschnitt davon ausgegangen werden kann, dass alle Experimentdaten vom CERN kommen, was ja auch im Experimentbetrieb ab 2008 der Fall sein wird. Die Rekonstruktionsarbeit wird von allen großen Computerzentren in gleichem Maße durchgeführt werden, wobei hier wieder ein gleichmäßiger Datenstrom vom CERN aus in die Welt geht. Die dritte Phase des PDC03 ist die Analysephase. In dieser Phase werden Physiker aus aller Welt mit Hilfe des grid-basierten

Analysesystems, bestehend aus ROOT, PROOF und AliEn, die weltweit verteilten Datensätze analysieren. Weil aber AliEn darauf ausgelegt ist, die Datenübertragungen minimal zu halten, sind die Netzanforderungen in dieser Phase eher gering. Der Ressourcenverbrauch während der einzelnen Phasen des Data Challenges wird in der Abbildung 12 noch einmal veranschaulicht. CERN übernimmt hierbei einen geschätzten Anteil von 15 Prozent der Computing-Ressourcen.

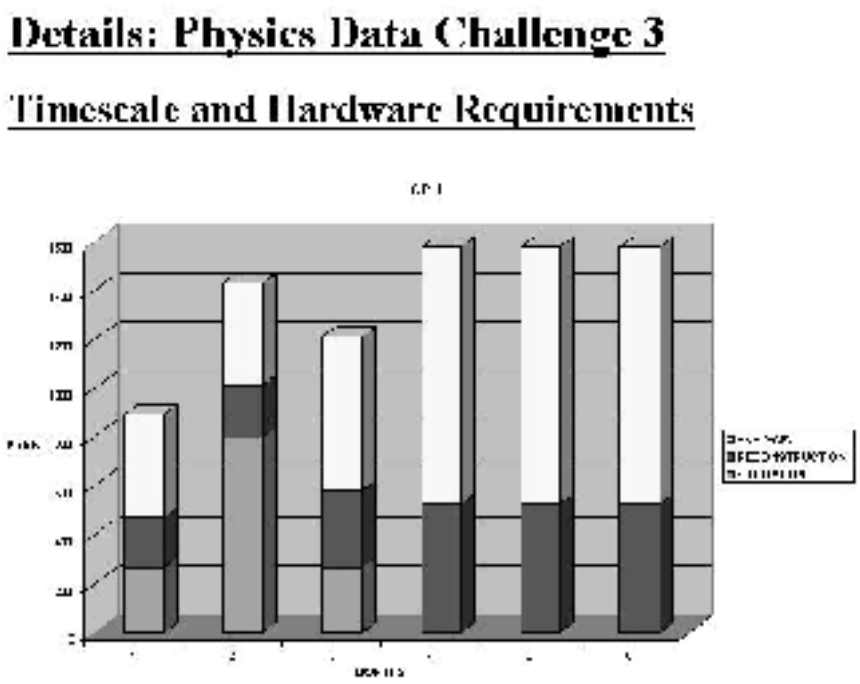

Abbildung 12: Entwicklung der CPU-Anforderungen während der sechs Monate des dritten ALICE Data Challenges. Der Verbrauch ist auf die einzelnen Phasen Simulation, Rekonstruktion und Analyse aufgeteilt.

Phase 1 des Physics Data Challenges, die Simulationsphase, wurde innerhalb der geplanten Zeit erfolgreich abgeschlossen. Im Mittel liefen auf das ganze Grid verteilt 550 bis 600 ALICE-Jobs gleichzeitig, wobei ein Simulationsjob auf einem heutigen Standardcomputer ungefähr 5 bis 6 Stunden dauert. In Abbildung 13 ist die Masterqueue zu sehen, in der sämtliche laufende Jobs angezeigt werden, sowie die AliEn-Queues der größeren Zentren und die LCG-Queues, die sich allerdings auf mehrere Zentren erstrecken.

Die Aufteilung der Rechenarbeit auf die einzelnen an das AliEn-Grid angeschlossenen Zentren sowie die Gewichtung der beiden verwendeten Grid-Systeme wird in Abbildung 14 gezeigt. Von den bis zum 24. Mai 2004 produzierten 57137 Datenfiles sind etwa ein Drittel unter LCG gerechnet worden, darunter wiederum die Meisten in Italien. Unter den

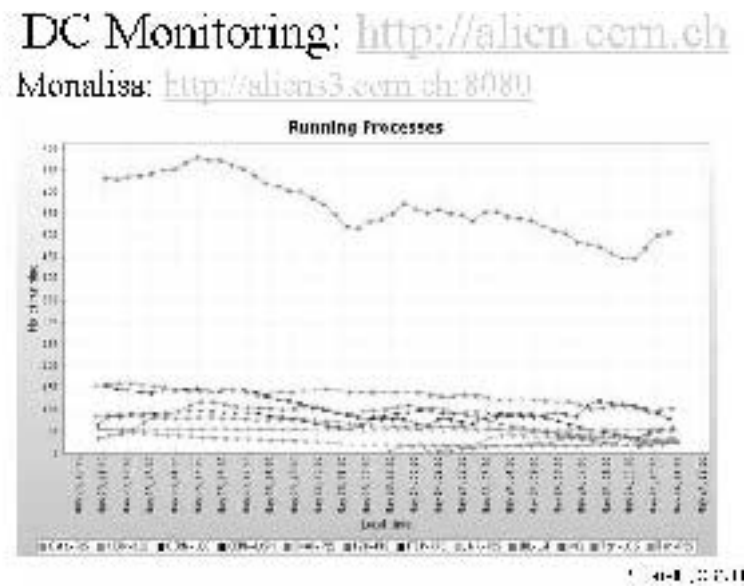

Abbildung 13: Die während des PDC03 gleichzeitig laufenden ALICE-Jobs werden in einem Zeitintervall von 24 Stunden gezeigt. Zu sehen sind die Masterqueue, die alle Jobs beinhaltet, sowie die einzelnen Queues der größeren Zentren und von LCG.

nativen AliEn-Sites war das GridKa (Forschungszentrum Karlsruhe) mit 8000 Jobs am produktivsten.

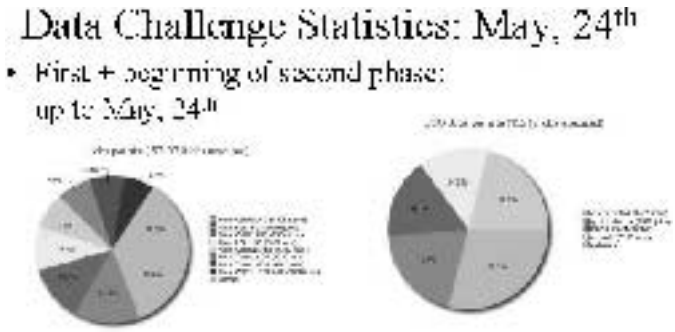

Abbildung 14: Die Aufteilung der Jobs auf die einzelnen Zentren und Grid-Systeme. 2/3 wurden unter AliEn, 1/3 unter LCG gerechnet. Die meisten LCG-Jobs wurden am CNAF, Italien, mit 4401 Jobs gerechnet. Die meisten AliEn-Jobs am GridKa mit 8000 Jobs.

4 Ausblick: verteilte Analyse mit PROOF

Interaktive Analysen von großen Datenmengen an einer lokalen Rechnerfarm sind mit PROOF, der „Parallel ROOT Facility“, einer Erweiterung des ROOT-Frameworks, möglich. Üblich ist, dass der Benutzer sich mit einem PROOF-Master-Server interaktiv verbindet, der wiederum sich auf den Batch-Knoten einer Rechnerfarm befindende PROOF-Slave-Server verwaltet, in dem er Arbeiten verteilt und die Ergebnisse einsammelt.

In Kapitel 1.3 wird beschrieben, wie sich ROOT, und damit auch PROOF, erweitern lässt, so dass es mit Grids und insbesondere AliEn zusammenarbeitet. Auf diese Weise kann man die PROOF-Umgebungen mehrerer Institute zusammenschalten, das Ergebnis wird „SuperPROOF“ genannt. Hierbei wird jedes Zentrum als SuperPROOF-Slave-Server betrachtet. Der SuperPROOF-Master-Server läuft auf der Maschine, auf der auch der AliEn-Server läuft (Abbildung 15).

In einer dynamischen Umgebung startet ein AliEn-Grid-Service auf Nachfrage PROOF-Daemonen, die in dedizierten Queues der Batchsysteme der lokalen Zentren laufen. Der SuperPROOF-Master verbindet sich über einen AliEn-TCP-Routing-Service mit den PROOF-Daemonen in den verteilten Zentren. AliEn weiss, wo die zu analysierenden Daten liegen, und lässt die C++-Analyse-Makros an jenen Zentren laufen, die die Datensätze vorhalten.

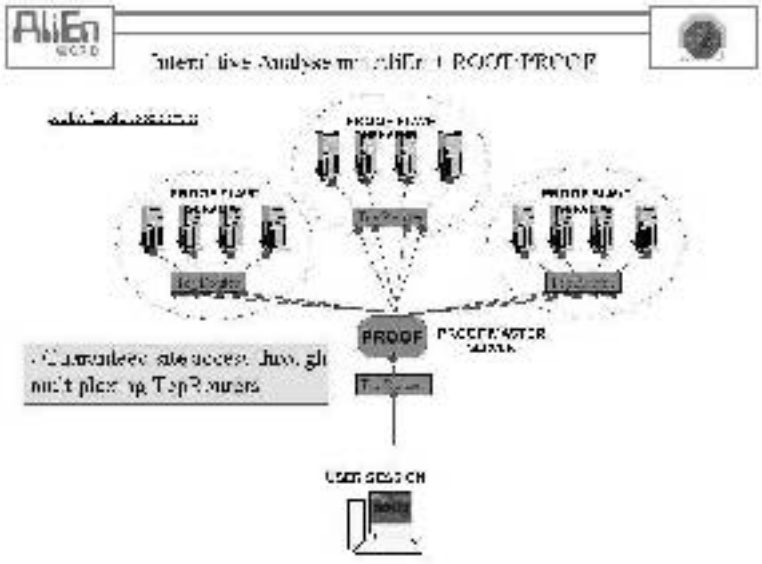

Abbildung 15: Die mit Hilfe von AliEn zusammengeschalteten PROOF-Umgebungen mehrerer Institute werden als SuperPROOF bezeichnet. Mit Hilfe von SuperPROOF können weltweit verteilte große Datenmengen interaktiv und parallel analysiert werden.

Literatur

[ALICE] http://alice.web.cern.ch/Alice/AliceNew

[LHC] http://lhc-new-homepage.web.cern.ch/lhc-new-homepage/

[CERN] http://public.web.cern.ch/public/

[ALICE-TDR] http://alice.web.cern.ch/Alice/TDR/
CERN/LHCC 95-71, LHCC/P3, 15 December 1995, ISBN 92-9083-077-8

[TheGrid] Ian Foster, Carl Kesselman, The Grid-Blueprint for a New Computing Infrastructure, Morgan Kaufmann Publishers (1998)

[EDG] http://www.eu-datagrid.org

[EGEE] http://public.eu-egee.org

[LCG] http://lcg.web.cern.ch/LCG/

[AliRoot] R. Brun, P. Buncic, F. Carminati, A. Morsch, F. Rademakers, K. Safarik, Computing in ALICE, Nucl. Instr. and Meth. A502 (2003) 339-346

[AliEn] http://alien.cern.ch
P. Saiz, L. Aphecetche, P. Buncic, R. Piskac, J.E. Revsbech, V. Sego, Nuclear Instruments and Methods 2003, 437-440.

[ROOT] R. Brun, F. Rademakers, ROOT – An Object Oriented Data Analysis Framework, Nucl. Instr. and Meth. A389 (1997) 81
http://root.cern.ch

[Geant3] http://wwwasdoc.web.cern.ch/wwwasdoc/geant_html3/geantall.html

[Geant4] http://wwwasd.web.cern.ch/wwwasd/geant4/geant4.html

[Fluka] http://www.fluka.org

[VirtualMC] http://root.cern.ch/root/vmc/VirtualMC.html

[Hijing] http://www-nsdth.lbl.gov/~xnwang/hijing/

[Pythia] http://www.thep.lu.se/~torbjorn/Pythia.html

[OGSA] http://www.globus.org/ogsa

[WSRF] http://www.globus.org/wsrf

[Perl] http://www.perl.com

[SOAP] http://www.soaplite.com

[MySQL] http://www.mysql.com

[OpenLDAP] http://www.openldap.org

[SASL] http:/asg.web.cmu.edu/sasl

[Globus] http://www.globus.org

[LUFS] http://lufs.sourceforge.net/lufs/intro.html

[gridka] http://www.gridka.de