

Status Quo: Accounting im Bereich des Grid Computing

DIPL.-INFORM. MATTHIAS GÖHNER
(Hrsg.)

Institut für Informationstechnische Systeme, IIS

Bericht Nr. 2006-03
Oktober 2006

Universität der Bundeswehr München

Fakultät für

INFORMATIK

Werner-Heisenberg-Weg 39 • D-85577 Neubiberg

Inhaltsverzeichnis

1	Einleitung	1
2	Forschungsarbeiten im Bereich des Grid Accounting	3
2.1	Account Management and Information Exchange (AMIE)	4
2.2	Accounting Processor for Event Logs (APEL)	7
2.3	Distributed Grid Accounting System (DGAS)	9
2.4	Grid Accounting Services Architecture (GASA) / GridBank	11
2.5	Grid-based Application Service Provision (GRASP)	14
2.6	Grid Service Accounting Extensions (GSAX)	16
2.7	Nimrod/G	18
2.8	SweGrid Accounting System (SGAS)	19
2.9	Gesamtbewertung existierender Abrechnungssysteme	22
3	Arbeiten von Standardisierungsgremien	25
3.1	Global Grid Forum (GGF)	25
3.1.1	Grid Resource Allocation Agreement Protocol (GRAAP)	26
3.1.2	Usage Record (UR)	30
3.1.3	Resource Usage Service (RUS)	32
3.1.4	Grid Economic Services Architecture (GESA)	34
3.2	Internet Engineering Task Force (IETF)	37
3.2.1	Arbeiten im Kontext der Authentication, Authorization and Accounting (AAA)-WG	38
3.2.2	Die erweiterte Authentication, Authorization, Accounting, Auditing und Charging (A4C)-Architektur	41
	Abkürzungsverzeichnis	46
	Literaturverzeichnis	57
	Index	59

Kapitel 1

Einleitung

Das folgende Dokument bietet einen Überblick über den derzeitigen Stand der Abrechnung im Kontext des Grid Computing. Basierend auf einer umfassenden Analyse gegenwärtiger Forschungsarbeiten [GR06] [RG06] innerhalb des deutschen D-Grid-Projektes [D-G06] werden existierende Abrechnungssysteme aus europäischen als auch internationalen Grid-Projekten begutachtet und abschließend evaluiert. Weiterhin werden Arbeiten von Standardisierungsgremien wie beispielsweise dem Global Grid Forum (GGF) [Glo06] und der Internet Engineering Task Force (IETF) [IET06] untersucht und die vorgestellten Spezifikationen bezüglich ihrer Einsatzfähigkeit im Hinblick auf eine Abrechnung innerhalb von dynamischen Virtuellen Organisationen (VO) bewertet.

In Kapitel 2 werden zunächst Forschungsarbeiten im Bereich des Grid Accounting untersucht. Im Zuge dessen werden gegenwärtig existierende Abrechnungssysteme und -werkzeuge aus verschiedenen Grid-Projekten präsentiert. Hierbei wird zunächst ein kurzer Überblick über das Forschungsprojekt offeriert und im Anschluss daran Grundzüge der Architektur des zugrunde liegenden Abrechnungssystems, der enthaltenen Komponenten sowie wesentlicher Schnittstellen und Funktionalitäten der vorgestellten Module beschrieben. Im Anschluss daran erfolgt für jedes vorgestellte Abrechnungssystem eine dedizierte Bewertung. Abschließend erfolgt mittels einer tabellarischen Übersicht eine Gesamtbewertung existierender Abrechnungssysteme.

Kapitel 3 beschäftigt sich schließlich mit der Analyse von abrechnungsrelevanten Spezifikationen einschlägiger Standardisierungsgremien. Hierbei werden verschiedene Arbeiten des Global Grid Forum und der Internet Engineering Task Force untersucht. Zu den begutachteten Standardisierungsansätzen zählen unter anderem Arbeiten im Hinblick auf das Scheduling und Ressourcenmanagement sowie hinsichtlich der Beschreibung von Abrechnungs- und Nutzungsinformationen von Diensten und Ressourcen im Kontext des Grid Computing.

Darüber hinaus werden verschiedene Spezifikationen in Bezug auf das Abrechnungsmanagement vorgestellt. In diesem Zusammenhang liegt der Fokus der Betrachtungen auf den, insbesondere für die Abrechnung einer Nutzung von Netzinfrastrukturen und -diensten relevanten Spezifikationen der Internet Engineering Task Force.

Kapitel 2

Forschungsarbeiten im Bereich des Grid Accounting

In den nachfolgenden Abschnitten werden gegenwärtige Forschungsarbeiten auf dem Gebiet der Abrechnung im Bereich des Grid Computing vorgestellt. Die Beschreibung der verschiedenen zugrunde liegenden Architekturen erfolgt hierbei teilweise auf unterschiedlichen Abstraktionsebenen. Folgende, derzeit bereits zum Teil in einigen Umgebungen produktiv eingesetzte Abrechnungssysteme und -werkzeuge werden im Zuge dessen untersucht:

- Account Management and Information Exchange (AMIE)
- Accounting Processor for Event Logs (APEL)
- Distributed Grid Accounting System (DGAS)
- Grid Accounting Services Architecture (GASA) / GridBank
- Grid Based Application Service Provision (GRASP)
- Grid Service Accounting Extensions (GSAX)
- Nimrod/G
- SweGrid Accounting System (SGAS)

2.1 Account Management and Information Exchange (AMIE)

Das Account Management und Information Exchange-System, welches innerhalb des US TeraGrid-Projektes [Ter06] in Zusammenarbeit mit der Boston University entwickelt wurde, bildet ein grundlegendes Kernelement der amerikanischen Grid-Infrastruktur [CBC⁺05]. Das TeraGrid, welches die weltgrößte Cyberinfrastruktur für offene Forschung darstellt, wird seitens der National Science Foundation (NSF) finanziert und besteht gegenwärtig aus neun Ressourcen Providern, die über Hochgeschwindigkeitsnetze miteinander verbunden sind. Grundlegendes Ziel des TeraGrid ist es, Ingenieuren und Wissenschaftlern die Möglichkeit zu offerieren, komplexe Nutzeraufträge die mit zugrunde liegenden Forschungsprojekten assoziiert sind, an geeignete Grid-Ressourcen innerhalb des TeraGrid submittieren und ausführen zu können [Jor06].

Grundlegende Aufgabe des AMIE-Systems liegt in der zentralen Übernahme des automatisierten *Account Management*. Zu den wesentlichen Funktionen des Abrechnungssystems zählen unter anderem das Anlegen, Löschen und Verwalten von Nutzerkonten sowie die Übermittlung von Informationen hinsichtlich der Ressourcennutzung an die zentral verwaltete TeraGrid Central Database (TGCD) [CBC⁺05]. Grundlegende Komponenten der in [Sha04] vorgestellten Architektur des Abrechnungssystems, welche im weiteren Verlauf skizziert werden, sind hierbei die Ressourcenanbieter, so genannte „*Local Sites*“ sowie eine zentral verwaltete Datenbank, welche innerhalb der so genannten „*Central Site*“ gehalten wird.

Die Anbieter von Ressourcen innerhalb des TeraGrid sind grundsätzlich in Eigenregie für die Anbindung an das Abrechnungssystem verantwortlich, um beispielsweise eine Übermittlung der Abrechnungsinformationen über die lokale Ressourcennutzung, als auch eine Konvertierung der Abrechnungsdatensätze in das von Seiten des Abrechnungssystems spezifizierten Format zu gewährleisten. Dies schließt ferner den Fall mit ein, dass Abrechnungsinformationen des zentral verwalteten AMIE-Systems in das Abrechnungsformat des jeweiligen Ressourcenanbieters konvertiert werden müssen. Für den Betrieb von AMIE wird an das lokale Account Management jedes Ressourcenanbieters eine so genannte „*Intermediate Database*“ angefügt, welche dem Datenaustausch mit der zentralen Abrechnungsdatenbank dient [Sha04]. Bei der Intermediate Database handelt es sich um eine Referenzimplementierung eines Datenbanksystems, welche als Bindeglied zwischen der TeraGrid Central Database und dem lokalen Account Management der Ressourcenanbieter agiert. Für die Übermittlung von abrechnungsrelevanten Informationen ist von Seiten der vorgestellten Implementierung gegenwärtig das Protokoll SSH (Secure Shell) basierend auf privaten resp. öffentlichen Schlüsseln vorgesehen [Sha03].

Innerhalb der Central Site unterhält das Abrechnungssystem AMIE ein zentrales Daten-

bankmanagementsystem, welches Daten aller beteiligten Projekte und assoziierter Nutzer sowie sämtliche Autorisierungsinformationen hinsichtlich der Ressourcennutzung beinhaltet. Bei der Durchführung eines Nutzerauftrages wird die Ressourcennutzung lokal aufgezzeichnet, das mit dem Nutzer assoziierte Forschungsprojekt ermittelt und das zugehörige Projektkonto entsprechend belastet. Da prinzipiell die Möglichkeit einer Zugehörigkeit von Nutzern zu verschiedenen Forschungsprojekten besteht, unterstützt das Abrechnungssystem AMIE eine Abrechnung auf Basis verschiedener, zugrunde liegender Projektkonten. Darüber hinaus werden von Seiten der Central Site lokal durchgeführte Änderungen eines Ressourcenanbieters, beispielsweise im Falle einer Erweiterung eines Forschungsprojektes um weitere Mitglieder, an sämtliche Anbieter von Ressourcen innerhalb des TeraGrid weitergeleitet [Sha04].

Mit Hilfe verschiedener Werkzeuge sowie mittels des TeraGrid Nutzerportals, dem so genannten „*TeraGrid Frontend*“ wird seitens des Abrechnungssystems unter anderem eine Durchführung von Abfragen der zentralen Datenbank hinsichtlich Statusinformationen des Nutzers resp. Projektes ermöglicht. Weiterhin besteht die Möglichkeit, mittels des Portals eine Einsicht des Kontoguthabens des jeweiligen TeraGrid-Projektes vorzunehmen.

Abbildung 2.1 visualisiert die grundlegenden Komponenten des Abrechnungssystems sowie deren Interaktionen während der Verarbeitung eines Nutzerauftrages, welche im Folgenden beschrieben werden [Jor06] [Sha03] [Sha04].

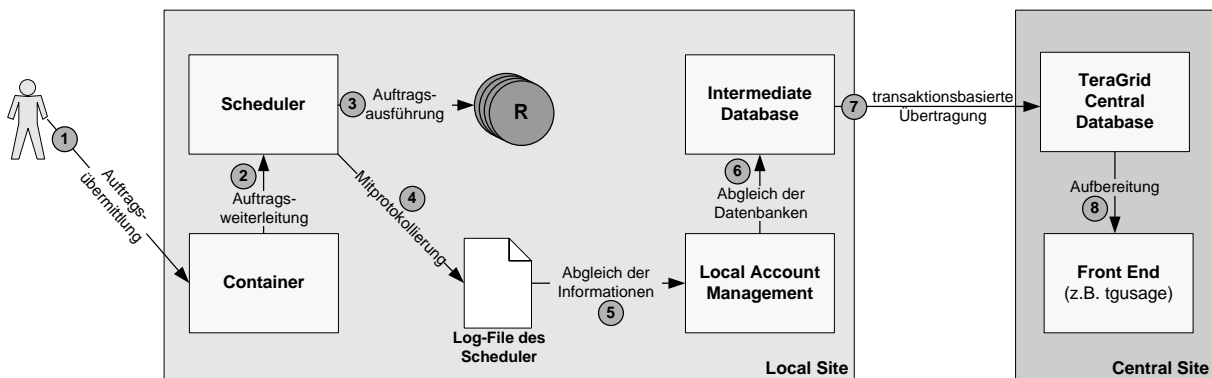


Abbildung 2.1: Architektur des Abrechnungssystems AMIE, n. [GK06].

Den ersten Schritt der Interaktionsfolge stellt die Submittierung des Nutzerauftrages an den für die Ressource zuständigen *Container* dar (1). Dieser dient dabei als Art Ausführungsumgebung grundlegender, angebotener Grid-Dienste basierend auf verschiedenen Web Service-Standards. In einem zweiten Schritt wird, auf Basis einer Auftragsweiterleitung, die Spezifikation des Nutzerauftrages an den zugehörigen Scheduler übermittelt (2). Basierend auf verschiedenen Scheduling-Algorithmen wird der Auftrag in die vorhandene Planung des Schedulers integriert und zu einem vorgesehenen Zeitpunkt auf geeigneten

Ressourcen ausgeführt ③. Parallel dazu wird die Ressourcennutzung mittels des Schedulers in einem Log-File, welches ein proprietäres Scheduler-Format aufweist, aufgezeichnet ④. Im Anschluss daran wird durch das lokale Account Management das erzeugte Log-File eingelesen und entsprechend ausgewertet ⑤, bevor ein Abgleich der Abrechnungsdaten auf Basis eines von Seiten der AMIE-Spezifikation standardisierten Formates mit der Intermediate Database stattfindet ⑥. Der Transaktionsmechanismus des Abrechnungssystems sorgt sodann für die gesicherte Weiterleitung der abrechnungsrelevanten Informationen an die TeraGrid Central Database ⑦. Mit Hilfe diverser Werkzeuge wie beispielsweise *tgusage* können abschließend von Seiten des Nutzers mittels des TeraGrid-Frontends die aufbereiteten Abrechnungsinformationen eingesehen und ferner visualisiert werden ⑧.

Bewertung des Ansatzes

Insgesamt stellt das vorgestellte Abrechnungssystem AMIE einen vielversprechenden Ansatz hinsichtlich eines automatisierten Account Management-Systems dar, welches gegenwärtig für sämtliche Forschungsprojekte innerhalb des TeraGrid seinen praxisnahen Einsatz findet. Ferner erlaubt der vorgestellte Ansatz eine Abrechnung verschiedener, weitgehend homogener Grid-Ressourcen. Darüber hinaus werden seitens der vorgestellten Spezifikation zahlreiche Sicherheitsaspekte berücksichtigt. Hierzu zählt beispielsweise die Verschlüsselung der Übertragung abrechnungsrelevanter Informationen mittels geeigneter Verfahren. Positiv fällt weiterhin die Verwendung eines Transaktionsmodells auf, das im Hinblick auf die Übermittlung der Abrechnungsdaten zwischen den Datenbanken herangezogen wird sowie der relativ geringe Einfluss des AMIE-Systems auf bestehende Abrechnungswerkzeuge.

Nachteilig wirkt sich seitens des Abrechnungssystems die fehlende Interoperabilität aus. Neben der Tatsache, dass der vorgestellte Ansatz eigene Übertragungsprotokolle definiert, müssen darüber hinaus zugrunde liegende Scheduler an die Spezifika des Systems angepasst werden. Darüber hinaus ist von Seiten der Spezifikation die Verwendung eines zentral ausgerichteten Datenbankansatzes vorgesehen. Dieser Aspekt stellt ebenso eine mögliche Schwachstelle des Ansatzes dar, wie die Tatsache, dass gegenwärtig nur wenige Abrechnungseinheiten basierend auf einer abstrakten Währung, so genannten „*Service Units*“ (SU) [Jor06] von dem System unterstützt werden. Ein weiteres Defizit stellt der Umstand dar, dass ausschließlich eine Abrechnung auf Basis von Projektkonten und nicht auf Grundlage von Virtueller Organisationen oder einzelnen Nutzergruppen möglich ist. Die basierend auf täglichen Intervallen stattfindenden Abrechnungen des AMIE-Systems sind zudem nur eingeschränkt in der Lage, der vorherrschenden, hohen Dynamik Virtueller Organisationen Rechnung zu tragen.

Dennoch stellt das vorgestellte Abrechnungssystem einen wichtigen Ansatz dar, der insbesondere den innerhalb des TeraGrid gestellten Anforderungen an die Abrechnung einer Nutzung von Grid-Ressourcen weitestgehend genügt.

2.2 Accounting Processor for Event Logs (APEL)

Das webbasierte „Abrechnungs-Werkzeug“ APEL befindet sich derzeit in den Projekten EGEE, welches auf dem LHG Computing Grid (LCG) basiert, sowie GridPP im produktiven Einsatz [Gor05]. APEL stellt eine Log Processing-Anwendung dar, welche eine Interpretation von Batch-, System- und Gatekeeper-Logdateien durchführt, die dort enthaltenen Informationen zusammenfasst und daraus Abrechnungsdatensätze, so genannte „*Accounting Records*“ generiert [BWK05]. Die Abrechnungsdaten werden innerhalb eines Grid Operations Centre (GOC) in einem zentral verwalteten, relationalen Datenbankmanagementsystem abgelegt und für eine Abfrage über das Internet konsolidiert. Verschiedenartige Plug-Ins ermöglichen den Zugriff auf die Datenbank (Abbildung 2.2), um Abrechnungsdatensätze mittels der Relational Grid Monitoring Architecture (R-GMA), einer Implementierung der Grid Monitoring Architecture (GMA), zu publizieren [BCC⁺05].

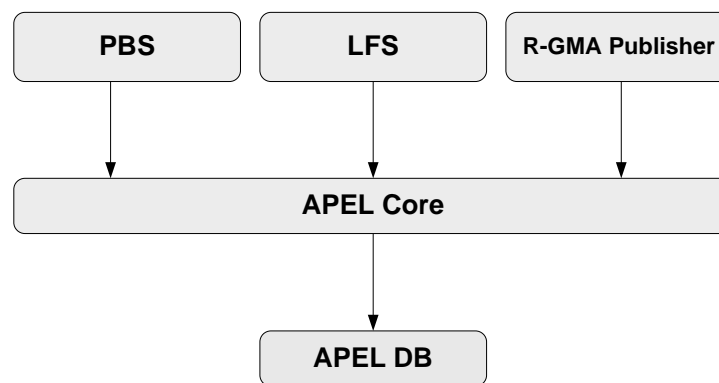


Abbildung 2.2: Schematische Darstellung des Zugriffs auf die APEL Datenbank mittels Plug-Ins, n. [BCC⁺05].

Gegenwärtig werden von dem Abrechnungswerkzeug APEL ausschließlich PBS- sowie LSF-Batchsysteme (PBS: Portable Batch System; LSF: Load Sharing Facility) unterstützt. Mittels entsprechender Plug-Ins besteht die Möglichkeit, weitere Varianten zu entwickeln und einzubinden. Das Filtern und Aggregieren der Usage Records wird mittels des Informations- und Monitoring-Systems R-GMA bewerkstelligt. Die zugrunde liegende R-GMA Architektur modelliert das Grid als Konsumenten, welche Informationen anfordern, Produzenten, welche Informationen bereitstellen sowie einer Registry, welche die Kommunikation zwischen den Produzenten und den Konsumenten vermittelt. R-GMA verwendet hierbei eine Datenbankabfragesprache, welche eine Teilmenge der deklarativen Datenbanksprache SQL darstellt [BWK05]. Konsumenten von Ressourcen haben somit die Möglichkeit, SQL-basierte Datenbankabfragen durchführen zu können, um entsprechende Einträge zu erhalten, welche im Anschluss daran von den Produzenten zur Verfügung

gestellt werden. Abbildung 2.3 verdeutlicht die grundlegende Architektur des Abrechnungssystems anhand eines Flussdiagramms.

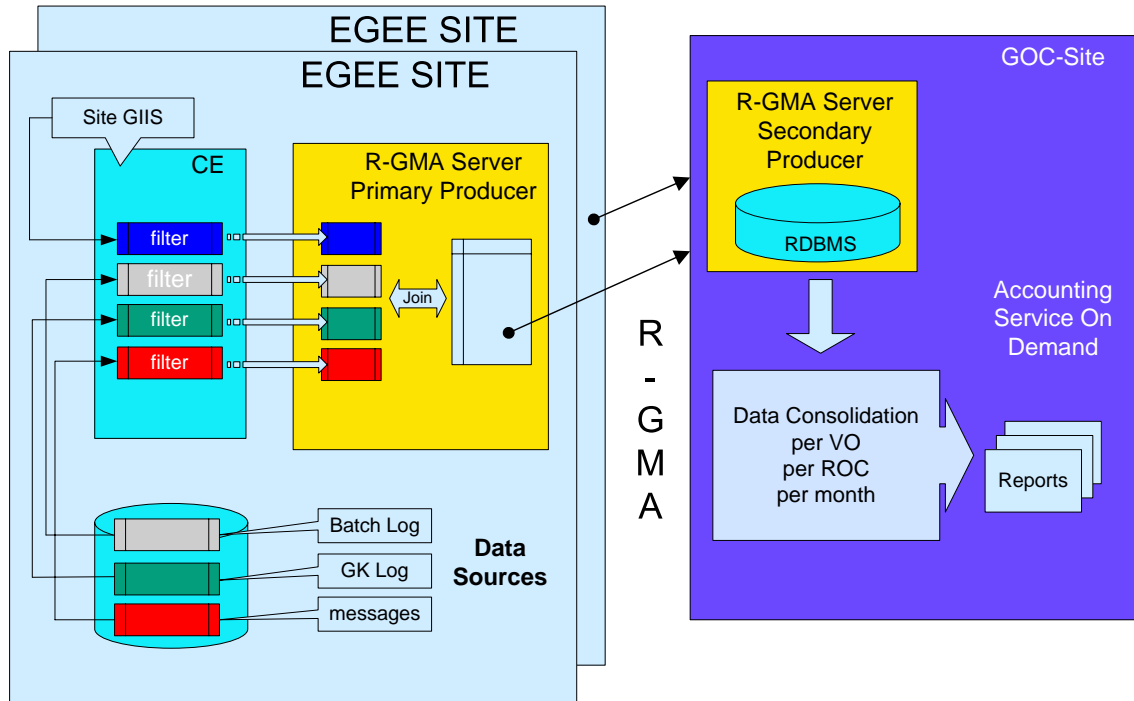


Abbildung 2.3: Grundlegende Architektur des Abrechnungswerkzeuges APEL, n. [EGE05a].

Die Abrechnungsdaten jeder einzelnen partizipierten Seite werden mittels eines so genannten „*Primary Producer*“ veröffentlicht, welcher den lokal zugewiesenen R-GMA Server verwendet. Um die Informationen von allen beteiligten Seiten zu sammeln, werden mittels eines so genannten „*Secondary Producer*“ die Daten in das zentrale Datenbanksystem des Grid Operations Centre übertragen [BWK05]. Für die Ermittlung der CPU-Auslastung der Grid-Ressourcen wird eine Anfrage an den *Grid Index Information Server* (GIIS) der jeweiligen Seite durchgeführt, auf welcher der Job submittiert und ausgeführt wurde. Von dort aus können die abgelegten „Usage Records“ aggregiert werden, um beispielsweise eine High-Level-Sicht auf die abgelegten Abrechnungsdaten zu ermöglichen [BCC⁺05].

Bewertung des Ansatzes

Das Abrechnungswerkzeug APEL besitzt einen für Grid-Umgebungen interessanten Ansatz, dessen Schwerpunkt allerdings in der Abrechnung von Batchsystem-Jobs wie beispielsweise LFS- als auch PBS-Systemen angesiedelt ist [EGE05a]. Dies erfordert die

Existenz von PBS- und Gatekeeper-Logs sowie von Systemnachrichten, welche für eine Abrechnung der Ressourcen benötigt werden. Trotz der Tatsache, dass eine Reihe von verschiedenartigen Maßeinheiten wie CPU-Zeit, Wallclock-Zeit sowie Speicherauslastung der Ressource, usw. abgerechnet werden können, werden seitens der Accounting-Architektur lediglich teilnehmende Anbieter unterstützt, welche einen hohen Grad an Homogenität hinsichtlich der eingesetzten Hardware-Ressourcen aufweisen [EGE05a]. APEL liegt ein zentraler ausgerichteter Datenbankansatz zugrunde. Ferner setzt APEL eine Umgebung voraus, die auf dem Monitoring- und Informationswerkzeug R-GMA basiert [EGE05a]. Einen vielversprechenden Ansatz stellt zudem die geplante Integration zwischen APEL und dem Workload Management System *Condor* dar [BWK05]. Somit soll es möglich sein, abrechnungsspezifische Informationen seitens des Workload Management Systems zu erhalten und diese in die Abrechnung mittels APEL mit einfließen zu lassen.

2.3 Distributed Grid Accounting System (DGAS)

Das Distributed Grid Accounting System (DGAS) wurde innerhalb des EU DataGrid Projektes (EDG) [DGA06] entwickelt und befand sich dort von 2001 bis 2004 unter dem Namen „DataGrid Accounting System“ im Einsatz [FMW04]. Seit April 2004 wird DGAS für eine Verwendung innerhalb des EGEE-Projektes unter der Bezeichnung „Distributed Grid Accounting System“ adaptiert bzw. erweitert [ABG⁺06].

DGAS liegt ein relativ stark ausgeprägter ökonomischer Ansatz zu Grunde, um die Verteilung der Ressourcen geeignet zwischen den autorisierten Grid Benutzern zu regulieren [PGW03]. Ähnlich wie beim Ansatz des SweGrid Accounting System werden sämtliche Transaktionen mittels eines dezentralen Bankdienstes durchgeführt. Die gesamte Implementierung des Distributed Grid Accounting System ist eng mit der DataGrid Workload Manager Software gekoppelt [ABC⁺03]. Abbildung 2.4 skizziert die grundlegende DGAS Architektur und stellt die Interaktionen zwischen den Komponenten graphisch dar.

Der innerhalb des Distributed Grid Accounting System vorgestellte Mechanismus basiert auf dem Konzept der *Home Location Register* (HLR). Hierbei handelt es sich um einen Ansatz, welcher sich sehr stark an einen Mechanismus anlehnt, der bei Mobilfunknetzen verwendet wird [Sch03]. Das HLR stellt ein Art Datenbanksystem dar, welches Konten für verschiedene Nutzer sowie für Ressourcen beinhaltet, die sich in einer gemeinsamen Virtuellen Organisation befinden. Des Weiteren werden mittels der Home Location Register Nutzungsdaten der Ressourcen durch den Benutzer aufgezeichnet [FMW04]. Jede Virtuelle Organisation ist mit mindestens einem HLR assoziiert, das sowohl Nutzer als auch vorhandene Ressourcen gruppiert.

Bei der so genannten „*Price Authority*“ handelt es sich um eine optionale Komponente des Abrechnungssystems, welche auf Basis verschiedener Algorithmen die Preise für die

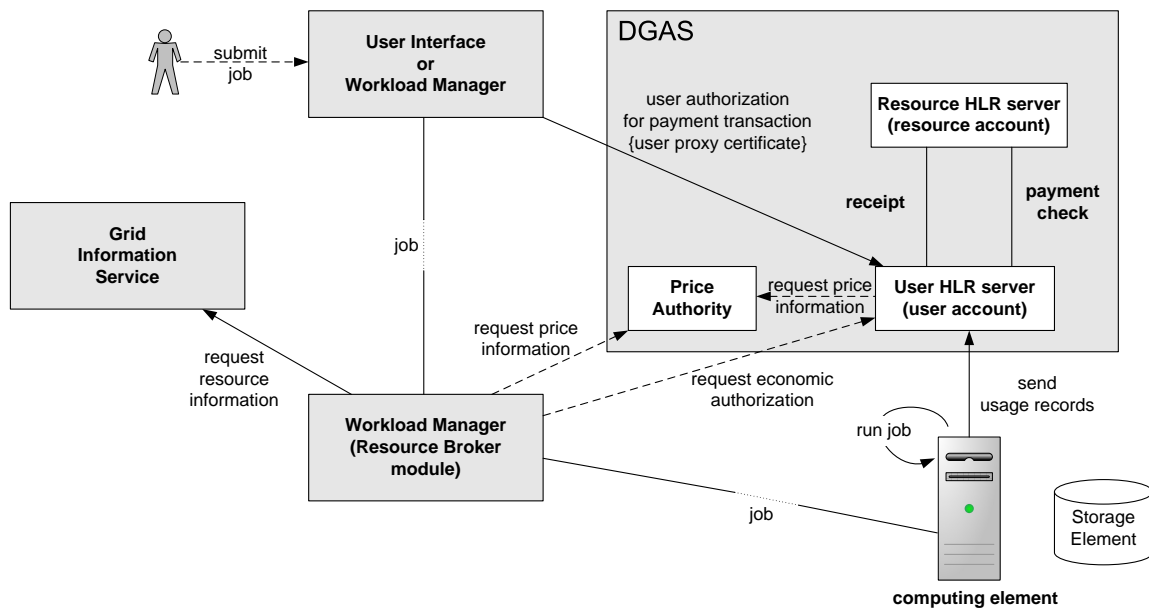


Abbildung 2.4: Architektur des Distributed Grid Accounting System, n. [ABG⁺06].

Ressourcen innerhalb des Grid festlegt. Ähnlich dem Ansatz der Home Location Register wird in jeder etablierten Virtuellen Organisation mindestens eine Price Authority eingesetzt.

Die Ermittlung der endgültigen Kosten für die Inanspruchnahme der Ressourcen findet innerhalb der HLR statt. Die Kosten der Ressourcen errechnen sich aus den anhand der Price Authority ermittelten Preisen der Ressourcen und den gewonnenen Abrechnungsdaten. Die Bilanzierung der Ressourcennutzung erfolgt mittels eines Austausches virtueller Währungseinheiten („Grid Credits“) zwischen dem Home Location Register des Nutzers und dem Home Location Register der Ressource, auf welcher der Nutzerauftrag submittiert und ausgeführt wurde [FMW04]. Ähnlich dem SweGrid Accounting System basiert DGAS, welches Bestandteil der Grid-Middleware LCG-2 ist, auf Globus Toolkit. Darüber hinaus wurde DGAS bereits erfolgreich auf einem gLite Testbed installiert [e-I05].

Bewertung des Ansatzes

Das Distributed Grid Accounting System stellt ein umfassendes Abrechnungssystem für Grid-Ressourcen dar, welches sich gegenwärtig bereits in einigen speziellen Umgebungen produktiv im Einsatz befindet [ABG⁺06]. Ein interessanter Aspekt der Architektur ist neben der Verwendung einer dezentral ausgerichteten Bankstruktur, welche auf dem Home Location Register-Ansatz basiert, die Unterstützung verschiedener zugrunde liegender

ökonomischer Marktmodelle [PGW03]. Zudem wurden innerhalb des Abrechnungssystems einige Sicherheitsaspekte berücksichtigt und umgesetzt [FMW04]. Hierzu zählt beispielsweise die Verwendung von PKI X.509 Zertifikaten sowie die Kommunikation zwischen beteiligten Entitäten mittels Secure Socket Layer (SSL). Ähnlich dem SweGrid Accounting System werden seitens DGAS so genannte „*Template Accounts*“ verwendet. Hierbei handelt es sich um temporäre Nutzer-Accounts der Ressourcenanbieter, welche autorisierten Nutzern resp. Nutzergruppen für die Dauer einer Jobausführung zugewiesen werden und somit einen flexiblen Ansatz hinsichtlich der Abbildung von Nutzern auf die entsprechenden Ressourcen bieten [Acc03].

Nachteilig wirkt sich seitens DGAS die teilweise fehlende Interoperabilität aus. Zahlreiche Standards wie beispielsweise die Open Grid Services Architecture (OGSA) bzw. Open Grid Services Infrastructure (OGSI) wurden von Seiten der Architektur nicht berücksichtigt [SGE⁺04]. Zudem liegt dem Distributed Grid Accounting System ein zu großen Teilen zentral ausgerichteter Ansatz vor. Zum Einen verwendet die zugrunde liegende Architektur einen zentral verwalteten Ressourcen-Broker [KACJ05] [EGM⁺03], zum Anderen müssen die Home Location Register, welche über das Grid verteilt sind, von einer zentralen Instanz verwaltet werden. Aufgrund der Tatsache, dass jede Ressource ein eigenes Home Location Register besitzt, welches von einer zentralen Instanz verwaltet werden muss, besteht die Möglichkeit, dass in manchen Fällen ein großer Overhead vorhanden sein kann [QL04]. Ein weiteres Defizit stellt die Tatsache dar, dass seitens der zugrunde liegenden Architektur größtenteils im Sinne einer Gutschrift lediglich eine Verrechnung des Ressourcenverbrauchs stattfindet und eine Abrechnung der Ressourcennutzung zum Teil nicht berücksichtigt wird [EGM⁺03].

2.4 Grid Accounting Services Architecture (GASA) / GridBank

Bei der Grid Accounting Services Architecture (GASA), deren konkrete Implementierung als *GridBank* bezeichnet wird, handelt es sich um ein sicheres, Grid-weites Abrechnungs- und Bezahlungssystem, welches im Zuge des australischen Grid-Bus Projektes entwickelt wurde [BB03]. Im Gegensatz zu SGAS und DGAS verwendet GridBank eine Architektur, in welcher die Registrierung der Ressourcenanbieter und der Nutzer von Grid-Ressourcen mittels eines zentral verwalteten Servers erfolgt. Infolgedessen besteht nicht der Bedarf, dass für Konsumenten von Grid-Ressourcen ein Konto bei dem jeweiligen Ressourcenanbieter angelegt sein muss [FMW04].

Die modular aufgebaute Architektur des *GridBank-Servers*, die in drei Ebenen (Accounting Layer, Payment Protocol Layer und Security Layer) unterteilt ist, ermöglicht die

Durchführung von Modifikationen und den Austausch einzelner Komponenten, ohne dass auf weitere Module des Abrechnungssystems Einfluss genommen wird. Der zentral ausgerichtete GridBank-Server kann allerdings eine mögliche Schwachstelle hinsichtlich der Skalierbarkeit als auch der Sicherheit darstellen, so dass eventuell die Grid Accounting Services Architecture zukünftig eine Migration zu einer vollkommen verteilten Struktur durchführen wird [FMW04]. Abbildung 2.5 stellt den Informationsfluss innerhalb der GridBank Architektur schematisch dar. Für eine ausführliche Beschreibung der Interaktionen zwischen den einzelnen Komponenten sei auf [BB03] verwiesen.

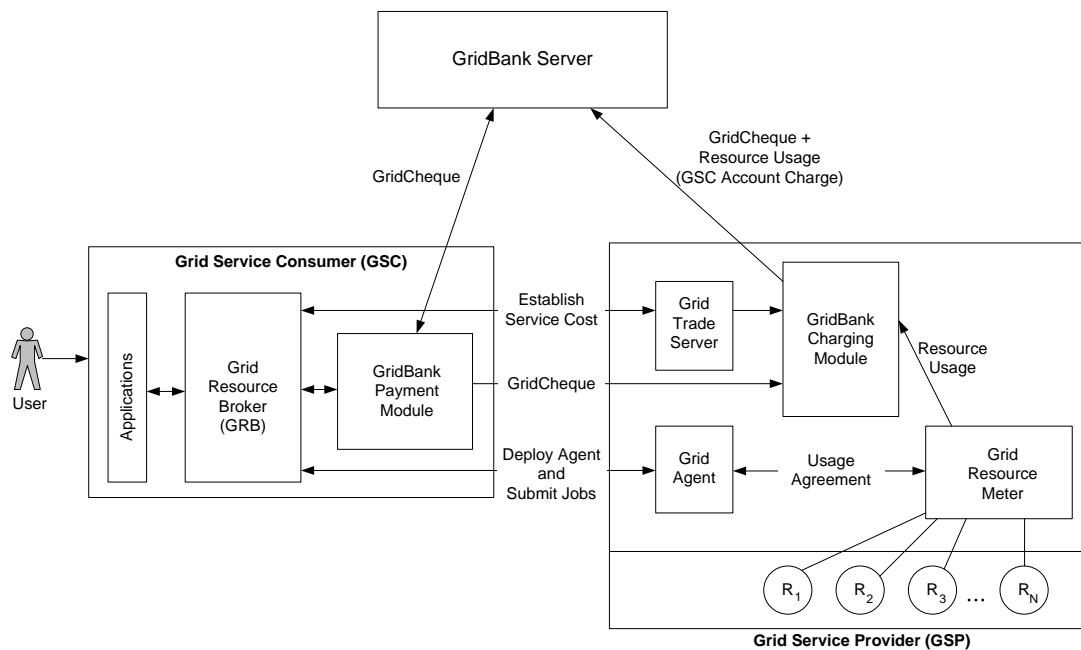


Abbildung 2.5: Interaktion von GridBank mit verschiedenen Grid-Komponenten, n. [BB03].

Wesentliche Komponenten der Grid Accounting Services Architecture sind die Ressourcenanbieter, die so genannten „Grid Service Provider“ (GSP) sowie die Konsumenten von Ressourcen, so genannte „Grid Service Consumer“ (GSC), für welche jeweils ein Konto auf dem zentralen GridBank-Server eingerichtet wird. Hauptaufgabe der *Grid Resource Broker*-Komponente (GRB) ist es, basierend auf dem übermittelten Nutzerauftrag und den existierenden Anforderungen des Benutzers, eine geeignete Ressource auszuwählen. Hierfür erfolgt eine Interaktion des Grid Resource Brokers mit dem *Grid Trade Server* (GTS), um Kosten für die Inanspruchnahme der Ressource auszuhandeln und einen geeigneten Ressourcenanbieter zu selektieren [BB03]. Für die Kommunikation zwischen dem Grid Resource Broker und dem Grid Trade Service werden seitens GASA verschiedene Protokolle unterstützt. Das *GridBank Payment Module* als auch das *GridBank Charging Module* sind für die Abwicklung der finanziellen Transaktionen auf Seiten des Nutzers

respektive des Anbieters der Ressource zuständig. Das *Grid Resource Meter* (GRM) sammelt und aggregiert Informationen darüber, welche Ressourcen für die Ausführung des Nutzer-Jobs in Anspruch genommen wurden und wie hoch die Auslastung bzw. der Verbrauch der Ressourcen war. Abbildung 2.6 veranschaulicht den schematischen Aufbau des Grid Resource Meter.

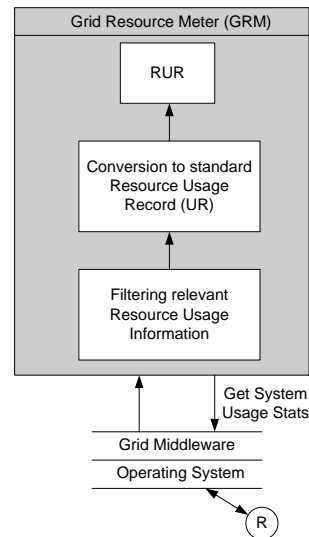


Abbildung 2.6: Aufbau der Grid Resource Meter-Komponente, n. [BB03].

Um Informationen über den Verbrauch der Grid-Ressourcen zu extrahieren, muss das Grid Resource Meter mit dem lokalen Resource Allocation System (Cluster Scheduler) bzw. einem Resource Broker wie beispielsweise dem Nimrod/G-Agenten gekoppelt werden. Die hierfür notwendige Schnittstelle wird von Seiten der Grid-Middleware wie z. B. Globus Toolkit zur Verfügung gestellt. Zur Speicherung der gefilterten Rohdaten werden ähnlich dem SweGrid Accounting System die vom Global Grid Forum spezifizierten Usage Records (UR) verwendet [BB03]. Hierbei handelt es sich um ein XML-basiertes Datenformat, welches zur einheitlichen Speicherung von Abrechnungsinformationen verwendet wird.

Bewertung des Ansatzes

Insgesamt stellt die Grid Accounting Services Architecture ein Abrechnungs- und Bezahlungssystem zur Verfügung, in welchem bereits eine Vielzahl interessanter Aspekte umgesetzt wurden. Hierzu zählt sowohl die Unterstützung standardisierter Protokolle wie beispielsweise Usage Records (UR), als auch die Integration zahlreicher Sicherheitsmechanismen wie z. B. PKI X.509 Zertifikate und Zugriffslisten [FMW04]. Sämtliche Transaktionen der

Banken werden seitens GASA über sichere Verbindungen durchgeführt. Die Grid Accounting Service Architecture ermöglicht außerdem die Anwendung verschiedener Bezahlungsschemata und unterstützt neben virtuellen Grid-Währungseinheiten (Grid Credits) die Bezahlung mittels einer monetären Währung [BB03]. Als Abrechnungseinheiten werden gegenwärtig bereits zahlreiche Metriken wie Netzverkehr, verwendeter Speicher als auch CPU- und Wallclock-Zeit unterstützt. Des Weiteren ermöglicht die modulare Architektur eine einfache Modifikation und Wartung existierender Komponenten des Abrechnungssystems.

Als nachteilig wirkt sich seitens der Grid Accounting Services Architecture aus, dass keine offenen, standardisierten Grid-Protokolle wie beispielsweise OGSA oder Web Services unterstützt werden und somit ein geringer Grad an Interoperabilität mit existierenden Grid-Systemen gewährleistet wird [SGE⁺04]. Aufgrund des zentral verwalteten GridBank-Servers liegt dem gesamten Abrechnungssystem eine teilweise zentral angeordnete Ausrichtung zu Grunde. Da sämtliche Nutzerkonten auf dem GridBank-Server angelegt sind, kann der Ausfall dieser Komponente eine mögliche Schwachstelle des Systems darstellen.

2.5 Grid-based Application Service Provision (GRASP)

Das europäische Forschungsprojekt GRASP [GRA06b] realisiert, aufbauend auf dem OGSI.NET Framework, eine Reihe von Diensten und Komponenten, welche für Application Service Provider (ASP) den Einsatz von Grids ermöglichen [Szc05] [DRY⁺03]. In diesem Zusammenhang wird neben der Realisierung eines Prototypen und einer Validierung mittels Anwendungen aus den Bereichen e-Science/e-Learning und der verteilten Datenanalyse ferner untersucht, inwieweit existierende Geschäftsmodelle auf der Basis von Grid-Technologien realisierbar sind [D-G04].

Ein grundlegendes Konzept in GRASP stellt das so genannte „*Virtual Hosting Environment*“ (VHE) dar. Ein VHE beinhaltet neben verschiedenartigen Ressourcen zahlreiche einfache Dienste, die innerhalb einer Virtuellen Organisation angeboten werden [D-G04]. Das Abrechnungs-Teilsystem wird mittels zwei Grid-Diensten, dem *Financial Accounting Service* und dem *Charging Service* realisiert. Abbildung 2.7 zeigt eine statische Sicht auf die Architektur des Abrechnungsmodells, welchem ein größtenteils ökonomischer Ansatz zu Grunde liegt.

Der Abrechnungsdienst ermöglicht die Ausführung von Anwendungen bzw. Aufträgen, indem er weitere Teilsysteme koordiniert, welche dem so genannten „*Business Component*

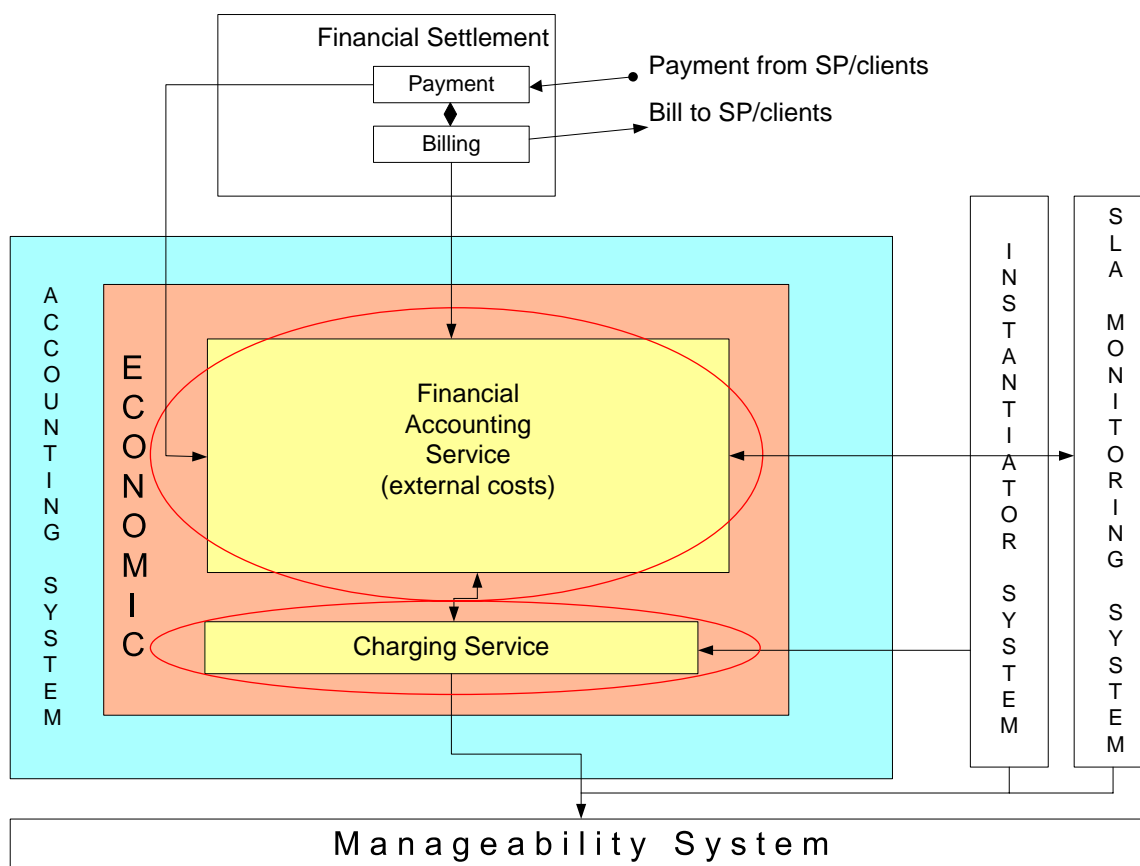


Abbildung 2.7: Statische Sicht auf das GRASP Abrechnungssystem, n. [GRA05].

„Services Layer“ zuzuordnen sind [GGH⁺05]. Vor der Ausführung eines Nutzungsauftrages oder einer Anwendung, besteht die Möglichkeit seitens des Financial Accounting Service, der Nachfragerseite eine grobe Abschätzung der Kosten für die Inanspruchnahme der Grid-Ressourcen zukommen zu lassen. Diese berechnet sich auf Basis der gegenwärtigen Auslastung sowie der Erreichbarkeit der Ressource. Nach der Submittierung des Nutzungsauftrages werden von Seiten des Abrechnungsdienstes Daten über den Verbrauch bzw. die Auslastung der Ressource gesammelt. Die akkumulierten und aggregierten Informationen über die Ressourcennutzung werden für die im Anschluss daran folgende Rechnungsstellung über so genannte „Notification Interfaces“ an das Billing-Subsystem übermittelt [GGH⁺05].

Grundsätzlich sind für GRASP eine verbrauchs- sowie eine service-level-basierte Abrechnung vorgesehen [GRA05]. Die service-level-basierte Abrechnung stützt sich hierbei auf Service Level Agreements (SLAs), welche in einem automatisch zu verarbeitenden XML-Format mittels Web Service Level Agreements (WSLAs) und so genannten „Business Extensions“ spezifiziert werden und somit eine flexible, dynamisch konfigurierbare Ab-

rechnung der Ressourcen ermöglichen sollen [GRA05].

Bewertung des Ansatzes

GRASP entwickelt interessante Komponenten im Umfeld des Application Service Provision, allerdings befinden sich die Komponenten für die Abrechnung sowie die Erfassung der Gebühren für die Ressourcennutzung innerhalb des Projektes gegenwärtig noch immer in einer frühen Entwicklungsphase [D-G04] [GGH⁺05]. Zahlreiche, für die Abrechnung von Grid-Ressourcen wichtige Aspekte wie beispielsweise unterstützte Ressourcenarten als auch Abrechnungseinheiten werden in der Spezifikation nicht näher beschrieben. Dennoch stellen sowohl die verbrauchs- als auch service-level-basierte Abrechnung der angebotenen Ressourcen sowie der dem Abrechnungsmodell zugrunde liegende ökonomische Ansatz interessante Aspekte dar, die es bei der Umsetzung der Abrechnungslösung zu berücksichtigen gilt. Weiterhin weist der vorgestellte Ansatz bereits teilweise eine Konformität mit existierenden Grid-Standards wie beispielsweise OGSA in der Version 1.0 auf und ermöglicht somit eine Unterstützung OGSA-basierter Grid-Middleware, welche in der Spezifikation allerdings nicht näher beschrieben werden.

2.6 Grid Service Accounting Extensions (GSAX)

Die Grid Service Accounting Extensions (GSAX) stellen ein erweiterbares OGSA-Accounting- und Logging-Framework dar, welches aus dem „Extreme Blue“ Grid Accounting-Projekt hervorgegangen ist und durch die Global Grid Forum Resource Usage Service-Working Group (RUS-WG) [OGS06] entwickelt wurde [BHH⁺02].

Das GSAX Rahmenwerk, das eine Erweiterung zum OGSA-Standard [FKS⁺05] bildet, basiert auf einer modular aufgebauten Architektur, welche einen hohen Grad an Erweiterbarkeit aufweist, um eine Abrechnung auf vielen unterschiedlichen Anwendungsebenen zu ermöglichen. Zudem können von Seiten der Grid Service Accounting Extensions abrechnungsspezifische Informationen in verschiedenen Granularitäten, wie beispielsweise Echtzeit-Informationen bis hin zu Auftrags- bzw. Job-spezifischen Informationen, zur Verfügung gestellt werden [BHH⁺02]. Weiterhin ermöglicht die modular aufgebaute Architektur eine Integration verschiedener Dienstgüte-Parameter (Quality of Service, QoS) sowie SLAs in unterschiedlichen Ebenen des Rahmenwerks, wie beispielsweise innerhalb der Metering- oder der Accounting-Schicht. Wesentliches Ziel der Abrechnungskomponente stellt auf der einen Seite die Unabhängigkeit hinsichtlich der zu Grunde liegenden ökonomischen Modelle, als auch auf der anderen Seite die Anwendung dynamischer Strategien für die Ermittlung der Ressourcenpreise dar.

Wie aus Abbildung 2.8 ersichtlich wird, untergliedert sich die Abrechnungskomponente in zwei grundlegende Dienste, den *Account Management Service* sowie den *Accounting Service*. Wesentliche Aufgabe des Account Management Service ist es, anhand geeigneter Schnittstellen, übergeordneten Komponenten und Diensten sowohl Abrechnungsdatensätze als auch abrechnungsspezifische Informationen zur Verfügung zu stellen. Im Gegensatz dazu stellt der Accounting Service eine Schnittstelle zu Komponenten dar, welche auf niederen Ebenen des Frameworks angesiedelt sind [BHH⁺02].

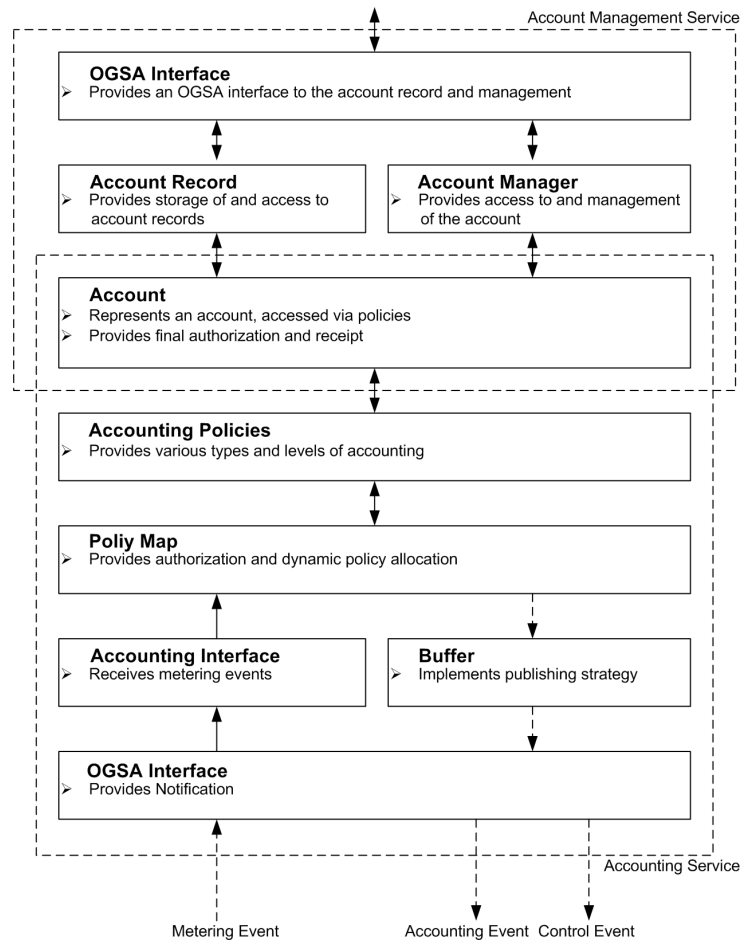


Abbildung 2.8: Architektur der GSAX Abrechnungskomponente, n. [BHH⁺02].

Grundlegende Aufgabenbereiche des Accounting Service umfassen unter anderem die Überwachung, als auch die Verarbeitung von Messereignissen. Sowohl der Account Management Service, als auch der Accounting Service besitzen eine eigene Instanz eines Kontos, auf welche mittels entsprechender Policies zugegriffen werden kann. Die Kontoinstanz beinhaltet unter anderem Informationen über den gegenwärtigen Kontostand sowie eine Liste mit autorisierten Benutzern, welchen ein Zugriff auf das Konto gewährt wird.

Bewertung des Ansatzes

Bei den Grid Accounting Services Extensions handelt es sich um einen weitgehend theoretisch ausgelegten Ansatz für ein Abrechnungssystem, welches zugleich eine Erweiterung zum OGSA-Standard darstellt. In das zugrunde liegende Framework sind bereits eine Vielzahl an interessanten Aspekten wie beispielsweise die Spezifikation von Ressourcenspezifischen Metriken wie z. B. CPU-Zeit, Netz- und Speicherauslastung sowie verschiedener anwendungsspezifischer Abrechnungseinheiten integriert. Ein weiteres, interessantes Merkmal der Spezifikation ist darüber hinaus die prinzipielle Unabhängigkeit gegenüber der zugrunde liegenden Grid-Middleware soweit diese eine Konformität zu der Open Grid Services Architecture aufweist. Ein vielversprechender Ansatz stellt zudem die Verwendung von Dienstgüteparametern und Service Level Agreements in unterschiedlichen Ebenen des Rahmenwerks dar [BHH⁺02].

2.7 Nimrod/G

Der *Nimrod/G Resource Broker* ist ein auf Grid-Technologien basierendes Ressourcen Management und Scheduling System, welches in Australien an der Monash University entwickelt wurde und die Durchführung von parametrisierten Studien über verteilte Rechenressourcen verwaltet und koordiniert [Bar04].

Die Systemarchitektur des Nimrod/G Resource Brokers, welcher einen integralen Bestandteil des Nimrod/G-Systems darstellt, besteht aus einer so genannten „*Task Farming Engine*“ (TFE), welche die Durchführung des gesamten Experiments durchführt, einem Meta-Scheduler sowie diversen Dispatchern, welche eine Integration in unterschiedliche Grid-Middleware-Systeme ermöglichen [ABG02]. Seitens der TFE werden zunächst benutzerspezifische Quality of Service-Anforderungen wie beispielsweise Zeit- oder Budgetrestriktionen entgegengenommen, welche im Anschluss daran dem Meta-Scheduler übermittelt werden. Dieser ist zum Einen für die Entdeckung und die Auswahl geeigneter Grid-Ressourcen und zum Anderen für die Übermittlung des Nutzerauftrags zur entsprechenden Ressource zuständig. Mittels so genannter „Grid-Informationsdienste“, wie z. B. dem *Grid Market Directory* werden verfügbare Ressourcen, welche den Anforderungen des Benutzers genügen, ermittelt [ABG01]. Anhand verschiedener, auf ökonomischen Ansätzen basierenden Algorithmen, wie z. B. der zentral regulierten Preisbildung, werden Verhandlungen mit den Ressourcenanbietern hinsichtlich der Kosten der Ressourcennutzung durchgeführt.

Nach Auswahl geeigneter Ressourcen wird ein so genannter „*Nimrod/G Agent*“ vom Dispatcher zur entsprechenden Ressource übermittelt. Der Nimrod/G Agent ist für den

Informations- und Datenaustausch, wie beispielsweise das Messen, Aufzeichnen und Übermitteln des Ressourcenverbrauchs, zwischen der Ressource und der TFE zuständig. Die anhand der Messungen gewonnenen Informationen können neben der Durchführung von Abrechnungen zudem für eine umfassende Leistungsbewertung der Ressource verwendet werden, um die ermittelten Ressourcenkennzahlen in zukünftige Scheduling-Strategien mit einfließen zu lassen.

Bewertung des Ansatzes

Der Nimrod/G Resource Broker stellt ein umfassendes Ressourcen Management System dar, welches eine Zuweisung geeigneter Grid-Ressourcen unter Berücksichtigung von benutzerspezifischen Quality of Service-Anforderungen vornimmt [Bar04].

Der Fokus des Scheduling-Systems ist vorwiegend in der Vermittlung und Zuordnung und nicht in der Abrechnung von Grid-Ressourcen angesiedelt. Eine Spezifikation von Abrechnungs- und Währungseinheiten, als auch eine Beschreibung von Nutzerkonten sind in dem zugrunde liegenden Ansatz nicht enthalten. Somit handelt es sich bei Nimrod/G nicht um ein eigenständiges Abrechnungssystem, vielmehr erweist sich eine Integration des Scheduling-Systems in existierende Accounting-Systeme wie z. B. GASA als sinnvoll [Bar04] [BB03].

2.8 SweGrid Accounting System (SGAS)

Das SweGrid Accounting System (SGAS) stellt ein bereits implementiertes Abrechnungsmodell für das Swedish National Grid (SweGrid) dar, welches sich aus sechs Clustern mit insgesamt 600 Rechnern zusammensetzt. Finanziert wird das SweGrid-Projekt durch die Swedish National Infrastructure for Computing (SNIC). Zum gegenwärtigen Zeitpunkt besteht das Swedish National Grid aus Rechenknoten, welche einen relativ hohen Grad an Homogenität aufweisen und auf gleichartigen Middleware-Installationen (Globus Toolkit) basieren, zukünftig soll ein Einsatz von heterogenen Hardware- und Softwarekomponenten innerhalb des SweGrid ermöglicht werden [FMW04].

Die grundlegende Architektur des Abrechnungssystems besteht im Wesentlichen aus einer Bankkomponente (so genannter „*Bank Service*“), dem *Job Account Reservation Manager* (JARM) sowie einem *Log and Usage Tracking Service* (LUTS) [EGMS04]. Abbildung 2.9 visualisiert die grundlegenden Komponenten des SweGrid Accounting Systems sowie deren Interaktionen [EGMS04].

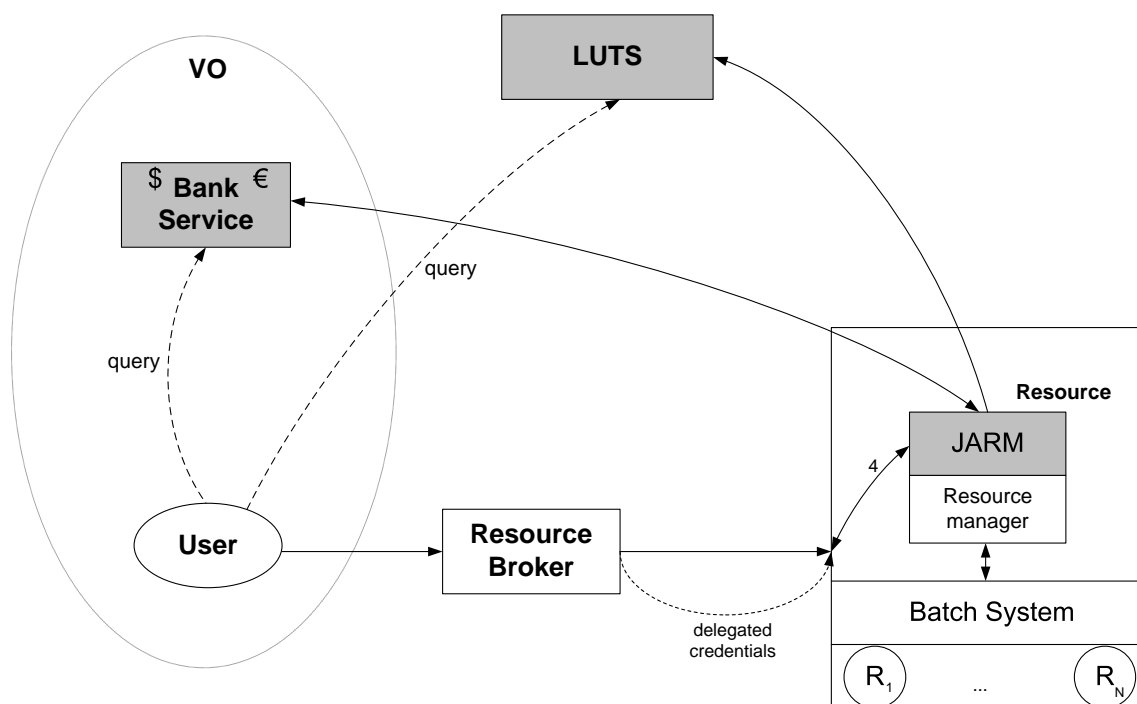


Abbildung 2.9: Interaktionen der Abrechnungskomponenten während der Jobsubmission, n. [EGMS04].

Innerhalb des SweGrid besitzt jede Virtuelle Organisation eine eigene assoziierte Bankkomponente, welche für die Verwaltung der Nutzerkonten zuständig ist. Somit liegt der Architektur ein dezentral ausgerichteter Ansatz zu Grunde. Der Bank Service basiert auf der Open Grid Services Infrastructure (OGSI) und ist neben einer Kontingentierung für die Abrechnung von Ressourcen der jeweiligen VO zuständig. Primäre Aufgabe der Bankkomponente stellt das Aufrechterhalten einer konsistenten Sicht auf die Ressourcen dar, welche von verschiedenen Institutionen auf Basis von Forschungsprojekten innerhalb des SweGrid in Anspruch genommen werden. Weiterhin beinhaltet die Bankkomponente eine Sicherung aller Bestände und Transaktionen auf einer externen, XML-basierten Datenbank [FMW04]. Weitere Aufgaben des Bankdienstes sind die Erstellung von Nutzerkonten und die Administration der Zugriffsrechte. Die Verwaltung der Konten, wie beispielsweise die Festlegung eines Kontoguthabens oder des Gültigkeitszeitraumes, obliegt dem jeweiligen Eigentümer des Benutzerkontos. Da die Bankkomponente selbst keine speziellen Ressourcentypen voraussetzt, kann diese in jede beliebige Grid-Umgebung integriert werden.

Jede Ressource innerhalb des SweGrid verfügt über einen so genannten „*Job Account Reservation Manager*“ (JARM), welcher auf dem Ressourcen- bzw. Job-Manager aufsetzt und theoretisch den einzigen Integrationspunkt für das Abrechnungssystem in jede belie-

bige Grid-Middleware darstellt [FMW04] [Gar03]. JARM bearbeitet eingehende Anfragen des Ressourcen-Brokers, ermittelt die Liquidität des Nutzers und führt eine (Vorab)-Reservierung der Projektkonten durch. Weiterhin nimmt JARM nach Abarbeitung des Auftrags die Abrechnung der in Anspruch genommenen Ressourcen bei der Bankkomponente vor. Somit stellt der Job Account Reservation Manager die einzige Komponente von SGAS dar, welche direkt mit der Auftragssubmission interagiert. Ferner wird mittels der so genannten „*Grid Resource Abstraction Integration*“ eine Integration verschiedener Scheduling Systeme und Workload Management Architekturen in die SGAS Architektur ermöglicht [San03]. Der Log and Usage Tracking Service (LUTS) gewährleistet einerseits Daten über die Ressourcennutzung zu protokollieren und andererseits, dass Benutzer diese Informationen auf eine konsistente Art und Weise abfragen können. Die Speicherung der Abrechnungsdaten erfolgt mittels der seitens des Global Grid Forum standardisierten Usage Records (UR) [FMW04]. Nach der Ausführung des Nutzerauftrags auf einer Grid-Ressource kann der Abrechnungsdatensatz in eine lokale Datenbank geschrieben, oder der Log and Usage Tracking Service dazu verwendet werden, die abrechnungsspezifischen Informationen auf einem entfernten Rechner mittels eines Dienstes zu veröffentlichen [FMW04].

Bewertung des Ansatzes

Das SweGrid Accounting System stellt ein wichtiges Abrechnungssystem dar, in welchem bereits eine Vielzahl an interessanten Aspekten und Ansätzen umgesetzt wurde. Die einheitliche, erweiterbare Architektur, welche auf der Grid-Middleware Globus Toolkit aufbaut, basiert auf verschiedenen, offenen standardisierten Grid-Protokollen wie beispielsweise Web Services oder Grid Services sowie der Open Grid Services Architecture (OGSA). Zudem wurden zahlreiche Sicherheitsaspekte innerhalb des Ansatzes berücksichtigt. Hierzu zählt beispielsweise die Authentifizierung mittels der Public Key Infrastructure (PKI) sowie mittels SSL-basierten Handshakes [FMW04]. Ein weiterer interessanter Aspekt ist die Verwendung einer Transaktionshistorie, welche die Möglichkeit der Verifikation durchgeführter Transaktionen bietet. Innerhalb des Ansatzes wird zudem jeder Virtuellen Organisation eine eigene Bankkomponente zugewiesen, was einen hohen Grad an Skalierbarkeit impliziert [EGMS04]. Darüber hinaus wurden in die zugrunde liegende Architektur zahlreiche Vorkehrungen für die Wiederherstellung und redundante Datenspeicherung integriert. Hierfür werden Informationen über den Zustand der Nutzerkonten sowie zahlreiche Transaktionslogs in einer XML-Datenbank abgelegt. Ähnlich der Grid Accounting Services Architecture (GASA) werden die Abrechnungsdatensätze in einem standardisierten Datenformat abgespeichert, was einen hohen Grad an Interoperabilität impliziert. Im Gegensatz zum Distributed Grid Accounting System werden von Seiten SGAS zahlreiche Funktionalitäten zur Administration der Benutzerkonten angeboten.

Nachteilig wirkt sich seitens der Architektur aus, dass gegenwärtig nur relativ homogene Grid-Ressourcen unterstützt werden und aufgrund dessen Mechanismen zur Bewertung von Ressourcen nicht vorhanden sind. Zudem wird zum gegenwärtigen Zeitpunkt lediglich Wallclock-Zeit als Abrechnungseinheit angeboten, die Unterstützung weiterer Abrechnungseinheiten ist geplant [EGM⁺03]. Zudem erfolgt innerhalb des SweGrid ausschließlich eine Abrechnung auf der Basis von Projektkonten [FMW04], hierbei sind ebenfalls weitere Entwicklungen denkbar.

2.9 Gesamtbewertung existierender Abrechnungssysteme

Der folgende Abschnitt sieht vor, wesentliche Defizite der gegenwärtig existierenden Abrechnungssysteme und -werkzeuge darzulegen. Um einen besseren Überblick über die Evaluationsergebnisse zu verschaffen, werden in Tabelle 2.1 die im Umfeld des Grid Computing betrachteten Abrechnungssysteme tabellarisch dargestellt und hinsichtlich verschiedener Kriterien resp. Anforderungen evaluiert.

In der Tabelle wird jeweils mit einem Haken ✓ angezeigt, ob eine Anforderung bzw. ein Kriterium grundsätzlich erfüllt wird. Eingeklammerte Haken bedeuten in diesem Zusammenhang, dass die jeweiligen Dokumente keine Informationen hierzu besitzen, dennoch aus anderen Quellen darauf geschlossen werden kann und somit die Anforderung bedingt erfüllt ist. Ferner bedeutet „n. a.“ dass in den existierenden Dokumenten als auch weiteren relevanten Arbeiten keine Angaben zu der jeweiligen Eigenschaft resp. Anforderung enthalten sind. Das Symbol – bedeutet schließlich, dass die gestellte Anforderung seitens des Abrechnungssystems nicht erfüllt ist.

Eines der wesentlichen Defizite bestehender Abrechnungslösungen für Grid-Umgebungen ist, dass diese im Hinblick auf die Abrechnung kein umfassendes Virtualisierungskonzept bezüglich der innerhalb von Virtuellen Organisationen angebotenen Diensten und Ressourcen beinhalten. Dies impliziert, dass seitens der existierenden Forschungsarbeiten keine Differenzierung zwischen Ressourcen und Diensten, die in realen Organisationen angeboten werden und virtuellen Diensten und Ressourcen, wie sie innerhalb von VOs offeriert werden, vorgenommen wird. Somit ist in keinem der begutachteten Abrechnungsansätze eine Definition von virtuellen Diensten und virtuellen Ressourcen enthalten, noch werden grundlegende Mechanismen zur Verfügung gestellt, die eine Abrechnung virtueller Dienste resp. virtueller Ressourcen gewährleisten.

Weiterhin lässt sich feststellen, dass der Fokus der betrachteten Abrechnungssysteme grundsätzlich auf der Abrechnung von physischen Grid-Ressourcen liegt, wie sie in realen

Anforderungen	Abrechnungssystem							
	AMIE	APEL	DGAS	GASA	GRASP	GSAX	Nimrod/G	SGAS
Interoperabilität und Portierbarkeit	(✓)	(✓)	(✓)	(✓)	n. a.	(✓)	✓	✓
Skalierbarkeit	(✓)	✓	(✓)	–	n. a.	✓	✓	✓
Integrationsfähigkeit	(✓)	(✓)	(✓)	n. a.	(✓)	✓	✓	✓
Organisations-übergreifende Abrechnung	–	✓	✓	✓	n. a.	✓	n. a.	✓
Flexibilität und Erweiterbarkeit	(✓)	✓	n. a.	✓	n. a.	✓	(✓)	✓
Unterstützung existierender Standards	–	–	–	(✓)	(✓)	(✓)	n. a.	✓
Virtualisierungskonzept	–	–	–	–	–	–	–	–
Abrechnung heterogener Ressourcen	–	(✓)	✓	✓	(✓)	n. a.	n. a.	–
Abrechnung virtueller Ressourcen	–	–	–	–	–	–	–	–
Abrechnung virtueller Dienste	–	–	–	–	–	–	–	–
Unterstützung von Multi-Provider-Szenarien	–	–	–	–	–	–	–	–
Kundenspezifische Darstellung d. Abrechnungsdaten	✓	✓	–	–	n. a.	n. a.	n. a.	–
Nutzertransparenz	(✓)	n. a.	n. a.	n. a.	n. a.	n. a.	n. a.	(✓)
Unterstützung einer hohen Dynamik	–	✓	(✓)	(✓)	n. a.	n. a.	✓	✓
Sicherheit	✓	n. a.	✓	✓	n. a.	✓	n. a.	✓
Einheitliche, generische Schnittstellen	–	–	–	–	n. a.	(✓)	✓	(✓)
Unterstützung verschiedener Maßeinheiten/Metriken	–	✓	✓	✓	n. a.	✓	n. a.	–
Präzision und Reichhaltigkeit	✓	✓	✓	✓	✓	✓	n. a.	✓
Unterstützung verschiedener Accounting-Policies	–	✓	✓	n. a.	n. a.	✓	n. a.	(✓)
Unterstützung von Hot-Accounting	–	–	✓	✓	n. a.	n. a.	✓	✓
Langzeitarchivierung	(✓)	✓	(✓)	(✓)	(✓)	(✓)	(✓)	(✓)
Analyse- und Prognosefunktionen	–	(✓)	–	–	–	–	–	–
Ausfallsicherheit und Fehlertoleranz	✓	n. a.	n. a.	(✓)	n. a.	n. a.	n. a.	✓
Administration und Verwaltung	✓	n. a.	(✓)	n. a.	n. a.	n. a.	n. a.	✓
Verifizierbarkeit	✓	n. a.	✓	✓	n. a.	n. a.	✓	✓

Tabelle 2.1: Übersicht über die Erfüllung der Anforderungen seitens der betrachteten Abrechnungssysteme und -werkzeuge, erweitert von [RG06].

Organisationen angeboten werden. Eine Abrechnung von virtuellen Grid-Ressourcen und komplexen Grid-Diensten, wie beispielsweise umfangreiche Berechnungs-, Visualisierungs- oder Informationsdienste, wird von den vorgestellten Arbeiten nur in einem sehr eingeschränkten Maße unterstützt. Einer Abrechnung von zusammengesetzten, komplexen Diensten und Ressourcen bleibt innerhalb der vorgestellten Abrechnungssysteme gänzlich unberücksichtigt. Insbesondere in Multi-Provider-Szenarien, in welchen mehrere reale als auch Virtuelle Organisationen gemeinsam komplexe, virtuelle Dienste und Ressourcen offerieren, stellt die Abrechnung von zusammengesetzten Diensten und Ressourcen einen wichtigen Aspekt dar, welcher durch das zugrunde liegende Virtualisierungskonzept gewährleistet werden muss. Ferner werden seitens der vorhandenen Abrechnungsansätze größtenteils statische Grid-Umgebungen betrachtet, in welchen die angebotenen Ressourcen einen hohen Grad an Homogenität aufweisen. Dynamische Grid-Umgebungen mit einem hohen Grad an Heterogenität im Hinblick auf die unterstützten Ressourcenarten,

Betriebssysteme sowie Grid-Middleware-Komponenten werden überwiegend nicht berücksichtigt. Weiterhin lässt sich beobachten, dass sämtliche untersuchten Abrechnungssysteme und -werkzeuge sehr hohe Abhängigkeiten von der eingesetzten Grid-Middleware sowie von den verwendeten Versionsständen der Middleware-Systeme aufweisen [RG06]. Zudem ist eine relativ hohe Abhängigkeit von zugrunde liegenden Werkzeugen wie Ressourcen-Broker und Scheduling als auch Workload Management Systemen zu erkennen.

Die Funktionalität des Hot-Accounting, welches eine Durchführung von Teilabrechnungen, beispielsweise basierend auf Push- resp. Pull-Mechanismen ermöglicht, wird von keinem der betrachteten Abrechnungsansätze unterstützt. Ferner lässt sich bei einem Teil der vorgestellten Forschungsarbeiten erkennen, dass seitens der Abrechnungssysteme nur wenige Abrechnungseinheiten wie beispielsweise ausschließlich CPU-Stunden oder Wallclock-Zeit unterstützt werden. Dies stellt insbesondere in Anbetracht einer Abrechnung von komplexen Grid-Diensten oder Grid-Ressourcen ebenfalls ein wesentliches Defizit bestehender Ansätze dar. Darüber hinaus lässt sich feststellen, dass bestehende Abrechnungslösungen mit Ausnahme von AMIE und APEL, keine Möglichkeiten der Visualisierung von Abrechnungsinformationen anbieten und somit eine kundenspezifische Darstellung der Abrechnungsdaten nicht unterstützt wird. Ein weiterer, wesentlicher Aspekt, welchem Rechnung getragen werden muss, ist die Verwendung einheitlicher, generischer Schnittstellen, um einen hohen Grad an Interoperabilität mit existierenden Abrechnungs- und Monitoring-Lösungen zu gewährleisten sowie die Verwendung von Analyse- und Prognosefunktionen hinsichtlich der Abrechnungsdaten. Im Hinblick auf derartige Aspekte weisen bestehende Abrechnungssysteme und -werkzeuge ebenfalls zahlreiche Defizite auf.

Kapitel 3

Arbeiten von Standardisierungsgremien

In den nachfolgenden Abschnitten werden die im Hinblick auf eine Abrechnung in Grid-Umgebungen bedeutenden Standardisierungsgremien wie das Global Grid Forum (GGF) und die Internet Engineering Task Force (IETF) in ihren Grundzügen vorgestellt. Anschließend werden die wesentlichen für die Abrechnung relevanten Arbeiten beschrieben und hinsichtlich ihrer Verwertbarkeit und Adaptionfähigkeit in Bezug auf eine Abrechnungslösung für dynamische Virtuelle Organisationen evaluiert.

3.1 Global Grid Forum (GGF)

Das Global Grid Forum [Glo06] ist eine Vereinigung von Unternehmen und Institutionen, die ein besonderes Interesse an der Nutzung und Entwicklung des Grid Computing haben. Innerhalb des GGF partizipieren gegenwärtig über 400 Organisationen aus mehr als 50 Ländern, schwerpunktmäßig aus den Vereinigten Staaten. In diesem Rahmen werden in mehr als zwei Dutzend Arbeitsgruppen, so genannten „*Working Groups*“ Grid-Standards und Grid-Spezifikationen für verschiedene Bereiche entwickelt. Das GGF unterstützt somit die Koordination, den Erfahrungsaustausch und die Zusammenarbeit von Grid-Entwicklern. „Global Grid Forum“ ist zugleich der Name der seit 2001 dreimal jährlich stattfindenden Symposien, in welchen sich die Teilnehmer mit dem Austausch von Best Practices und der Weiterentwicklung von Grid-relevanten Spezifikationen in Kooperation mit weltweit führenden Standardisierungsgremien wie beispielsweise der *Organization for the Advancement of Structured Information Standards* (OASIS) sowie der IETF befassen.

Neben der Entwicklung von Standardisierungsansätzen ist das grundlegende Ziel des GGF, Kosten und Entwicklungszeit von Grid-Anwendungen zu verringern und deren Sicherheit,

Anwendbarkeit und Kompatibilität zu fördern. Zu den wichtigsten veröffentlichten GGF-Standards zählen unter anderem die Open Grid Services Architecture (OGSA) [FKS⁺05] und die Open Grid Services Infrastructure (OGSI) [TCF⁺03]. Weiterhin ist das Global Grid Forum maßgeblich an der Entwicklung des Web Service Resource Framework (WSRF) [FMS05] [WSR06] beteiligt. Abbildung 3.1 stellt den organisatorischen Aufbau des Global Grid Forum graphisch dar.

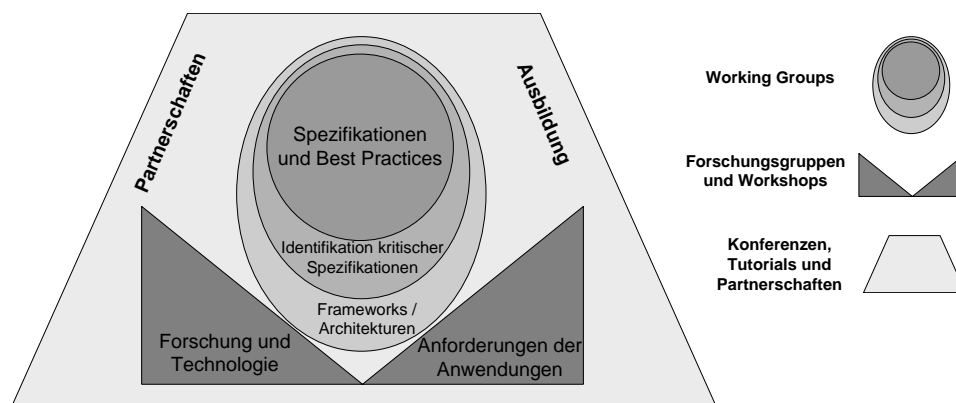


Abbildung 3.1: Organisatorischer Aufbau des Global Grid Forum, n. [Cat03].

Innerhalb des Global Grid Forum existieren vier Arbeitsgruppen, welche sich ausschließlich mit der Spezifikation von abrechnungsrelevanten Standards für das Grid-Umfeld befassen. Zu den wesentlichen in diesem Forschungsgebiet veröffentlichten Standardisierungsansätzen gehören das Grid Resource Allocation Agreement Protocol (GRAAP), die Usage Records (UR) und Resource Usage Services (RUS) sowie die Grid Economic Services Architecture (GESA), welche in den folgenden Abschnitten skizziert und im Hinblick auf die Entwicklung eines Abrechnungssystems für dynamische Virtuelle Organisationen bezüglich einer möglichen Verwendung bzw. Adaption beurteilt werden.

3.1.1 Grid Resource Allocation Agreement Protocol (GRAAP)

Gegenwärtig erfolgt seitens der Grid Resource Allocation Agreement Protocol-Working Group (GRAAP-WG) [GRA06a], welche innerhalb des Global Grid Forum im Forschungsbereich des Scheduling und Ressourcenmanagement angesiedelt ist, die Entwicklung der *Web Service Agreement Specification* (WS-Agreement) [ACD⁺05]. Hierbei handelt es sich um ein auf Service Level Agreements basierendes Web Service-Protokoll, welches für die Definition und Beschreibung von Vereinbarungen (so genannten „*Agreements*“) zwischen den Anbietern und den Nachfragern resp. Konsumenten von Ressourcen und Diensten

zuständig ist. Weiterhin wird seitens des Protokolls die Möglichkeit offeriert, dass Vereinbarungen zwischen Entitäten wie beispielsweise Ressourcen-Broker oder Scheduler, die im Auftrag der beteiligten Vertragsparteien handeln, geschlossen werden können.

Grundlegendes Ziel der vorgestellten Spezifikation stellt dabei die Standardisierung der verwendeten Terminologie und Konzepte, als auch eine formale Beschreibung der grundlegenden Struktur der zugrunde liegenden Vertragsvereinbarungen dar [ACD⁺05]. Darüber hinaus erfolgt in dem dazu veröffentlichten Dokument neben einer umfassenden Beschreibung grundlegender Anforderungen, die an das Protokoll gestellt werden, eine Spezifikation zahlreicher Operationen hinsichtlich der Erstellung und Aufhebung sowie der Überwachung von getroffenen Vereinbarungen. Eine Vereinbarung wird in diesem Kontext als dynamisch etablierte und dynamisch verwaltete Beziehung, d. h. als wechselseitiges Verhältnis zwischen den beteiligten Parteien definiert [ACD⁺05]. Wesentliches Ziel der aufgebauten Beziehung zwischen den Vertragsbeteiligten ist dabei das Anbieten eines Dienstes im Rahmen der getroffenen Vereinbarung. Neben funktionalen Eigenschaften, welche unter anderem eine Unterstützung für die Identifikation und Erzeugung von Diensten bereitstellen, werden innerhalb der Vereinbarung nicht-funktionale Eigenschaften des angebotenen Dienstes, wie beispielsweise das Verhalten, mögliche Antwortzeiten und die Verfügbarkeit der bereit gestellten Ressourcen spezifiziert. Abbildung 3.2 zeigt den grundlegenden Aufbau einer mittels WS-Agreement spezifizierten Vereinbarung. Die Vereinbarung besteht aus verschiedenen Abschnitten und wird mittels der standardisierten Auszeichnungssprache XML beschrieben [ACD⁺05].

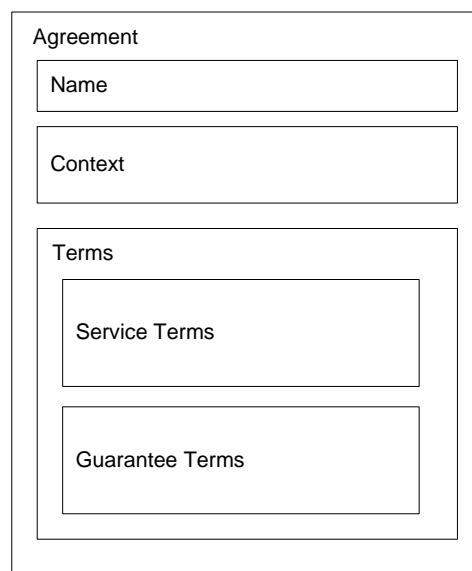


Abbildung 3.2: Aufbau eines WS-Agreements, n. [ACD⁺05].

In der spezifizierten Beschreibung der Vereinbarung dient das optionale Feld „Name“ dabei

der eindeutigen Identifikation der Vereinbarung. Im Abschnitt „Context“ werden sämtliche Meta-Daten des spezifizierten Agreements wie beispielsweise die Namen der Beteiligten sowie die Lebensdauer der Vereinbarung beschrieben. Die beiden darauf folgenden Abschnitte „Service Terms“ und „Guarantee Terms“ beinhalten verschiedene Bezeichnungen und Ausdrücke in Form von Attribut-basierten Beschreibungen, die eine Formalisierung der Vereinbarung ermöglichen.

Die in [ACD⁺05] vorgestellte Spezifikation des Web Service Agreement besteht im Wesentlichen aus drei Teilen. Neben einem Schema, welches der Definition von Vereinbarungen dient, wird in der Spezifikation ein weiteres Schema beschrieben, welches die Möglichkeit offeriert, generische Vorlagen für Vereinbarungen zu erstellen, um somit die Ermittlung von geeigneten Vertragsbeteiligten auf Basis von SLAs zu erleichtern. Bei der konkreten Beschreibung einer Vereinbarung können die vordefinierten Vorlagen sodann an die spezifischen Anforderungen angepasst werden. Darüber hinaus werden in der Spezifikation zahlreiche Operationen beschrieben, um den Lebenszyklus des Agreements, welcher die Erstellung und Aufhebung, als auch die Überwachung der Vereinbarung umfasst, verwalten zu können. Ferner werden in der WS-Agreement Spezifikation Aspekte wie Dienstgarantien und Dienstgüteparameter sowie die bei einer Vertragsverletzung gegebenenfalls zu initiiierenden Strafmaßnahmen berücksichtigt.

Technische Grundlage des WS-Agreement Protokolls sind dabei WS-ResourceProperties [GCF⁺03], welche von Seiten des OASIS Web Services Resource Framework (WSRF) Technical Committee spezifiziert wurden sowie WS-Addressing [Web06]. WS-Addressing beschreibt in diesem Kontext einen Mechanismus, welcher zur Adressierung von Web Services dient. Darüber hinaus kann das WS-Agreement Protokoll in Verbindung mit weiteren Web Service-Spezifikationen eingesetzt werden. Basierend auf einem konkreten Beispielszenario kann die grundlegende Struktur eines auf WS-Agreement basierenden, geschichteten Dienstmodells sowie eine Darstellung der benötigten Zugriffsschnittstellen beschrieben werden [ACD⁺05]. Dieses wird in Abbildung 3.3 schematisch dargestellt.

Das konzeptionelle Modell ist grundsätzlich in zwei Ebenen, das so genannte „*Service-Layer*“ und das „*Agreement-Layer*“ untergliedert. Letzteres offeriert dabei zahlreiche Web Service-basierte Dienstschnittstellen, welche der Repräsentation und Überwachung von Vereinbarungen verschiedener Anwendungsinstanzen dienen, die seitens des Service-Layer angeboten werden. Für eine umfassende Beschreibung der vorgestellten Architektur sowie der grundlegenden Funktionalität der darin enthaltenen Ebenen sei in diesem Zusammenhang auf [ACD⁺05] verwiesen.

In einem weiteren innerhalb der GRAAP-WG vorgestellten Dokument [CPPS06] werden im Hinblick auf das WS-Agreement Protokoll zahlreiche Anwendungsfälle und Szenarien beschrieben. Grundlegende Ausrichtung des Dokumentes ist die praxisnahe Spezifikation verschiedener Vorgaben hinsichtlich einer möglichen technischen Implementierung.

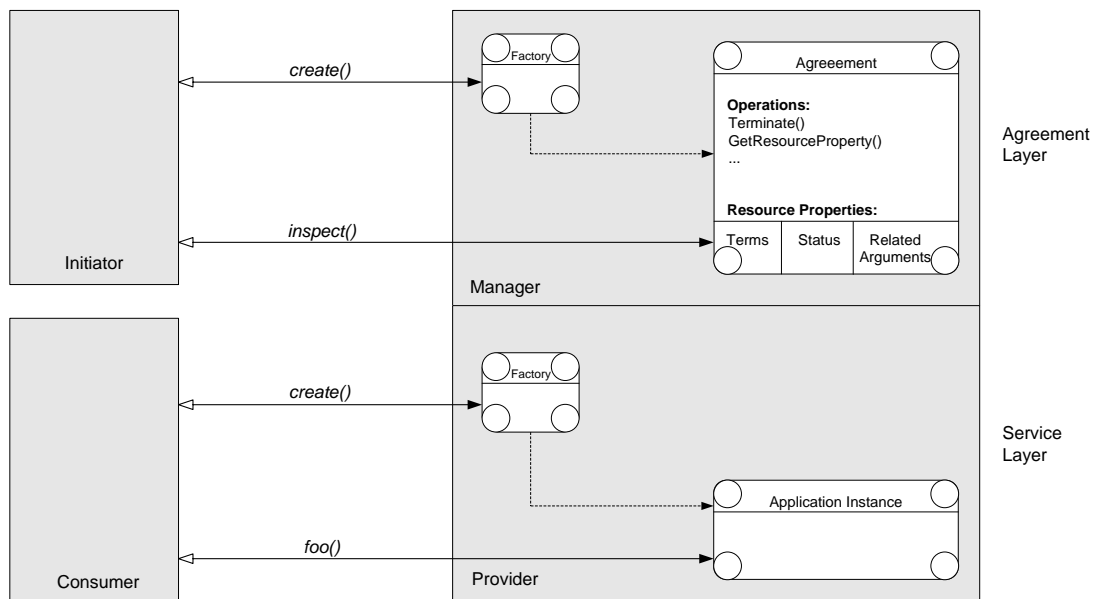


Abbildung 3.3: Konzeptionelles WS-Agreement Dienstmodell, n. [ACD⁺05].

Ein konkretes Szenario sieht dabei vor, das WS-Agreement Protokoll zwischen *Super-Scheduler-Diensten* (so genannten „Grid Level Scheduler“) und lokalen Schemulern einzusetzen, um somit eine Reservierung und Allokation von Grid-Ressourcen als Basis für die Offerierung von Grid-Diensten zu gewährleisten [MSZ02].

Bewertung

Die Arbeiten der GRAAP-WG bieten, basierend auf konkreten Anwendungsfällen und praxisnahen Anwendungsszenarien, eine Beschreibung zahlreicher interessanter Ansätze im Umfeld der Spezifikation von Vereinbarungen zwischen Anbietern und Nachfragern von Grid-Diensten und Grid-Ressourcen. Weiterhin unterstützt das entwickelte WS-Agreement Protokoll die Festlegung von Vertragsvereinbarungen zwischen beteiligten Entitäten wie beispielsweise Ressourcen-Broker oder Scheduler, die im Auftrag von Vertragsbeteiligten handeln können. Hierbei handelt es sich um einen vielversprechenden Ansatz, welchem insbesondere in kommerziellen Grid-Umgebungen, in denen zahlreiche Provider und Konsumenten in verschiedenen Domänen existieren, eine besondere Bedeutung zukommt.

Neben einer ausführlichen Beschreibung der zugrunde liegenden Terminologie und zahlreicher Anforderungen, welche an das zu entwickelnde Protokoll gerichtet sind, werden innerhalb der vorgestellten Spezifikation bereits einige grundlegende Begriffe bezüglich der Beschreibung von Vertragsvereinbarungen definiert. Seitens der Web Services Agreement

Spezifikation ist dennoch vorgesehen, die Definition einer Vielzahl von relevanten Begriffen in den anwendungsspezifischen Kontexten der jeweiligen Domäne vorzunehmen, beispielsweise hinsichtlich einer bestimmten Job Description Language (JDL). In der umfassenden Beschreibung des Protokolls werden darüber hinaus teilweise Grundzüge weiterer bedeutender Aspekte wie beispielsweise die bei einer Vertragsverletzung zu initiiierenden Maßnahmen berücksichtigt. Eine ausführliche Beschreibung derartiger Aspekte ist allerdings in der Spezifikation nicht zu finden und bedarf zahlreicher weiterer Betrachtungen.

Zusammenfassend lässt sich dennoch feststellen, dass die vorgestellten Arbeiten der GRAAP-WG interessante Ansätze vor dem Hintergrund einer Beschreibung und Festlegung von Service Level Agreements formulieren, die es im Zuge der Entwicklung eines Abrechnungssystems für dynamische Virtuelle Organisationen zwischen Kunde und Anbieter zu berücksichtigen gilt.

3.1.2 Usage Record (UR)

Seitens der Global Grid Forum Usage Record-Working Group (GGF UR-WG) [UR-06] wurde im Jahr 2003 eine Empfehlung für ein einheitliches Format zur Spezifikation von Abrechnungsdaten vorgestellt [MLMA⁺03]. Dieses kann als der gegenwärtige De-facto-Standard für die strukturierte Beschreibung von Nutzungs- und Abrechnungsinformationen im Umfeld des Grid Computing betrachtet werden.

Abrechnungsdatensätze, so genannte „Usage Records“ enthalten das von einem bestimmten Nutzer resp. Nutzergruppe in einem gegebenen Zeitraum in Anspruch genommene Ressourcen- bzw. Dienstonutzungsvolumen. Bei den von Seiten des GGF spezifizierten Usage Records handelt es sich um ein XML-basiertes Datenformat, dessen grundlegende Struktur in einem Schema spezifiziert wird, welches konform zur XML Schema Definition (XSD) ist [Jac03]. In den Abrechnungsdatensätzen sind gegenwärtig 26 Felder vorgesehen, welche als „Properties“ bezeichnet werden und der Spezifikation und Beschreibung von Eigenschaften einer Nutzung von Grid-Ressourcen dienen. Hierzu zählen unter anderem Nutzer-, Job-, als auch Ressourcen-spezifische Informationen. Die Verwendung der Felder „RecordIdentity“ und „Status“ ist für eine eindeutige Identifikation der Abrechnungsdatensätze erforderlich. Ferner sind in den Usage Records zahlreiche Felder für die Beschreibung von Abrechnungsinformationen für Grid-Ressourcen wie beispielsweise Berechnungselemente, Speicherressourcen sowie Netzkomponenten spezifiziert. Darüber hinaus ist von Seiten der Spezifikation ein Feld vorgesehen, welches der Beschreibung von Dienstgüteparametern dient, die mit der Ressourcennutzung assoziiert sind. Mittels des optionalen Feldes „Extension“ können zusätzlich spezifische Eigenschaften definiert werden. Hierbei kann es sich beispielsweise um Abrechnungseinheiten für spezielle, bis dato in der Spezifikation nicht berücksichtigte Ressourcentypen oder Dienstarten handeln.

Grundlegendes Ziel der seitens des GGF vorgestellten Spezifikation ist es auf der einen Seite ein hohes Maß an Spezifität aufzuweisen, um den Informationsaustausch von Abrechnungsdaten zwischen verschiedenen Einrichtungen zu erleichtern und auf der anderen Seite dennoch generisch zu sein, so dass die ermittelten Nutzungsinformationen neben einer Abrechnung für weitere Anwendungsbereiche wie beispielsweise die Erstellung von Nutzungsstatistiken, die Prioritätensteuerung und die Kapazitätsplanung verwendet werden können.

Tabelle 3.1 enthält eine Auflistung der für die Spezifikation von Abrechnungsdaten seitens der UR vorgesehenen Felder sowie eine kurze Erläuterung der jeweiligen Eigenschaften. Bei den mit dem Symbol * versehenen Feldern handelt es sich um optionale Eigenschaften zur Beschreibung der Abrechnungs- bzw. Nutzungsinformationen. Für eine umfassende Beschreibung der in den Usage Records enthaltenen Felder und möglicher Attribute sei auf die ausführliche Spezifikation der Usage Record-Working Group verwiesen [MLMA⁺03].

<i>Name der UR-Eigenschaft</i>	<i>Kategorie</i>	<i>Beschreibung</i>
RecordIdentity	Usage Record	Eindeutiger Identifikator des Usage Records.
GlobalJobId*	Jobdetails	Globaler Job-Identifikator, welcher seitens des Meta- oder Federation-Scheduler zugewiesen wurde.
LocalJobId*	Jobdetails	Lokaler Job-Identifikator, welcher von Seiten der Batchqueue zugewiesen wurde.
ProcessId*	Jobdetails	Prozess-ID (PID) des Nutzauftrags.
JobName*	Jobdetails	Name des Nutzauftrags oder der Anwendung.
Charge*	Jobdetails	Ermittelte Gesamtkosten des Nutzauftrags.
Status	Jobdetails	Status des Nutzauftrags, z. B. „abgebrochen“, „erfolgreich beendet“ usw.
StartTime*	Jobdetails	Startzeitpunkt des Nutzauftrags.
EndTime*	Jobdetails	Zeitpunkt, zu welchem Nutzauftrag beendet wurde.
Queue*	Jobdetails	Name der Warteschlange, von welcher der Nutzauftrag ausgeführt oder übermittelt wurde.
MachineName*	Nutzerdetails	Beschreibung des Rechners, auf welchem der Nutzauftrag ausgeführt wurde.
SubmitHost*	Nutzerdetails	Name des Host, von welchem der Nutzauftrag submittiert wurde.
LocalUserId*	Nutzerdetails	Lokale Identität des Nutzers, welcher mit Ressourcennutzung assoziiert ist.
GlobalUsername*	Nutzerdetails	Global eindeutige Identität des Nutzers, z. B. Distinguish Name (DN) des Nutzerzertifikates.
ProjectName*	Nutzerdetails	Name des Projektes, welches mit der Ressourcennutzung assoziiert ist.
Host*	Ressourcendetails	Name des Host, auf welchem der Nutzauftrag ausgeführt wurde.
WallDuration*	Ressourcendetails	Verstrichene Wallclock-Zeit während der Jobausführung.
CpuDuration*	Ressourcendetails	Genutzte CPU-Zeit, summiert über alle Prozesse des Jobs.
NodeCount*	Ressourcendetails	Anzahl der in Anspruch genommenen Knoten.
Processors*	Ressourcendetails	Anzahl der genutzten oder angeforderten Prozessoren.
Memory*	Ressourcendetails	Genutzter Hauptspeicher aller nebenläufigen Prozesse des Nutzauftrags.
Disk*	Ressourcendetails	Genutzter Plattenspeicher.
Swap*	Ressourcendetails	Informationen über genutzte Swap-Partitionen.
Network*	Ressourcendetails	Genutzte Netzkomponenten.
ServiceLevel*	Ressourcendetails	Dienstgüteparameter, welche mit der Ressourcennutzung assoziiert sind.
Extension*	Spezifische Eigen.	Optionales Feld zur Definition spezifischer Eigenschaften.

Tabelle 3.1: Vordefinierte Attribute der Usage Records.

Bewertung

Mit den Usage Records stellen die Arbeiten der UR-WG einen wertvollen Ansatz hinsichtlich der Beschreibung von Abrechnungs- und Nutzungsdaten von Grid-Ressourcen zur Verfügung. Der relativ weit verbreitete Ansatz wird gegenwärtig bereits von zahlreichen Abrechnungssystemen und -werkzeugen wie beispielsweise GridBank [BB03] und SGAS [EGM⁺03] eingesetzt und gewährleistet somit einen relativ hohen Grad an Interoperabilität hinsichtlich des Domänen-übergreifenden Austausches von Abrechnungsinformationen. Von Seiten der vorgestellten Spezifikation werden zum gegenwärtigen Zeitpunkt bereits zahlreiche Felder für die Ressourcen-spezifische Beschreibung der Abrechnungsdaten offeriert. Weiterhin ist seitens der Spezifikation grundsätzlich die Möglichkeit vorgesehen, zusätzliche Abrechnungs- und Nutzungsdaten, die in dem gegenwärtigen Standard nicht berücksichtigt werden, spezifizieren zu können. Die Usage Records weisen somit einen relativ hohen Grad an Erweiterbarkeit auf und lassen sich zu großen Teilen an die spezifischen Anforderungen, welche an die Beschreibung der Nutzung von einfachen Diensten und Ressourcen innerhalb dynamischer Virtueller Organisationen gerichtet werden, adaptieren.

Eine Möglichkeit der Beschreibung von Nutzungsinformationen hinsichtlich der Inanspruchnahme und Abrechnung von komplexen Grid-Diensten, welche insbesondere im Kontext von Multi-Provider-Szenarien innerhalb von Virtuellen Organisationen offeriert werden können, ist in der Spezifikation bis dato nicht enthalten. Ferner ist in den Usage Records zum gegenwärtigen Zeitpunkt lediglich ein Feld vorgesehen, welches der Beschreibung von ermittelten Kosten der angebotenen Dienste und Ressourcen dient. Insbesondere im Kontext der im Anschluss an die Abrechnung folgenden Rechnungsstellung stellt die Beschreibung von Informationen hinsichtlich der Preise und der ermittelten Kosten für die Nutzung von Ressourcen einen wesentlichen, zu betrachtenden Aspekt dar. In diesem Zusammenhang sind neben der Spezifikation von zusätzlichen Feldern, die der ausführlichen Beschreibung einer Nutzung von (virtuellen) komplexen Grid-Diensten dienen, unter Umständen zahlreiche Erweiterungen und Anpassungen an dem spezifizierten Format vorzunehmen.

3.1.3 Resource Usage Service (RUS)

Seitens der Resource Usage Service-Working Group (RUS-WG) [OGS06] wurde mit dem *Resource Usage Service* ein Standard für Grid-Dienste entwickelt, welcher die Möglichkeit bietet, Abrechnungsdatensätze wie die in Abschnitt 3.1.2 beschriebenen Usage Records in Datenspeichern wie beispielsweise Abrechnungsdatenbanken oder XML-Dateien abzuliegen. Darüber hinaus bieten die spezifizierten Dienste zahlreiche Funktionalitäten und Schnittstellen an, welche es ermöglichen, eine Extraktion abrechnungsspezifischer Informationen aus den Datensätzen vorzunehmen und diese weiteren Komponenten zur Verfügung

zu stellen. In der vorgestellten Spezifikation sind für den strukturierten Zugriff auf die Abrechnungsdatensätze zahlreiche Operationen definiert, um mittels vordefinierter *Service Data Elements* (SDE), sowohl Anfragen als auch Aktualisierungen der Usage Records vornehmen zu können. Bei SDEs handelt es sich um Attribut-ähnliche Konstrukte, welche anhand entsprechender Operationen Zustandsinformationen von Grid-Dienstinstanzen abfragen können.

Konzeptionell betrachtet stellt der Resource Usage Service einen einfachen Logging-Dienst dar, welcher von verschiedenen Entitäten wie beispielsweise Ressourcen, Ressourcen-Brokern sowie Nutzern resp. Nutzergruppen, die an abrechnungsspezifischen Informationen interessiert sind, verwendet werden kann. Beispiele für Komponenten, welche mittels eines geeigneten, einheitlichen Dienstes Zugriff auf die üblicherweise in Datenbanken abgelegten Abrechnungsinformationen benötigen, sind unter anderem Workload Manager, Visualisierungsdienste sowie Komponenten, welche beispielsweise für die Fakturierung von Ressourcen und Diensten zuständig sind.

Ferner werden in der vorgestellten Spezifikation zahlreiche Sicherheitsaspekte berücksichtigt. Neben einer Authentifizierung von Nutzern mittels MTLS (Mutual Transport Layer Security) sowie signierten Nachrichten unter der Verwendung von WS-Security [WSS06] finden von Seiten der Autorisierungskomponente Rollen-basierte Zugriffskontrollen ihren Einsatz [ANM05]. Für die Sicherstellung der Integrität der ausgetauschten Nachrichten werden verschiedene Auditing-Mechanismen verwendet.

Bewertung

Mit dem Resource Usage Service wird seitens des GGF ein Ansatz zur Beschreibung von Diensten für den einheitlichen Zugriff auf die abgelegten Abrechnungsdate vorgestellt, der für zahlreiche Anwendungsfälle ausreichend ist.

Dennoch stellt eine der wesentlichen Charakteristika des RUS die Abhängigkeit von den ebenfalls innerhalb des Global Grid Forum spezifizierten Usage Records dar. Somit ist eine Verwendung der innerhalb der RUS-WG spezifizierten Dienste mit weiteren Beschreibungsformaten für die Spezifikation von Nutzungs- und Abrechnungsinformationen aufgrund verschiedenartiger Standards und Schnittstellen in der Regel nicht gewährleistet. Ein weiteres Defizit der Spezifikation begründet sich in dem hohen Grad an Komplexität von Seiten der angebotenen Schnittstelle. Zahlreiche in der Spezifikation vorgesehene Operationen wie beispielsweise die Abfrage sowie die Speicherung von abrechnungsspezifischen Nutzungsinformationen sind auch ohne die Verwendung von RUS, beispielsweise mittels Operationen der einheitlichen OGS:GridService-Schnittstelle realisierbar. Da es sich bei der Empfehlung der Resource Usage Service-WG um einen ersten Entwurf handelt, sind

zum gegenwärtigen Zeitpunkt zahlreiche Aspekte in einem noch nicht ausreichenden Maße spezifiziert und bedürfen weiterer Entwicklungen.

Über die beschriebenen Einschränkungen hinaus werden insbesondere verschiedene, grundlegende Aspekte wie beispielsweise die Ermittlung von Informationen hinsichtlich der Bepreisung und Bezahlung der Nutzung von Grid-Ressourcen und Grid-Diensten in der Spezifikation nicht berücksichtigt.

3.1.4 Grid Economic Services Architecture (GESA)

Die innerhalb des Global Grid Forum von Seiten der GESA-Working Group [GES06] entwickelte *Grid Economic Services Architecture* hat zur Aufgabe, das in [FKS⁺05] spezifizierte OGSA-Modell um eine konkrete Bezahlschnittstelle zu erweitern und somit die Möglichkeit zu erschließen, dieses an eine ökonomische Umgebung anzubinden. Grundlegendes Ziel der Arbeitsgruppe stellt die Spezifikation von Protokollen und Dienstschnittstellen dar, um eine Integration verschiedener ökonomischer Marktmodelle für die Abrechnung von Seiten der Open Grid Services Architecture spezifizierten Grid Services zu ermöglichen. In dem innerhalb der GESA-WG veröffentlichten Dokument [New03b] werden zahlreiche Erweiterungen für Grid-Dienste definiert, welche die Errichtung eines Art „virtuellen Marktplatzes“ ermöglichen und somit, auf Grundlage von einheitlichen Grid-Protokollen, einen Handel von Ressourcen und Diensten in Grid-Umgebungen gewährleisten. Eine wesentliche Anforderung an die Architektur ist dabei, dass die zugrunde liegenden OGSA-Dienstschnittstellen nicht grundlegend verändert, sondern lediglich geringfügige Erweiterungen vorgenommen werden sollen.

Sämtliche, innerhalb des Global Grid Forum entwickelten abrechnungsrelevanten Standardisierungsansätze sind in dem in [New03a] vorgestellten, einheitlichen Modell wiederzufinden. Folglich basiert die Architektur unter anderem auf dem in Abschnitt 3.1.1 beschriebenen WS-Agreement Protokoll welches von Seiten der GRAAP-Working Group spezifiziert wurde, als auch auf den in Abschnitt 3.1.3 vorgestellten Resource Usage Services. Des Weiteren finden die innerhalb der UR-WG spezifizierten Usage Records (siehe Abschnitt 3.1.2), welche als Basis für die Bereitstellung der Abrechnungsdaten dienen, in der zugrunde liegenden Architektur ihren Einsatz. Abbildung 3.4 stellt den Aufbau der Grid Economic Services Architecture graphisch dar und visualisiert ferner die spezifizierten Schnittstellen zum OGSA-Modell. Wesentliche Komponenten der vorgestellten Architektur sind der *Chargeable Grid Service* (CGS), das *Grid Economic Service Interface* (GESI) sowie der *Grid Banking Service* (GBS).

Der von Seiten der Architektur spezifizierte Chargeable Grid Service beinhaltet spezielle Beahldienste, die vorhandene Grid-Dienste um zusätzliche *Service Data Elements* erweitern. Bei den spezifizierten SDEs handelt es sich um statische als auch dynamische

Meta-Daten, welche eine Assoziation der angebotenen Diensten mit spezifischen Informationen wie beispielsweise zugehörigen Kosten, unterstützten Bezahlmechanismen und Währungseinheiten sowie der konsumierten Grid-Ressourcen, ermöglichen. Dies impliziert, dass die seitens der Nutzer resp. Nutzergruppen in Anspruch genommenen Dienste und Ressourcen von dem Anbieter in Rechnung gestellt werden können und somit eine finanzielle Kompensation auf Basis monetärer Währungen gewährleistet wird.

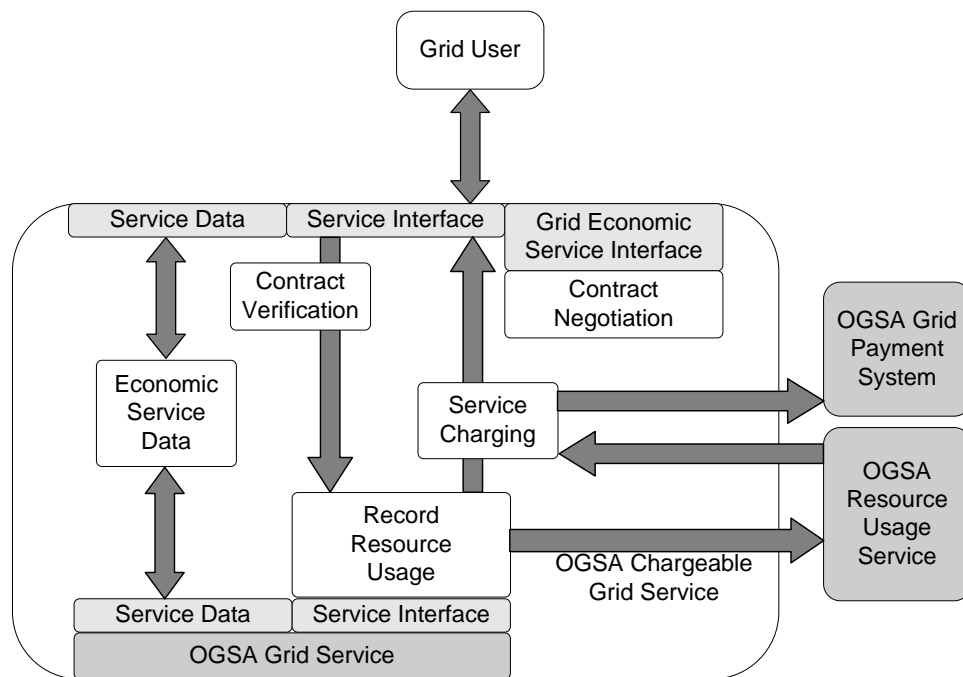


Abbildung 3.4: Komponenten und Schnittstellen der Grid Economic Services Architecture, n. [New03b].

Grundlegende Aufgabe des Grid Economic Service Interface ist die Bereitstellung so genannter „Port-Types“, mittels derer zahlreiche Funktionalitäten angeboten werden. In der Spezifikation der Web Service Description Language (WSDL) [WSD06] in der Version 1.1 ist ein Port-Type dabei als eine Menge von abstrakten Operationen und Nachrichten definiert, die mittels einer Web Service-Schnittstelle unterstützt werden. Ein grundlegender Port-Type des Grid Economic Service Interface stellt unter anderem der „*Factory Port*“ dar, welcher für die Instantiierung der im vorangegangenen Abschnitt dargestellten Chargeable Grid Services zuständig ist. Darüber hinaus werden seitens der spezifizierten Dienstschnittstelle weitere Port-Types zur Verfügung gestellt. Diese offerieren Funktionalitäten, um unter anderem eine Erfassung von Ressourcen- und Dienstkosten, als auch eine von Seiten des Nutzers initiierte Einwilligung bzw. Zustimmung im Hinblick auf den ermittelten Kostenvoranschlag, zu gewährleisten.

Der Grid Banking Service (GBS) offeriert zahlreiche Dienste, die eine Interaktion mit einer außerhalb der Architektur befindlichen Infrastruktur hinsichtlich der Bezahlung von Grid-Diensten und Grid-Ressourcen ermöglichen. Die Spezifikation einer konkreten Bezahlinfrastruktur ist dabei nicht Gegenstand der Betrachtungen. Darüber hinaus werden seitens des Bankdienstes verschiedene Interaktionen zwischen dem GBS und weiteren Entitäten der Architektur wie beispielsweise den CGS als auch Nutzern bzw. Nutzergruppen definiert. Details hinsichtlich einer konkreten Implementierung des Bankdienstes werden nicht näher spezifiziert. Dennoch besteht grundsätzlich die Möglichkeit, den Grid Bank Service mittels jeder beliebigen Infrastruktur, welcher reale Bankkonten zugrunde liegen, zu implementieren. Die seitens des GBS unterstützten Bezahlinfrastrukturen umfassen unter anderem Systeme wie Electronic Cash (EC), Kreditkarten sowie Prepaid-Konten [New03a].

Eng im Zusammenhang mit der vorgestellten Architektur stehen die im Jahr 2005 seitens der GESA-Working Group beschriebenen *Grid Economic Services*, welche eine Vielzahl ökonomischer Aspekte im Hinblick auf die Offerierung von Grid-Diensten und Grid-Ressourcen abdecken. Die in [MHL⁺05] beschriebenen Anwendungsfälle, welche unter anderem die Sichtweise von Ressourcenanbietern, Application Service Provider (ASP), Software Application Provider sowie Brokering Service Provider widerspiegeln, sind praxisnahe Beispiele für eine konkrete Umsetzung der entwickelten Architektur.

Bewertung

Von Seiten der GESA-WG wird mit der Grid Economic Services Architecture ein umfassendes Rahmenwerk angeboten, welches die Möglichkeit offeriert, das OGSA-Modell an eine konkrete Bezahlinfrastruktur anzubinden. Die vorgestellte Spezifikation enthält eine Beschreibung zahlreicher interessanter Dienste, die eine Bepreisung und Bezahlung von Ressourcen und Diensten in Grid-Umgebungen basierend auf monetären Währungseinheiten gewährleisten. Insbesondere im kommerziellen Grid-Umfeld stellen die beschriebenen Ansätze wertvolle Grundlagen dar, die es bei der Entwicklung eines Abrechnungssystems für dynamische Virtuelle Organisationen zu berücksichtigen gilt. Weiterhin werden seitens des vorgestellten Rahmenwerks zahlreiche Schnittstellen spezifiziert, die eine Anbindung der Architektur an reale Bezahlssysteme ermöglichen. In dem veröffentlichten Dokument werden sämtliche abrechnungsrelevante Arbeiten wie UR, RUS und GRAAP zu einer umfassenden Architektur vereint. Dies impliziert einen hohen Grad an Interoperabilität innerhalb der eingesetzten Lösung, als auch ein reibungsloses Zusammenspiel der einzelnen Bestandteile und Komponenten der beschriebenen Spezifikation.

Kritisch betrachtet stellt die ausschließliche Verwendung von GGF-basierten Abrechnungsstandards innerhalb der vorgestellten Architektur dennoch einen wesentlichen Nachteil dar. Diese Tatsache begründet sich darin, dass in der Regel nicht gewährleistet wird,

Standardisierungsansätze die außerhalb des GGF entwickelt wurden aufgrund fehlender Schnittstellen in die Spezifikation zu integrieren. Dies impliziert einen hohen Grad an Abhängigkeit hinsichtlich der möglichen zu verwendenden Ansätze. Zusammenfassend lässt sich feststellen, dass von Seiten der GESA-WG teilweise interessante Werkzeuge für den Handel von Grid-Ressourcen beschrieben werden, ohne dass eine genaue Betrachtung stattfindet, auf welche Art und Weise die Transaktionen realisiert werden können. Dieser Aspekt wird durch die Tatsache untermauert, dass die Entwicklung von geeigneten ökonomischen Marktmodellen nicht im Forschungsbereich der Arbeitsgruppe angesiedelt ist. Ein weiteres, wesentliches Defizit der Spezifikation welchem Rechnung getragen werden muss ist, dass seitens GESA zahlreiche Anpassungen hinsichtlich der Dienstschnittstellen erforderlich sind, um die Durchführung einer Bepreisung und Bezahlung einer Nutzung der angebotenen Grid-Dienste und Grid-Ressourcen zu ermöglichen. Darüber hinaus findet innerhalb des vorwiegend theoretisch ausgelegten Ansatzes keine Betrachtung relevanter Sicherheitsaspekte in Bezug auf die Abrechnung statt.

Aufgrund der fehlenden Beteiligung von Interessenten innerhalb der Forschungs-Community wurde die Arbeitsgruppe im Jahr 2005 vorübergehend geschlossen, so dass in nächster Zeit mit keinen weiteren Entwicklungen und Erweiterungen der Architektur zu rechnen sein wird [MN05].

3.2 Internet Engineering Task Force (IETF)

Die Internet Engineering Task Force [IET06] stellt neben der Internet Research Task Force (IRTF) eine von zwei Arbeitsgruppen des Internet Architecture Board (IAB) dar. Neben der Förderung von Internetstandards ist eine der wesentlichen Aufgaben der IETF die Definition und Weiterentwicklung von Technologiespezifikationen für das Internet. Die entwickelten Standards werden als Request for Comments (RFC) veröffentlicht. Ferner können die Ergebnisse z. B. als experimentelle Protokolle oder in Form von Informationstexten veröffentlicht werden. Innerhalb der IETF wird das Aufgabenfeld in mehrere Bereiche (Areas) untergliedert. Die verschiedenen Aspekte jedes Bereiches werden wiederum von lokalen Arbeitsgruppen (Working Groups) bearbeitet. Die IETF-Working Groups sind nach verschiedenen Bereichen untergliedert, die jeweiligen Bereichsleitern (Area Directors) unterstehen. Darüber hinaus werden die Mitglieder der Internet Engineering Steering Group (IESG) aus den Area Directors gebildet. Die Experten einer Working Group beschäftigen sich innerhalb eines begrenzten Zeitraums mit einem fest vorgegebenen Themenbereich. Das Thema bzw. die Problemstellung als auch die spezifischen Ziele werden in einer eigenen Working-Group-Charta festgehalten.

Die speziell für die Abrechnung bzw. das Abrechnungsmanagement relevanten Arbeiten der Working Groups sind *Authentication, Authorization and Accounting* (AAA) sowie

die erweiterte Sicht A4C (AAA zuzüglich *Auditing* und *Charging*). In den folgenden Abschnitten werden die Grundzüge der beiden Ansätze dargestellt und die wesentlichen Charakteristika skizziert. Darüber hinaus werden im Zuge einer Beschreibung der AAA-Protokolle die beiden Ansätze RADIUS und Diameter, welche sich als die gegenwärtigen De-facto-Standards etabliert haben, ausführlicher behandelt.

3.2.1 Arbeiten im Kontext der Authentication, Authorization and Accounting (AAA)-WG

Der Fokus der AAA-WG liegt in der Entwicklung von Anforderungen hinsichtlich der Authentifizierung, Autorisierung und Abrechnung in Bezug auf die Spezifikation von Netzzugangsprotokollen, als auch der Nutzung von Netzinfrastrukturen und -diensten. Grundlegendes Ziel der AAA-WG ist die Spezifikation eines neuen Dienstzugangsprotokolls (Diameter), das den heutigen Anforderungen bzgl. der Authentifizierung, Autorisierung und Abrechnung genügt. Im Zuge der durchgeführten Entwicklung werden bestehende AAA-Protokolle wie „Remote Authentication Dial In User Service“ (RADIUS) [RWRS00], als auch Anforderungen, welche unter anderem aus der Unterstützung von Mobilität resultieren, analysiert. Ein Beispiel hierfür stellt der Einsatz von Mobile IP [Per96] dar. Seitens der Arbeitsgruppe wurde eine generische AAA-Architektur vorgestellt, welche grundlegende Aspekte aus den Teilbereichen Authentifizierung, Autorisierung und Accounting berücksichtigt. Die in [dGG⁺00] spezifizierte Architektur umfasst eine auf abstrakter Ebene vorgenommene Beschreibung der benötigten Netzkomponenten und deren grundlegende Funktionalität.

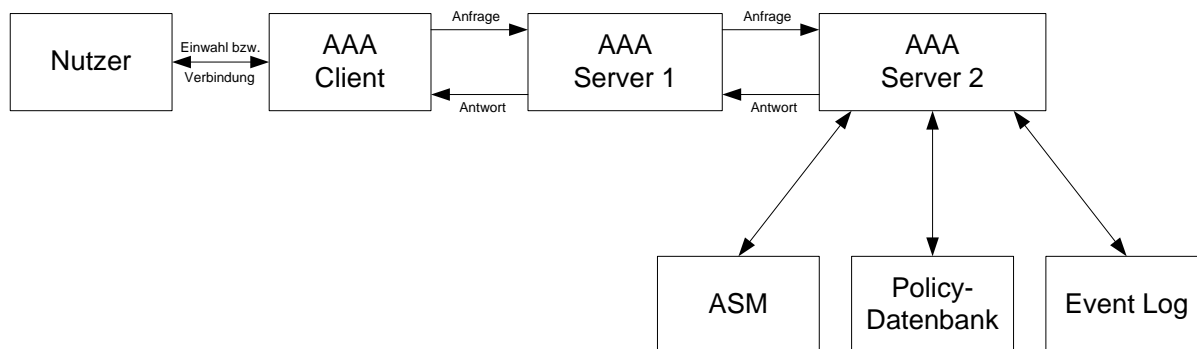


Abbildung 3.5: Komponenten der generischen AAA-Architektur, n. [Kno05].

Die AAA-Architektur, welche in Abbildung 3.5 dargestellt ist, besteht im Wesentlichen aus *AAA-Clients*, die im Auftrag des Nutzers agieren sowie verschiedenen *AAA-Servern*. Die Kommunikation zwischen den AAA-Clients und den AAA-Servern erfolgt mittels formatierter Anforderungs- und Antwortnachrichten. Die Vorformatierung der Anfrage ist

von entscheidender Bedeutung, da somit gewährleistet wird, dass die Nachricht sofort bearbeitet werden kann, ohne eine Aufbereitung und damit verbundene Interpretation der anwendungsspezifischen Anfrage durchführen zu müssen. Die seitens der Architektur vorgesehenen Clients können dabei integraler Bestandteil eines *Network Access Server* (NAS) oder weiterer Komponenten sein, die Netzdienste zur Verfügung stellen und diese kontrollieren. Die in der Spezifikation vorgestellten AAA-Server sind für sämtliche Authentifizierungs- und Autorisierungsentscheidungen sowie für die Erfassung von Abrechnungsinformationen zuständig. Generell besteht die Möglichkeit, dass der AAA-Server die Anforderungen des Clients an einen weiteren Server, welcher in der Literatur als generischer AAA-Server bezeichnet wird, weiterleitet [dGG⁺00].

Ferner ist in der angestrebten Realisierung die Verwendung eines *ASM* (Application Specific Module)-Server vorgesehen. Dieser verfügt über anwendungsspezifische Informationen und verwaltet und konfiguriert angebotene Netzressourcen derart, dass die Verfügbarkeit des angebotenen Dienstes aus dieser Sicht gewährleistet wird. Darüber hinaus wird der ASM-Server aufgrund seiner anwendungsspezifischen Informationen für die Unterstützung bei der Durchführung von Autorisierungsentscheidungen verwendet [Kno05]. Ein weiterer integraler Bestandteil der AAA-Architektur ist die so genannte „*Policy-Datenbank*“ in welcher Entscheidungsrichtlinien enthalten sind, auf die im Zuge der Durchführung von Autorisierungsprozessen von Seiten des AAA-Servers zugegriffen werden können. Vor dem Gesichtspunkt von Auditing-Mechanismen sieht die generische Architektur den Einsatz einer Logging-Komponente, dem so genannten „*Event Log*“ vor. Dieser stellt eine Art Datenspeicher dar, welcher von Seiten des generischen AAA-Servers für die Speicherung von Ereignissen verwendet wird, die innerhalb des AAA-Servers aufgetreten sind. Darüber hinaus kann der Event Log als Basis für Authentifizierungsentscheidungen herangezogen werden. Beispielweise kann der Fall existieren, nur dann einen Zugriff auf das System zu gewähren, falls bereits in der Vergangenheit ein ähnliches Ereignis festgehalten wurde.

Im Kontext des IP-basierten Netzzugriffes erfolgt sehr häufig die Kommunikation zwischen den AAA-Clients und dem AAA-Server mittels RADIUS [RWRS00] [Rig00]. Hierbei handelt es sich in erster Linie um einen Dienst, der vorrangig für die zentral durchgeführte Authentifizierung und Autorisierung für verschiedenartig realisierte Einwahlzugänge (z. B. per PPP [Sim94]) verwendet wird. RADIUS kann neben Diameter als einer der gegenwärtigen De-facto-Standards bei der zentralen Authentifizierung von Einwahlverbindungen über beispielsweise Modem, ISDN, VPN, Wireless LAN (802.1x) und DSL betrachtet werden. In [Rig00] ist eine Erweiterung des RADIUS-Protokolls hinsichtlich der Abrechnung spezifiziert, so dass ein RADIUS-Server als vollständiger AAA-Server bezüglich Einwahldiensten agieren kann. Die Erweiterung sieht insbesondere die Definition von neuen Protokollattributen vor, auf deren Basis Daten zu Abrechnungszwecken übertragen werden können. Dies impliziert, dass insbesondere im kommerziellen Umfeld mittels der vorgestellten Protokollerweiterung eine Abrechnung der genutzten Dienste gewährleistet wird. Um eine Integration von sowohl RADIUS-Clients als auch RADIUS-

Servern in ein vorhandenes Managementumfeld zu gewährleisten, wurden entsprechende Management Information Bases (MIB) definiert [Abo99] [ZA99]. Allerdings sehen diese in der Regel lediglich Managementvariablen für die Erstellung von Statistiken vor, mit denen beispielsweise der gegenwärtige Status eines RADIUS-Clients resp. eines Servers bezüglich Accounting-Anfragen, zugeordneten Clients/Servern, usw. abgefragt werden kann. Ein wesentlicher Nachteil von RADIUS begründet sich insbesondere in der schlechten Unterstützung von Inter-Domain-Anwendungen.

Während das Dienstzugangsprotokoll RADIUS einen weit verbreiteten Einsatz als AAA-Protokoll, insbesondere im Hinblick einer Unterstützung der Authentifizierung als auch der Bereitstellung von Abrechnungsfunktionalitäten erfahren hat, definiert Diameter [CLG⁺03] ein AAA-Protokoll der nächsten Generation. Diameter stellt hierbei eine auf dem RADIUS Protokoll basierende, abwärts-kompatible Entwicklung dar. Im Gegensatz zu RADIUS weist Diameter einen wesentlich höheren Grad an Erweiterbarkeit auf und bietet zuverlässigen Datentransfer, zahlreiche Mechanismen im Hinblick auf die Ausfallsicherheit, eine verbesserte Fehlerbehandlung sowie erweiterte Sicherheitsmechanismen. Das Protokoll Diameter besteht im Wesentlichen aus einem generischen Basisprotokoll und verschiedenen anwendungsspezifischen Erweiterungen beispielsweise für Mobile IP, Abrechnungen hinsichtlich der Dienstnutzung sowie Erweiterungen im Hinblick auf Sicherheitsfunktionen. Ferner ermöglicht Diameter die Spezifikation so genannter „*Attribut-Wert Paare*“ (Attribute Value Pairs, AVP), welche insbesondere vor dem Gesichtspunkt der Durchführung einer Abrechnung, die Definition von abrechnungsspezifischen Parametern erlauben [CLG⁺03].

Bewertung

Der Fokus der Betrachtungen der IETF AAA-WG liegt primär auf der Authentifizierung, Autorisierung sowie der Abrechnung einer Nutzung von Netzinfrastrukturen und Netzdiensten. Die vorgestellte, generische AAA-Architektur umfasst hierfür eine abstrakte Beschreibung der benötigten Netzkomponenten sowie deren grundlegende Funktionalität. In der vorgestellten Architektur, welche insbesondere eine wertvolle Grundlage für die Beschreibung einer Messung und Abrechnung von Datenverkehr offeriert, wurden zahlreiche interessante Aspekte wie beispielsweise die Authentifizierung und Autorisierung von Nutzern umgesetzt. Insbesondere in Grid-Umgebungen, in welchen Nachfragern kommerzielle Dienste und Ressourcen offeriert werden, stellen Aspekte der Authentifizierung und Autorisierung ein wesentliches Grundgerüst dar um sicherzustellen, dass Nutzer vor der Inanspruchnahme von Diensten resp. Ressourcen korrekt authentifiziert wurden und über ausreichende Privilegien verfügen, um einen Zugriff auf die angebotenen Dienste und Ressourcen zu erhalten. Insbesondere bei einer organisationsübergreifenden Abrechnung von Grid-Diensten und Grid-Ressourcen im Kontext von dynamischen Multi-Provider-

Szenarien erfahren die Authentifizierung und die Autorisierung eine besondere Bedeutung. Insofern gilt es derartige Aspekte, welche seitens der Arbeiten der IETF AAA-Working Group beschrieben wurden, bei der Entwicklung eines Abrechnungssystems im Grid-Umfeld zu berücksichtigen.

Trotz der Tatsache, dass die vorgestellten Arbeiten im Kontext der AAA-WG wertvolle Ansätze offerieren, sind diese aufgrund der innerhalb von Grid-Umgebungen vorherrschenden spezifischen Charakteristika nur teilweise universal anwendbar. Zudem sind die spezifizierten Ansätze nur bedingt auf die Abrechnung von Grid-Diensten und Grid-Ressourcen übertragbar. Dies begründet sich in der Tatsache, dass die Arbeiten der IETF vordergründig für Internetdienste und nicht für Grid-Anwendungen entwickelt wurden. An ein Abrechnungssystem im Kontext des Grid Computing werden zudem zahlreiche weitere Anforderungen wie beispielsweise die Unterstützung eines hohen Grades an Heterogenität hinsichtlich der eingesetzten und abzurechnenden Ressourcenarten, verwendeten Betriebssysteme sowie Middleware-Komponenten gestellt. Ferner gilt es im Grid-Umfeld neben den vielfältigen angebotenen komplexen Diensten darüber hinaus weitere Ressourcengruppen und -untergruppen wie beispielsweise Berechnungselemente, Speicherkomponenten, Softwarelizenzen und Programmbibliotheken sowie wissenschaftliche Geräte und Apparaturen zu betrachten. Hierbei handelt es sich zu großen Teilen um Ressourcentypen, die seitens der IETF AAA-WG aufgrund der Netz-spezifischen Ausrichtung der Arbeiten in den vorgestellten Spezifikationen nicht berücksichtigt werden.

3.2.2 Die erweiterte Authentication, Authorization, Accounting, Auditing und Charging (A4C)-Architektur

Unter den Schlagwörtern AAAA oder A4C - *Authentication, Authorization, Accounting, Auditing and Charging* wurde in jüngster Zeit intensiv an übergreifenden Standards für heterogene, überwiegend mobile Kommunikationssysteme resp. Kommunikationsnetze gearbeitet. Auf Basis der entwickelten Architekturen und vorgestellten Lösungen soll zukünftig neben einer Authentifizierung, Autorisierung und einer Abrechnung der Nutzung von Kommunikationsinfrastrukturen und -diensten ferner sowohl eine Leistungsverrechnung (so genanntes „*Charging*“) als auch eine Überwachung resp. Kontrolle der zugesicherten Systemparameter Domänen-übergreifend ermöglicht werden.

Im Zuge der Entwicklungen wurde innerhalb des EU-Projektes „Moby Dick“ (Mobility and Differentiated Services in a Future IP Network) [Mob06] die in Abschnitt 3.2.1 vorgestellte generische AAA-Architektur [dGG⁺00] um Auditing- als auch Charging-spezifische Aspekte erweitert. Die vorgestellte A4C-Architektur sieht dabei die Verwendung spezieller A4C-Server vor, welche in verschiedenen administrativen Domänen eingesetzt werden und einen organisationsübergreifenden Betrieb in Echtzeit gewährleisten [Wes05]. Hierzu zählt

unter anderem die Authentifizierung und Autorisierung von Roaming-Nutzern, als auch die Abrechnung von so genannten „Service Sessions“. Hierbei handelt es sich um Dienste, die sich über zahlreiche organisatorische Bereiche erstrecken können [MWS06]. Für den Nachrichtenaustausch zwischen den A4C-Servern wird ein Kommunikationsprotokoll verwendet, welches die Möglichkeit offeriert, Nutzerprofile als auch so genannte „Credentials“, d.h. Legitimationen in Form von Daten, auszutauschen. Um dies zu gewährleisten, wurde das in Abschnitt 3.2.1 vorgestellte AAA-Protokoll (Diameter) im Hinblick auf eine Unterstützung der A4C-Funktionalität erweitert [ESS⁺04]. Seitens der A4C-Server werden ferner zahlreiche generische Mechanismen für die Speicherung der Abrechnungsdatensätze angeboten, welche als Basis für die im Anschluss stattfindende Leistungsverrechnung dienen. Der Charging-Prozess sieht dabei vor, den Preis für die Inanspruchnahme eines Dienstes auf Basis der ermittelten Abrechnungsdaten sowie den festgelegten Service Level Agreements und zugrunde liegenden Tarifmodellen zu berechnen [ESS⁺04]. Die Bestimmung des Tarifes kann dabei beispielsweise basierend auf spezifizierten Dienstgüteparametern, genutztem Volumen, Sitzungsdauer als auch auf Basis der Domäne, innerhalb welcher der Dienst in Anspruch genommen wurde, erfolgen [ESS⁺04]. In Abbildung 3.6 (n. [MWS06]) wird der grundlegende Aufbau einer A4C-Architektur im Kontext des EU-Projektes „Akogrimo“ graphisch dargestellt.

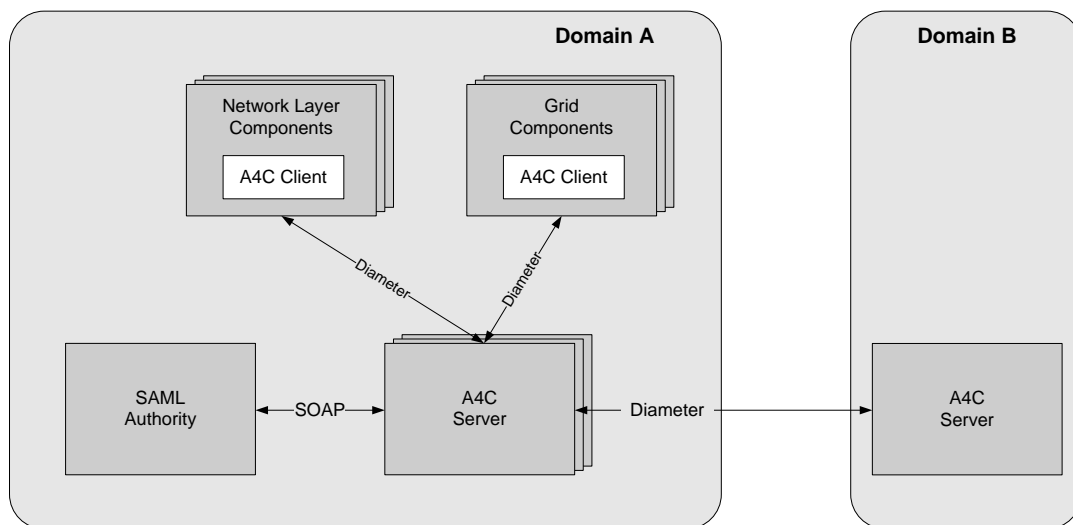


Abbildung 3.6: Komponenten der A4C-Architektur, n. [MWS06].

Die seitens der A4C-Architektur vorgesehene Auditing-Komponente dient vorwiegend einer Überwachung der durchgeführten Abrechnung. Darüber hinaus erfolgt von Seiten der Auditing-Komponente eine Überwachung der Inanspruchnahme von angebotenen Diensten hinsichtlich der Einhaltung von zuvor spezifizierten Dienstgüteparametern und seitens der Vertragsparteien festgelegten Service Level Agreements [ESS⁺04]. Hierfür werden von Seiten des A4C-Server Dienst- und Nutzer-spezifische Profile erstellt und ähnlich

den Abrechnungsdaten in einer Datenbank abgelegt.

Die entwickelte A4C-Architektur basiert auf der von Seiten der IETF spezifizierten generischen AAA-Architektur [dGG⁺00]. Grundlegende Komponenten der in [MWS06] beschriebenen Architektur sind neben dem *A4C-Server* verschiedene *A4C-Clients*. Wesentliche Aufgabe der A4C-Clients stellt in diesem Kontext die Gewährung von Zugriffen auf die A4C-Dienste für Netz- und Grid-Komponenten dar. Für jede Dienstkomponente, welche Funktionalitäten des A4C-Servers in Anspruch nehmen möchte, ist die Integration eines entsprechenden A4C-Clients erforderlich. Weiterhin berücksichtigt die vorgestellte Architektur die Unterstützung einer Single Sign On-Funktionalität, welche mittels der *SAML* (Security Assertion Markup Language) *Authority*-Komponente gewährleistet wird. Sämtliche Kommunikation zwischen den A4C-Clients und dem A4C-Server, als auch die Domain-übergreifende Kommunikation zwischen beteiligten A4C-Servern findet auf Basis des Dienstzugangsprotokolls Diameter statt. Für eine umfassende Beschreibung der A4C-Architektur, der darin enthaltenen Komponenten sowie der angebotenen Funktionalitäten sei auf die Ausführungen in der Literatur verwiesen [MWS06] [ESS⁺04].

Insbesondere im Umfeld von dynamischen Grid-Umgebungen ist es in der Regel notwendig, die Dienste des A4C-Servers in jeder Domäne, in der Grid-Dienste und Grid-Ressourcen offeriert werden, anzubieten. Um geeignete Abrechnungs- und Charging-Informationen zu erhalten, muss seitens des A4C-Servers die Nutzung sämtlicher im Zuge der Dienstleistung in Anspruch genommenen Grid-Ressource ermittelt und aufgezeichnet werden. Die Daten können dabei mittels Hardware- und Software-basierter Sensoren bzw. Kollektoren von Seiten der internen Netze des jeweiligen Diensteanbieters erfasst werden. Grundsätzlich besteht dabei die Möglichkeit, statt der Verwendung eines dedizierten A4C-Servers die Information hinsichtlich der Dienst- und Ressourcennutzung einer zentralen Entität zu übermitteln, welche den Mitgliedern einer dynamischen Virtuellen Organisation entsprechende A4C-Dienste offeriert [Gub06]. Eine exemplarische A4C-Monitoring-Architektur wird in Abbildung 3.7 dargestellt.

Bewertung

Mit der A4C-Architektur wurde die generische AAA-Architektur um wertvolle Funktionalitäten im Hinblick auf die Leistungsverrechnung (Charging), als auch der Überwachung und Kontrolle (Auditing) von Dienstgüteparametern und Service Level Agreements erweitert. In verschiedenen Forschungsprojekten wie beispielsweise dem EU-Projekt Akogrimo, werden zahlreiche interessante Anforderungen und Aspekte wie Single Sign On (SSO), Anonymität, flexible Abrechnungsverfahren sowie Auditing-Mechanismen betrachtet und in konkreten Implementierungen berücksichtigt [MWS06]. Darüber hinaus sind weitere Anwendungen der A4C-Architektur im Hinblick auf Grid-Infrastrukturen zu erkennen.

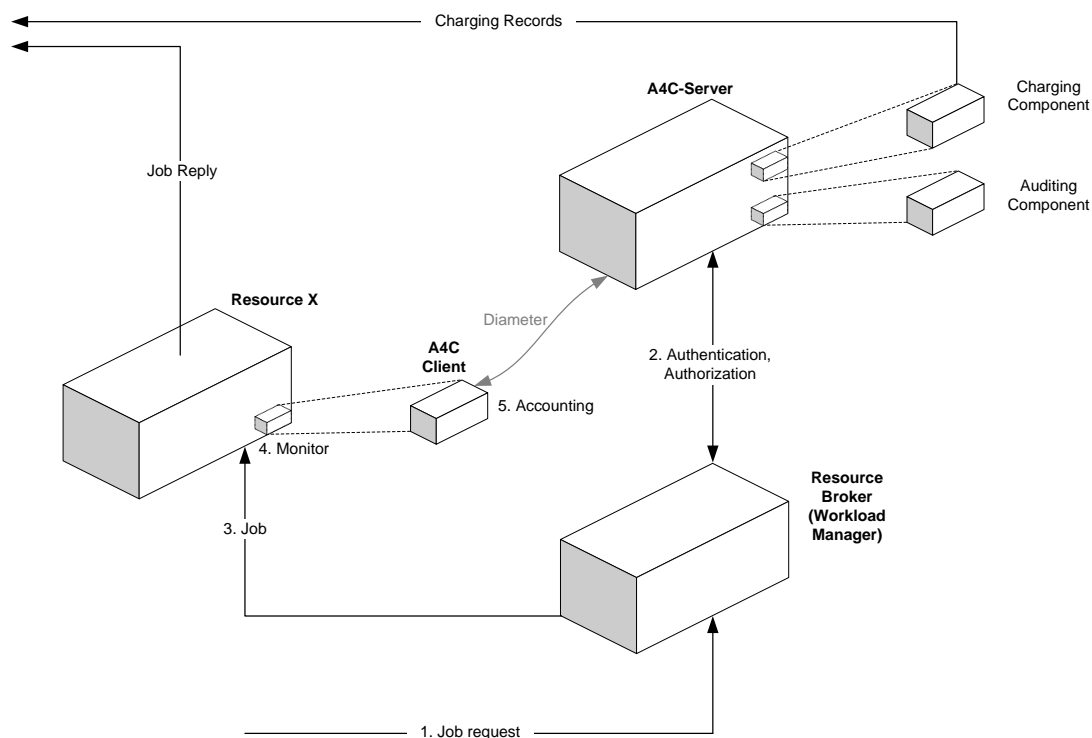


Abbildung 3.7: Exemplarische Architektur eines A4C Monitoring-Systems, n. [Gub06].

Sämtliche Aspekte wie beispielsweise die Authentifizierung und Autorisierung von Nutzern, die Verwaltung und Speicherung von abrechnungsspezifischen Informationen, als auch die Anbindung an Abrechnungssysteme, kann idealerweise vollständig an die A4C-Architektur delegiert werden.

Zusammenfassend lässt sich darstellen, dass mit der erweiterten A4C-Architektur neben Aspekten wie Authentifizierung und Autorisierung eine Vielzahl vielversprechender Ansätze im Hinblick auf die Organisations-übergreifende Abrechnung beschrieben und teilweise bereits prototypisch umgesetzt wurden. Ferner stellen die spezifizierten Ansätze hinsichtlich der Überwachung von Dienstgüteparametern und Service Level Agreements interessante Aspekte dar, welche es ebenfalls insbesondere im Umfeld von zukünftigen kommerziellen Grid-Umgebungen zu berücksichtigen gilt.

Abkürzungsverzeichnis

A4C	Authentication, Authorization, Accounting, Auditing and Charging
AAA	Authentication, Authorization and Accounting
AMIE	Account Management and Information Exchange
APEL	Accounting Processor for Event Logs
ASM	Application Specific Module
ASP	Application Service Provider
AVP	Attribute Value Pairs
DGAS	Distributed Grid Accounting System
DN	Distinguish Name
EDG	EU DataGrid
GASA	Grid Accounting Services Architecture
GESA	Grid Economic Services Architecture
GGF	Global Grid Forum
GIIS	Grid Index Information Server
GMA	Grid Monitoring Architecture
GOC	Grid Operations Centre
GRAAP	Grid Resource Allocation Agreement Protocol
GRASP	Grid-based Application Service Provision
GRB	Grid Resource Broker
GRM	Grid Resource Meter
GSAX	Grid Service Accounting Extensions
GSC	Grid Service Consumer
GSP	Grid Service Provider
GTS	Grid Trade Server

HLR	Home Location Register
IAB	Internet Architecture Board
IETF	Internet Engineering Task Force
JDL	Job Description Language
LCG	LHG Computing Grid
LSF	Load Sharing Facility
MIB	Management Information Base
NFS	Information Technology
OGSA	Open Grid Services Architecture
OGSI	Open Grid Services Infrastructure
PBS	Portable Batch System
PKI	Public Key Infrastructure
QoS	Quality of Service
RFC	Request for Comments
RUS	Resource Usage Service
SAML	Security Assertion Markup Language
SDE	Service Data Element
SGAS	SweGrid Accounting System
SLA	Service Level Agreement
SNIC	Swedish National Infrastructure for Computing
SSL	Secure Socket Layer
SSO	Single Sign On
SU	Service Unit
TFE	Task Farming Engine
TGDC	TeraGrid Central Database
UR	Usage Record
VHE	Virtual Hosting Environment
VO	Virtuelle Organisation
WSLA	Web Service Level Agreements
WSRF	Web Service Resource Framework
XSD	XML Schema Definition

Abbildungsverzeichnis

2.1	Architektur des Abrechnungssystems AMIE.	5
2.2	Schematische Darstellung des Zugriffs auf die APEL Datenbank mittels Plug-Ins.	7
2.3	Grundlegende Architektur des Abrechnungswerkzeuges APEL.	8
2.4	Architektur des Distributed Grid Accounting System.	10
2.5	Interaktion von GridBank mit verschiedenen Grid-Komponenten.	12
2.6	Aufbau der Grid Resource Meter-Komponente.	13
2.7	Statische Sicht auf das GRASP Abrechnungssystem.	15
2.8	Architektur der GSAX Abrechnungskomponente.	17
2.9	Interaktionen der Abrechnungskomponenten während der Jobsubmission.	20
3.1	Organisatorischer Aufbau des Global Grid Forum.	26
3.2	Aufbau eines WS-Agreements.	27
3.3	Konzeptionelles WS-Agreement Dienstmodell.	29
3.4	Komponenten und Schnittstellen der Grid Economic Services Architecture	35
3.5	Komponenten der generischen AAA-Architektur.	38
3.6	Komponenten der A4C-Architektur.	42
3.7	Exemplarische Architektur eines A4C Monitoring-Systems.	44

Literaturverzeichnis

- [ABC⁺03] G. Avellino, S. Beco, B. Cantalupo, A. Maraschini, F. Pacini, A. Terracina, Barale S., A. Guarise and A. Werbrouck. *The EU DataGrid Workload Management System: towards the second major release*, 2003. URL: http://www.gridpp.ac.uk/papers/chep03_MOAT007.PDF.
- [ABG01] D. Abramson, R. Buyya and J. Giddy. *A Case for Economy Grid Architecture for Service Oriented Grid Computing*, 2001. URL: <http://www.csse.monash.edu.au/~davida/papers/ecogrid.pdf>.
- [ABG02] D. Abramson, R. Buyya and J. Giddy. *A Computational Economy for Grid Computing and its Implementation in the Nimrod-G Resource Broker*, 2002. URL: <http://arxiv.org/ftp/cs/papers/0111/0111048.pdf>.
- [ABG⁺06] C. Anglano, S. Barale, L. Gaido, A. Guarise, G. Patania, R. Piro, F. Rosso and A. Werbrouck. *The Distributed Grid Accounting System (DGAS) Webpage*, 2006. URL: <http://www.to.infn.it/grid/accounting/main.html>.
- [Abo99] B. Aboba. *RADIUS Accounting Client MIB*. Request for Comments (RFC), Internet Engineering Task Force (IETF), 1999. URL: <http://www.ietf.org/rfc/rfc2620.txt>.
- [Acc03] *An Accounting System for the DataGrid Project – Preliminary Proposal – v.3.1*, 2003. URL: http://www.to.infn.it/grid/accounting/techrep/DataGrid-01-TED-0115-3_0.pdf.
- [ACD⁺05] A. Andrieux, K. Czajkowski, A. Dan, K. Keahey, H. Ludwig, T. Nakata, J. Pruyne, J. Rofrano, S. Tuecke and M. Xu. *Web Services Agreement Specification (WS-Agreement)*. Grid Resource Allocation Agreement Protocol Working Group, Global Grid Forum, 2005. URL: https://forge.gridforum.org/sf/docman/do/downloadDocument/projects.graap-wg/docman.root.current_drafts/doc6090.

- [ANM05] J. Ainsworth, S. Newhouse and J. MacLaren. *Resource Usage Service (RUS) based on WS-I Basic Profile 1.0*. Usage Record Working Group, Global Grid Forum, 2005. URL: http://www.gridforum.org/Public_Comment_Docs/Documents/Oct-2005/draft-ggf-wsi-rus-15.pdf.
- [Bar04] A. Barmouta. *Authorisation and Accounting Services for the World Wide Grid*. Mastersthesis, University of Western Australia, 2004. URL: <http://www.gridbus.org/students/AlexMastersThesis.pdf>.
- [BB03] A. Barmouta and R. Buyya. *GridBank: A Grid Accounting Services Architecture (GASA) for Distributed Systems Sharing and Integration*, 2003. URL: <http://www.gridbus.org/papers/gridbank.pdf>.
- [BCC⁺05] R. Byrom, R. Cordenonsi, L. Cornwall, M. Craig, D. Abdeslem, A. Ducan, S. Fisher, J. Gordon, S. Hicks, D. Kant, J. Leake, R. Middleton, M. Thorpe, J. Walk and A. Wilson. *APEL: An implementation of Grid accounting using R-GMA*, 2005. URL: <http://www.gridpp.ac.uk/abstracts/allhands2005/apel.pdf>.
- [BHH⁺02] A. Beardsmore, K. Hartley, S. Hawkins, S. Laws, J. Magowan and A. Twigg. *GSAX Grid Service Accounting Extensions*, 2002. URL: <http://www.doc.ic.ac.uk/~sjn5/GGF/ggf-rus-gsax-01.pdf>.
- [BWK05] R. Byrom, J. Walk and D. Kant. *User Guide for Apel – Accounting using PBS Event Logging*, 2005. URL: <http://hepunix.rl.ac.uk/edg/wp3/documentation/apel/apel-user-guide.pdf>.
- [Cat03] C. Catlett. *Global Grid Forum: Standards for Grid Computing*. Argonne National Laboratory and University of Chicago, 2003. URL: <http://www-unix.mcs.anl.gov/fl/events/agr2003/proceedings/catlett-ggf.pdf>.
- [CBC⁺05] C. Catlett, R. Bair, T. Cockerill, C. Hempel, J.P. Navarro, L. Liming, T. Rimovsky, R. Roskies, S. Sanielevici, J.R. Scott, M. Showerman, R. Stock and N. Wilkins-Diehr. *Integrating TeraGrid Computational Resources and Facilities – An Overview*, 2005. URL: http://www.teragrid.org/basics/docs/TeraGrid_Primer.pdf.
- [CLG⁺03] P. Calhoun, J. Loughney, E. Guttman, G. Zorn and J. Arkko. *Diameter Base Protocol*. Request for Comments (RFC), Internet Engineering Task Force (IETF), 2003. URL: <http://www.ietf.org/rfc/rfc3588.txt>.

- [CPPS06] K. Czajkowski, S. Pickles, J. Pruyne and V. Sander. *Usage Scenarios for a Grid Resource Allocation Agreement Protocol*. Grid Resource Allocation Agreement Protocol Working Group, Global Grid Forum, 2006. URL: http://www.gridforum.org/Public_Comment_Docs/Documents/Apr-2006/draft-ggf-graap-usagescenarios-03.pdf.
- [D-G04] *D-Grid AK2 Middleware und Services Bestandsaufnahme*, 2004. URL: http://www-grid.desy.de/d-grid/ak2/DGrid_AK2_Anhang_Bestandsaufnahme.pdf.
- [D-G06] *D-Grid Initiative Webpage*, 2006. URL: <http://www.d-grid.de/>.
- [DGA06] *The DataGrid Project Webpage*, 2006. URL: <http://eu-datagrid.web.cern.ch/eu-datagrid/>.
- [dGG+00] C. de Laat, G. Gross, L. Gommans, J. Vollbrecht and D. Spence. *Generic AAA Architecture*. Request for Comments (RFC), Internet Engineering Task Force (IETF), 2000. URL: <http://www.ietf.org/rfc/rfc2903.txt>.
- [DRY+03] T. Dimitrakos, D. Randal, F. Yuan, M. Gaeta, G. Laria, P. Ritrovato, B. Serhan, S. Wesner and K. Wulf. *An Emerging Architecture Enabling Grid Based Application Service Provision*, 2003. URL: <http://eu-grasp.net/english/dissemination/articles/EDOCconference03.pdf>.
- [e-I05] e-Infrastructure Reflection Group. *e-Infrastructures Reflection Group White Paper*, 2005. URL: <http://www.e-irg.org/whitepapers/2005-Luxembourg-pre-accepted.pdf>.
- [EGE05a] EGEE. *Review of Accounting and Monitoring Software Deliverable: D1*, 2005. URL: <https://www.egee.cesga.es/documents/D1/EGEE-D1-Review-Accounting-Monitoring-v0.8.pdf>.
- [EGE05b] *EGEE Users Guide*, 2005. URL: <http://www.to.infn.it/grid/accounting/techrep/EGEE-DGAS-HLR-Guide-20050713.pdf>.
- [EGM+03] E. Elmroth, P. Gardfjäll, O. Mulmo, Å. Sandgren and T. Sandholm. *A Coordinated Accounting Solution for SweGrid*, 2003. URL: <http://www.pdc.kth.se/grid/sgas/docs/SGAS-0.1.3.pdf>.
- [EGMS04] E. Elmroth, P. Gardfjäll, O. Mulmo and T. Sandholm. *An OGSA-Based Bank Service for Grid Accounting Systems*, 2004. URL: <http://delivery.acm.org/10.1145/1040000/1035207/p279-sandholm.pdf?key1=1035207&key2=4344981411&coll=GUIDE&d1=GUIDE&CFID=66913941&CFTOKEN=43854057>.

- [ESS⁺04] H. Einsiedler, A. Sarma, B. Stiller, J. Jähnert, D. Pal Singh, V. Marques, H.-Y. Lach, M. Wetterwald, P. Pacyna, P. Reichl and P. Mishra. *Mobility and Differentiated Services in a Future IP Network*. Final Project Report, IST-2000-25394 Project Moby Dick, 2004. URL: <http://www.ist-mobydick.org/deliverables/FinalReport.pdf>.
- [FKS⁺05] I. Foster, H. Kishimoto, A. Savva, D. Berry, D. Abdeslem, A. Grimshaw, B. Horn, F. Maciel, F. Siebenlist, S. Ravi, J. Treadwel and J. von Reich. *The Open Grid Services Architecture, Version 1.0*, 2005. URL: <http://www.ggf.org/documents/Drafts/draft-ggf-ogsa-spec.pdf>.
- [FMS05] I. Foster, T. Maguire and D. Snelling. *OGSA WSRF Basic Profile 1.0*. Usage Record Working Group, Global Grid Forum, 2005. URL: http://www.gridforum.org/Public_Comment_Docs/Documents/Oct-2005/draft-ggf-ogsa-wsrf-basic-profile-v43.pdf.
- [FMW04] C. Frogner, T. Mandt and S. Wethal. *Cluster and grid computing: Accounting and banking systems*, 2004. URL: http://hovedprosjekter.hig.no/v2004/data/gruppe05/files/cgc2004_report.pdf.
- [Gar03] P. Gardfjäll. *SweGrid Accounting System Bank*, 2003. URL: <http://www.sgas.se/docs/SGAS-BANK-DD-0.1.pdf>.
- [GCF⁺03] S. Graham, K. Czajkowski, D.F. Ferguson, I. Foster, J. Frey, F. Leymann, T. Maguire, N. Nagaratnam, M. Nally, T. Storey, I. Sedukhin, D. Snelling, S. Tuecke, W. Vambenepe and S. Weerawarana. *Web Services Resource Properties (WS-ResourceProperties)*, 2003. URL: <http://www-128.ibm.com/developerworks/library/ws-resource/ws-resourceproperties.pdf>.
- [GES06] *Grid Economic Services Architecture Working Group Webpage*, 2006. URL: <https://forge.gridforum.org/projects/gesa-wg>.
- [GGH⁺05] F. Gagliardi, P. Graham, M. Heikkurinen, J. Nabrzyski, A. Oleksiak, M. Parsons, H. Stockinger, K. Stockinger, M. Stroinski and Weglarz J. *GRIDSTART Project: IST Grid Projects Inventory and Roadmap*, 2005. URL: <http://www.gridstart.org/download/GRIDSTART-IR-D2.2.1.4-V1.6.doc>.
- [GK06] M. Golling and M. Kretzschmar. *Entwicklung einer Architektur für das Accounting in dynamischen Virtuellen Organisationen*. Diplomthesis, University of Federal Armed Forces Munich, Germany, 2006.
- [Glo06] *Global Grid Forum (GGF)*, 2006. URL: <http://www.gridforum.org/index.php>.

- [Gor05] J. Gordon. *Accounting, 'the last A'*, 2005. URL: <http://www.e-irg.org/meetings/2005-NL/johngordon-accounting.pdf>.
- [GR06] M. Göhner and C.-P. Rückemann. *Accounting-Ansätze im Bereich des Grid-Computing*. 10. April 2006, D-Grid, Fachgebiete Monitoring, Accounting und Billing im D-Grid-Integrationsprojekt, 2006. URL: http://dgi.d-grid.de/index.php?id=118&filename=mab_accounting_ansaetze.pdf&dir=FG2/koordination_mab&task=download&mountpoint=2.
- [GRA05] *GRASP Tutorial, First Presentation: SLA Document, Manageability and Accounting Subsystem*, 2005. URL: <http://eu-grasp.net/english/SalernoMeeting/GRASP%20Tutorial%20Final%20-%20Verdino.ppt#1>.
- [GRA06a] *Grid Resource Allocation Agreement Protocol Working Group Webpage*, 2006. URL: <https://forge.gridforum.org/projects/graap-wg>.
- [GRA06b] *The Grid-based Application Service Provision (GRASP) Project Webpage*, 2006. URL: <http://eu-grasp.net/english/default.htm>.
- [Gub06] F. Gubler. *Accountable Units for Grid Services in Mobile Dynamic Virtual Organisations*. Diplomathesis, University of Zurich, Switzerland, 2006. URL: https://www.ifi.unizh.ch/fileadmin/site/teaching/Diplomarbeiten/Abgeschlossene_Diplomarbeiten/Jahrgang_2006/Gubler_Fabian.pdf.
- [IET06] *The Internet Engineering Task Force (IETF) Webpage*, 2006. URL: <http://www.ietf.org/>.
- [Jac03] S. Jackson. *Usage Record – XML Format*. Usage Record Working Group, Global Grid Forum, 2003. URL: http://www.gridforum.org/mail_archive/ur-wg/Archive/doc00011.doc.
- [Jor06] C. Jordan. *Lessons learned from the TeraGrid, Part 1: Manage a large geographically distributed grid, What you can learn from their motivations, strategies, tools, and challenges*. IBM, 2006. URL: <http://www-128.ibm.com/developerworks/grid/library/gr-teragrid1/>.
- [JPX04] H. Jin, Y. Pan and N. Xiao. *Grid and Cooperative Computing*. In *GCC Workshops*, Band 3252 von *Lecture Notes in Computer Science*. Springer, 2004. ISBN 3-540-23578-7.
- [KACJ05] B.K. Kim, D.U. An, S.J. Chung and H.J. Jang. *Grid Accounting Information Gathering System with Access Control*. In Zhang et al. [ZTY⁺05], Seiten 845–850. URL: <http://www.springerlink.com/media/>

- 80jppvltyqkg4d2v9xaw/contributions/k/2/0/t/k20t1ejycgufd8r3.pdf.
- [Kno05] H. Knospe. *On Further Development of Authentication, Authorization and Accounting in Next Generation Networks*, 2005. URL: http://www.verwaltung.fh-koeln.de/imperia/md/content/verwaltung/dezernat5/sg51/wissberichte/wissreihe2_2005.pdf.
- [MHL⁺05] J. MacLaren, T. Haupt, W. Lee, S. Newhouse and K. Keahey. *Grid Economy Use Cases*. Grid Economic Services Architecture Working Group, Global Grid Forum, 2005. URL: http://www.gridforum.org/Public_Comment_Docs/Documents/Sep-2005/draft-ggf-gesa-usecases-01-9.pdf.
- [MLMA⁺03] R. Mach, R. Lepro-Metz, Booz A.H., S. Jackson and L. McGinnis. *Usage Record – Format Recommendation*. Usage Record Working Group, Global Grid Forum, 2003. URL: http://forge.ggf.org/sf/sfmain/do/downloadAttachment/projects.ggf-editor/tracker.submit_ggf_draft/artf3385?id=atch3485.
- [MN05] J. MacLaren and S. Newhouse. *Grid Economic Services Architecture Working Group GESA-WG*. Management Area Meeting (GGF14 in Chicago), 2005. URL: https://forge.gridforum.org/sf/docman/do/downloadDocument/projects.gesa-wg/docman.root.presentations_and_minutes/doc5913;jsessionId=E9BB7F3ED7AF93C477D39636ED45A605.
- [Mob06] *Moby Dick - Mobility and Differentiated Services in a Future IP Network Webpage*, 2006. URL: <http://www.ist-mobydick.org/>.
- [MSZ02] J. MacLaren, V. Sander and W. Ziegler. *Grid Resource Allocation Agreement Protocol GRAAP – Basic Building Blocks for a Super-Scheduler Service*. Grid Resource Allocation Agreement Protocol Working Group, Global Grid Forum, 2002. URL: http://www.fz-juelich.de/zam/RD/coop/ggf/graap/GRAAP_Charter_04.10.02.pdf.
- [MWS06] C. Morariu, M. Waldburger and B. Stiller. *An Accounting and Charging Architecture for Mobile Grids*. Technischer Bericht ifi-2006.06, Department of Informatics, University of Zurich, 2006. URL: <ftp://ftp.ifi.unizh.ch/pub/techreports/TR-2006/ifi-2006.06.pdf>.
- [New03a] S. Newhouse. *Grid Economic Services*. Grid Economic Services Architecture Working Group, Global Grid Forum, 2003. URL: <http://www.lesc.ic.ac.uk/markets/Resources/gesaservicesdraft.pdf>.

- [New03b] S. Newhouse. *Grid Economic Services Architecture (GESA)*. Grid Economic Services Architecture Working Group, Global Grid Forum, 2003. URL: <http://www.doc.ic.ac.uk/~sjn5/GGF/CompEconArch-GGF7.pdf>.
- [OGS06] *OGSA Resource Usage Service GGF Working Group*, 2006. URL: <https://forge.gridforum.org/projects/rus-wg>.
- [Per96] C. Perkins. *IP Mobility Support*. Request for Comments (RFC), Internet Engineering Task Force (IETF), 1996. URL: <http://www.ietf.org/rfc/rfc2002.txt>.
- [PGW03] R. Piro, A. Guarise and A. Werbrouck. *An Economy-based Accounting Infrastructure for the DataGrid*, 2003. URL: <http://ieeexplore.ieee.org/iel5/8914/28196/01261716.pdf?arnumber=1261716>.
- [QL04] Q. Qian and M. Li. *HiGAF: A Hierarchical Grid Accounting Framework*. In *GCC Workshops [JPX04]*, Seiten 650–657. URL: <http://www.springerlink.com/media/9h0bwpwvvg4v2r15hh1k/contributions/1/b/w/k/1bwkjmjxcjxfya21.pdf>.
- [RG06] C.-P. Rückemann and M. Göhner. *Bewertung bestehender Accounting-Ansätze im Bereich des Grid-Computing*. 12. April 2006, D-Grid, Fachgebiete Monitoring, Accounting und Billing im D-Grid-Integrationsprojekt, 2006. URL: http://dgi.d-grid.de/index.php?id=118&filename=mab_accounting_bewertung.pdf&dir=FG2/koordination_mab&task=download&mountpoint=2. 16 Seiten.
- [Rig00] C. Rigney. *RADIUS Accounting*. Request for Comments (RFC), Internet Engineering Task Force (IETF), 2000. URL: <http://www.ietf.org/rfc/rfc2866.txt>.
- [RWRS00] C. Rigney, S. Willens, A. Rubens and W. Simpson. *Remote Authentication Dial In User Service (RADIUS)*. Request for Comments (RFC), Internet Engineering Task Force (IETF), 2000. URL: <http://www.ietf.org/rfc/rfc2865.txt>.
- [San03] T. Sandholm. *SweGrid Job Account Reservation Manager (JARM)*, 2003. URL: <http://www.sgas.se/docs/SGAS-JARM-DD-0.1.pdf>.
- [Sch03] J. Schiller. *Mobilkommunikation*. Pearson Studium, zweite Auflage, 2003. ISBN 3-8273-7060-4.
- [SGE⁺04] T. Sandholm, P. Gardfjäll, E. Elmroth, L. Johnsson and O. Mulmo. *An OGSA-Based Accounting System for Allocation Enforcement across*

- HPC Centers*, 2004. URL: http://portal.acm.org/ft_gateway.cfm?id=1035207&type=pdf&coll=GUIDE&dl=GUIDE&CFID=70792374&CFTOKEN=58477303.
- [Sha03] M. Shapiro. *AMIE Installation and Configuration*, 2003. URL: <http://scv.bu.edu/AMIE/Docs/AMIEInstallationAndConfiguration.doc>.
- [Sha04] M. Shapiro. *AMIE in the TeraGrid*, 2004. URL: <http://scv.bu.edu/AMIE/Docs/AMIE-TeraGrid.doc>.
- [Sim94] W. Simpson. *PPP in HDLC-like Framing*. Request for Comments (RFC), Internet Engineering Task Force (IETF), 1994. URL: <http://www.ietf.org/rfc/rfc1662.txt>.
- [Szc05] P. Szczytowski. *VCLab as an Example of GRIDifying Virtual Scientific Experiments*, 2005. URL: <http://ewic.bcs.org/conferences/2005/1stelegi/session2/paper7.pdf>.
- [TCF⁺03] S. Tuecke, K. Czajkowski, I. Foster, J. Frey, S. Graham, C. Kesselman, T. Maquire, T. Sandholm, D. Snelling and P. Vanderbilt. *Open Grid Services Infrastructure (OGSI), Version 1.0*. Open Grid Services Architecture Working Group, Global Grid Forum, 2003. URL: <http://www.gridforum.org/documents/GFD.15.pdf>.
- [Ter06] *TeraGrid Webpage*, 2006. URL: <http://www.teragrid.org/>.
- [UR-06] *Usage Record Working Group Webpage*, 2006. URL: <http://www.psc.edu/~lfm/PSC/Grid/UR-WG/>.
- [Web06] *Web Services Addressing Working Group Webpage*, 2006. URL: <http://www.w3.org/2002/ws/addr/>.
- [Wes05] Stefan Wesner. *Towards an Architecture for the Mobile Grid*. *it - Information Technology*, 47(6):pages 336–342, 2005. URL: <http://www.atypon-link.com/OLD/doi/pdf/10.1524/itit.2005.47.6.336>.
- [WSD06] *Web Services Description Language (WSDL) 1.1 Webpage*, 2006. URL: <http://www.w3.org/TR/wsdl>.
- [WSR06] *The Web Service Resource Framework (WSRF) Webpage*, 2006. URL: <http://www.globus.org/wsrp/>.
- [WSS06] *OASIS Web Services Security (WSS) TC Webpage*, 2006. URL: http://www.oasis-open.org/committees/tc_home.php?wg_abbrev=wss#technical.

- [ZA99] G. Zorn and B. Aboba. *RADIUS Accounting Server MIB*. Request for Comments (RFC), Internet Engineering Task Force (IETF), 1999. URL: <http://www.ietf.org/rfc/rfc1662.txt>.
- [ZTY⁺05] Y. Zhang, K. Tanaka, J.X. Yu, S. Wang and M. Li, Herausgeber. *Proceedings, Web Technologies Research and Development – APWeb 2005, 7th Asia-Pacific Web Conference, Shanghai, China, March 29 – April 1, 2005*, Band 3399 von *Lecture Notes in Computer Science*. Springer, 2005. ISBN 3-540-25207-X.

Index

- A4C, 38, 41
- A4C-Client, 43
- A4C-Server, 41–43
- AAA, 37, 38
- AAA-Client, 38, 39
- AAA-Server, 38, 39
- Abrechnung
 - service-level-basierte, 15
 - verbrauchs-basierte, 15
- Abrechnungsdaten, 4, 6–8, 10, 13, 17, 21, 23, 24, 30–34, 43
- Abrechnungseinheiten, 6, 14, 16, 18, 19, 22, 24, 30
- Abrechnungsmanagement, 2, 37
- Account Management and Information Exchange, 3–6, 23, 24
- Accounting Processor for Event Logs, 3, 7–9, 23, 24
- Agreement, 26
- AMIE, *siehe* Account Management and Information Exchange
- Analysefunktion, 23
- APEL, *siehe* Accounting Processor for Event Logs
- Application Service Provider, 14, 36
- ASP, *siehe* Application Service Provider
- Auditing, 38, 39, 41, 43
- Bankdienst, 36
 - dezentraler, 9
- Bankkomponente, 19–21
- Berechnungsdienst, 23
- Billing-System, 15
- Charging, 38, 41
- Condor, 9
- CPU-Zeit, 8, 9, 14, 18, 24
- D-Grid, 1
- Datenbank, 4, 6, 7, 9, 33, 43
 - lokale, 21
 - relationale, 7
 - XML-basierte, 20
 - zentrale, 5, 6, 8, 9
- DGAS, *siehe* Distributed Grid Accounting System
- Diameter, 39, 40, 43
- Dienste
 - virtuelle, 22, 23
- Dispatcher, 18
- Distributed Grid Accounting System, 3, 9, 21
- Dynamik, 23
- Erweiterbarkeit, 16
- GESA, *siehe* Grid Economic Services Architecture
- GGF, *siehe* Global Grid Forum
- GIIS, *siehe* Grid Index Information Server
- Global Grid Forum, 1, 16, 21
- Globus Toolkit, 19, 21
- GOC, *siehe* Grid Operations Centre
- GRAAP, 26, 34
- GRASP, *siehe* Grid Based Application Service Provision
- Grid Accounting Services Architecture, 3, 12, 14, 19, 21, 23, 32
- Grid Based Application Service Provision, 3, 14–16, 23

- Grid Economic Services Architecture, 26, 34, 36, 37
- Grid Index Information Server, 8
- Grid Operations Centre, 7, 8
- Grid Service Accounting Extensions, 3, 16, 23
- Grid Services, 21, 34
- Grid-based Application Service Provision, 14
- Grid-Dienste, 29, 32, 34, 36, 37, 40, 41, 43 virtuelle, 32
- Grid-Informationsdienste, 18
- Grid-Middleware, 18, 19, 21, 24
- Grid-Ressourcen, 4, 6, 8, 10, 11, 13, 16, 18, 19, 22, 24, 29, 30, 32, 34–37, 40, 41, 43 heterogene, 23
homogene, 6, 22
virtuelle, 22, 23
- Grid-Standards, 25
- GridBank, *siehe* Grid Accounting Services Architecture, 32
- GSAX, *siehe* Grid Service Accounting Extensions

- HLR, *siehe* Home Location Register
- Home Location Register, 9, 10
- Homogenität, 9
- Hot-Accounting, 23

- Informationsdienst, 23
- Interoperabilität, 14, 21, 23, 24, 32

- JDL, *siehe* Job Description Language
- Job Description Language, 30

- Langzeitarchivierung, 23
- LSF-Batchsysteme, 7

- Marktplatz
virtueller, 34
- Messereignisse, 17
- Meta-Scheduler, 18
- Metriken, 14, 18, 23
- MTLS, 33
- Multi-Provider-Szenarien, 23, 32, 41

- Netzzugangsprotokoll, 38
- Nimrod/G, 3, 18, 23
- Nimrod/G Agent, 18
- Nutzerauftrag, 5, 10, 12, 15, 18, 21
- Nutzerkonten, 4, 14, 19–21
- Nutzungsstatistiken, 31

- OASIS, 25
- OGSA, *siehe* Open Grid Services Architecture
- OGSI, *siehe* Open Grid Services Infrastructure
- Open Grid Services Architecture, 14, 16, 18, 21, 26, 34, 36
- Open Grid Services Infrastructure, 20, 26

- PBS-Batchsysteme, 7
- PKI, *siehe* Public Key Infrastructure
- Plug-Ins, 7
- Policy, 17
- Policy-Datenbank, 39
- Port-Type, 35
- Prioritätensteuerung, 31
- Prognosefunktion, 23
- Projektkonto, 5, 6, 21, 22
- Public Key Infrastructure, 11, 13, 21

- QoS, *siehe* Quality of Service
- Quality of Service, 16, 18, 19, 42

- R-GMA, *siehe* Relational Grid Monitoring Architecture
- RADIUS, 38–40
- Relational Grid Monitoring Architecture, 7, 9
- Resource Usage Service, 26, 32–34, 36
- Ressourcen-Broker, 21, 24, 27, 29
- Ressourcenanbieter, 4, 5, 11, 12, 18
- Ressourcenmanagement, 1

- RUS, *siehe* Resource Usage Service
- SAML, 43
- Scheduler, 5, 6, 18, 24, 27, 29
- Scheduling, 1
- Service Data Elements, 33, 34
- Service Level Agreements, 15, 16, 18, 26, 28, 30, 42–44
- SGAS, *siehe* SweGrid Accounting System
- Sicherheitsaspekte, 33, 37
- Single Sign On, 43
- Skalierbarkeit, 21
- SLAs, *siehe* Service Level Agreement
- Standardisierungsgremien, 1, 25
 - GGF, 1, 25, 26, 30, 33, 34
 - IETF, 1, 25, 37, 40, 41, 43
- Super-Scheduler-Dienst, 29
- SweGrid Accounting System, 3, 9, 19, 23

- Task Farming Engine, 18
- Teilabrechnung, 24
- TeraGrid, 4–6
- TeraGrid Central Database, 4, 6
- TFE, *siehe* Task Farming Engine
- TGCD, *siehe* TeraGrid Central Database
- Transaktionshistorie, 21

- UR, *siehe* Usage Record
- Usage Record, 7, 8, 13, 21, 26, 30, 32, 34

- VHE, *siehe* Virtual Hosting Environment
- Virtual Hosting Environment, 14
- Virtualisierungskonzept, 22, 23
- Virtuelle Organisation, 1, 6, 9, 10, 14, 20–23, 26, 32, 36, 43
- Visualisierungsdienst, 23
- VO, *siehe* Virtuelle Organisation

- Währungseinheiten, 19, 35
 - monetäre, 14, 36
 - virtuelle, 10, 14
- Wallclock-Zeit, 9, 14, 22, 24
- Web Service Description Language, 35
- Web Service Level Agreements, 15
- Web Services, 14, 21
- Working Groups, 25, 37
- Workload Manager, 33
- WS-Addressing, 28
- WS-Agreement, 26–29, 34
- WSDL, *siehe* Web Service Description Language
- WSLA, *siehe* Web Service Level Agreements
- WSRF, 26, 28

- XML, 27

- Zugriffsrechte, 20