



Middleware-übergreifendes Monitoring: Evaluierung und Auswahl von Komponenten

(AP 1.2)

Autoren:

Timo Baur (LRZ)

Tibor Kalman (GWDG)

Tobias Lindinger (LMU)

Anne Milbert (AEI)

Gevorg Pogoshyan (FZK)

Mathilde Romberg (FZJ)

29.02.2008

1	Einleitung	3
2	Bestandsaufnahme	3
2.1	Im D-Grid verfügbare Sensoren.....	3
2.2	Im D-Grid verfügbare Monitoringdienste.....	3
2.3	Im D-Grid verfügbare Komponenten für VO-Management.....	7
2.4	Im D-Grid verfügbare Benutzerschnittstellen	8
2.5	Zugrundeliegende Datenformate	11
3	Evaluierung zu integrierender Komponenten	12
3.1	Kriterien	12
3.2	Evaluierung der Monitoringdienste.....	13
3.3	Datenspeicherung: MySQL oder Postgres.....	14
3.4	VO-Managementkomponenten	14
3.5	Benutzerschnittstellen.....	15
3.6	Datenformate.....	16
4	Auswahl der Komponenten für das D-MON Monitoring-Testbett.....	16
4.1	Sensoren	17
4.2	Monitoringdienste	17
4.3	Komponenten des VO-Management	17
4.4	Benutzerschnittstellen.....	17
4.5	Zugrundeliegendes Datenformat	18
5	Entwurf einer Architektur	18
6	Referenzen	20
	Anhang A-1: Vergleich der Bezeichner für Computing-Dienste und Ressourcen	21
	Anhang A-2: Vergleich der Bezeichner für Jobinformation.....	24
	Anhang A-2: Vergleich der Bezeichner für Siteinformation.....	24
	Anhang B: Review of existing Grid Monitoring Systems.....	25

1 Einleitung

Das Projekt D-MON beschäftigt sich mit der middleware-übergreifenden Weiterentwicklung und der Zusammenführung zentraler Monitoringsysteme im D-Grid.

Im vorliegenden Dokument wird beschrieben, welche Software-Komponenten und Konzepte für eine solche Weiterentwicklung im Rahmen des Projektes in Frage kommen, wie diese zusammenspielen sollen und an welcher Stelle eine konkrete Überarbeitung stattfinden muss.

Dazu werden in Kapitel 2 Software-Komponenten vorgestellt, die derzeit im D-Grid eingesetzt werden. Anschließend wird in Kapitel 3 auf Basis eines Kriterienkatalogs evaluiert, welche Komponenten eingesetzt werden könnten, und wo deren Stärken und Schwächen für den Einsatz in D-MON liegen. In Kapitel 4 werden die ausgewählten Komponenten explizit genannt, welche im Arbeitspaket 2 im D-Mon Testbett aufgesetzt werden, ihr Zusammenspiel wird in Kapitel 5 im Rahmen einer überblickartigen Architektur betrachtet und es werden die dazu von D-MON vorzunehmenden Erweiterungen beschrieben.

Als Motivation und Hintergrundinformation wird ferner ein Review existierender Grid Monitoring Systeme im Anhang des Dokuments in englischer Sprache beigetragen.

2 Bestandsaufnahme

Das Ziel des D-MON Projektes ist eine horizontale (middleware-übergreifende) und vertikale Integration des Ressourcen- und Dienstmonitoring im D-Grid. Schritte in diese Richtung sollen durch den Aufbau eines Monitoringsystems, welches virtuelle Organisationen unterstützt (VOs) und middlewareübergreifend arbeitet, unternommen werden.

Grundsätzlich muß ein VO-basiertes Monitoringsystem aus fünf Klassen von Komponenten bestehen:

- 1) Sensoren, welche die Daten über den Ressourcenzustand messen (siehe Kapitel 2.1)
- 2) Monitoringdienste, welche die Daten zur Verfügung stellen (siehe Kapitel 2.2)
- 3) VO-Managementsysteme, welche über die Zuordnungspolicy von Ressourcen zu virtuellen Organisationen wissen (siehe Kapitel 2.3)
- 4) Schnittstellen zum Monitoringdienst, beispielsweise Benutzerschnittstellen, welche Daten für den Benutzer darstellen (siehe Kapitel 2.4)
- 5) Datenformate, welche den von den verschiedenen Sensoren, Diensten und Schnittstellen prozessierten Daten Schemavereinbarungen zugrunde legen und die ausgetauschte Information eindeutig beschreiben (siehe Kapitel 2.5)

Im Folgenden werden die Wichtigsten der Systeme aufgeführt, die im D-Grid für die jeweiligen Komponentenklassen zur Verfügung stehen, bzw. eingesetzt werden könnten.

2.1 Im D-Grid verfügbare Sensoren

Sensoren befinden sich direkt an den Ressourcen (bzw. den von ihnen überwachten Diensten) und dienen den Monitoringdiensten als Datenquellen. Die Daten über den Zustand der überwachten Ressourcen und Dienste werden in jeweils bestimmten Datenformaten beschrieben und einem Monitoringdienst bereitgestellt.

Sensoren sind in der Regel eng mit Monitoringdiensten verbunden bzw. passen aufgrund der verwendeten Schnittstellen oft nur zu bestimmten Monitoringdiensten. Deshalb wird mit Hinweis auf Kapitel 2.2 (Monitoringdienste) und Kapitel 2.5 (Datenformate) auf eine detaillierte Bestandsaufnahme eingesetzter Sensoren verzichtet; der D-MON Prototyp setzt auf den Monitoringdiensten auf.

2.2 Im D-Grid verfügbare Monitoringdienste

Monitoringdienste dienen dem Auffinden der von den Sensoren gewonnenen Monitoringdaten.

Als Monitoringdienste dienen im D-Grid mehrere verschiedene Implementierungen, welche aus historischen Gründen verschiedene Schnittstellen, Datenformate und Sensoren unterstützen. Welche Implementierung im Einzelfall konkret eingesetzt wird, ist in der Regel abhängig von der eingesetzten Middleware sowie der virtuellen Organisation und dem Projekt.

Die meist verbreiteten dieser Dienste werden im Folgenden im Überblick dargestellt, und es wird kurz beschrieben, welche Schnittstellen, Datenformate und Sensoren verwendet werden. Dabei werden zunächst verbreitete Systeme gängiger Middlewares genannt, dann die Systeme, die von einzelnen virtuellen Organisationen und Projekten im D-Grid eingesetzt werden.

2.2.1 MDS4 (Globus)

MDS4 [1] ist der Standard Informationsdienst zum Monitoring von Ressourcen und Diensten in einer Globus 4 Infrastruktur.

Es ist gemäß des in der OGSA (Open Grid Service Architecture – Standard für Grid Architekturen) [2] empfohlenen WS-RF (Webservice Resource Framework) implementiert und ermöglicht so neben der Gewinnung von Zustandsdaten von Gridressourcen auch ein granulares, verteiltes und übergreifendes Management der Ressourcen innerhalb einer Grid-Infrastruktur. Es sieht Notifikationsmechanismen und Trigger vor und stellt Schnittstellen für OGSA, WS-RF und Globus-konforme Dienste bereit, die auf Monitoringdaten zugreifen müssen (wie z.B. Scheduler, Monitoringportlets, Benchmarkplattformen).

Das in D-Grid eingesetzte MDS 4 verwendet neben dem GLUE 1.1 Schema D-Grid spezifische Erweiterungen zur Speicherung der Monitoringdaten, welche anlehnend an GLUE 1.3 spezifiziert sind. Die WS-RF basierten GT4-Dienste (RFT, WS-GRAM) publizieren ihre Monitoringinformationen automatisch in MDS4, welches zusätzlich um Informationsprovider für beispielsweise Job-Monitoring erweiterbar ist.

Im D-Grid existieren bereits eine Reihe solcher für D-Grid-Zwecke angepasste Informationsprovider, z.B. ein Provider für Wartungsmeldungen (Geomaint), sowie Informationsprovider für verschiedene Batchsysteme.

MDS4 hält die Monitoringdaten nicht-persistent in dem verteilten Speicher der MDS4 Webservices vor. Suchanfragen erfolgen über XQuery. Es gibt derzeit keine Möglichkeit, die Monitoring-Daten zu archivieren oder historische Monitoring-Informationen abzufragen.

2.2.2 CIS (UNICORE)

UNICORE 6 [3] bietet einen Common Information Service (CIS) [4], der statische und dynamische Monitoring-Daten der UNICORE Services sammelt, in einer XML-Datenbank speichert und über RSS-Feed im Atom Syndication Format publiziert. Daneben können die CIS Monitoring-Daten über XQUERY abgefragt werden. Auf den jeweiligen UNICORE Services sammelt eine Information Provider Komponente die Daten und gibt sie auf Anfrage an den CIS weiter. Der CIS benutzt als Datenformat aktuell das Common Information Model (CIM)-Schema (DMTF Standard) und wird, sowie es verfügbar ist, das GLUE2 Schema zur Weitergabe von Daten benutzen. Der CIS stellt Informationen zur Systemüberwachung zur Verfügung, zurzeit jedoch noch keine detaillierten Jobmonitoring- und Accountingdaten. Die Integration von SAML-VOMS (Security Assertion Markup Language – Virtual Organisation Management System) ist in der Entwicklung. Für CIS wird ein map-basierter Monitoring Service als Benutzerschnittstelle entwickelt.

2.2.3 MDS2 (Globus Toolkit 2)

MDS2 basiert auf OpenLDAP und stellt eine dezentrale Monitoring-Architektur dar. Der Information Provider sammelt dynamische und statische Informationen von den Ressourcen und stellt sie im LDIF Format von LDAP zur Verfügung. MDS 2 implementiert ferner einen Cache Mechanismus, der die Informationen mit einem Cache Time-To-Live (TTL) Feld kennzeichnet. Die Monitoring-Daten werden

von einem Grid Resource Information Service (GRIS) gespeichert, der auf allen Ressourcen läuft. Jeder GRIS kann sich bei einem Grid Information Index Service (GIIS) anmelden, der auf der Site Ebene zuständig für die Monitoring-Daten ist. Die MDS2-Hierarchie kann einfach erweitert werden, denn ein GIIS kann Client und Server sein.

Das MDS Core Schema oder das GLUE 1.1 Schema können als GRIS und GIIS Datenformat benutzt werden. Da MDS2 auf dem Standard Lightweight Directory Access Protocol (LDAP) v3 basiert, kann sowohl ein GRIS, als auch ein GIIS mit einem normalen LDAP-Klienten abgefragt werden.

Die Standardversion enthält insbesondere ein Sicherheitsproblem in Bezug auf die OpenLDAP und Cyrus SASL Bibliotheken und wurde von den Globus Entwicklern als obsolet gekennzeichnet. Es gibt einen Patch, der über die offizielle Globus Webseite heruntergeladen werden. Dennoch wird von den Globus-Entwicklern dringend empfohlen, bestehende Installationen auf MDS4 zu migrieren. MDS2 gilt als „deprecated“ (obsolet) und wird nicht mehr Teil künftiger Versionen des Globus Toolkit sein [5].

2.2.4 BDII (gLite)

BDII (Berkeley Database Information Index) [6] basiert auf dem von älteren Globus-Versionen sowie der Middleware gLite [7] eingesetzten Monitoringdienst MDS2. Dieser dient als Datenquelle, hält Daten vor und stellt sie per LDAP bereit. Ebenso wie beim MDS2, läuft ein lokaler GRIS (Grid Resource Information Service) auf den Grid Ressourcen und sammelt dynamische und statische Informationen von den erweiterbaren Information Providern. Ein BDII dient dann als domänen-übergreifender Aggregator der die Daten aller GRIS-Komponenten einsammelt. Die Hierarchie ist frei stapelbar: Ein BDII kann auch die Daten anderer BDII-Installationen sammeln.

Obwohl MDS2 von seinen Entwicklern als „deprecated“ gekennzeichnet wurde, sind MDS2 und BDII im produktiven Umfeld vor allem im Bereich LCG nach wie vor äußerst wichtige Komponenten, auf die derzeit noch nicht verzichtet werden kann.

2.2.5 R-GMA (gLite)

R-GMA [8] stellt eine relationale und verteilbare virtuelle Datenbank aus dem gLite Umfeld dar, welche SQL-Anfragen ermöglicht. Die Informationen werden in einer relationalen Datenbank gespeichert. Für jede VO wird eine virtuelle Datenbank benutzt. Das Consumer-Producer Modell des GMA Standards ist so implementiert, dass der Registry-Dienst nicht direkt abgefragt werden muss. Ein Consumer bekommt die Daten immer direkt von den R-GMA Komponenten an den Ressourcen und so werden die Daten immer aktuell ausgeliefert. Die Information Provider an den Ressourcen sind Teil von R-GMA oder extern und arbeiten mit SQL-Statements.

R-GMA verfügt über eine vielversprechende Architektur, scheint aber im produktiven Betrieb nicht immer unproblematisch zu sein, vor allem hinsichtlich der Performanz und Skalierbarkeit. R-GMA hat eigene Schnittstellen, welche u.a. Webservice Anfragen verwenden.

2.2.6 AMON (HEP)

AMON [9] ist ein Job-Monitoring System aus dem HEP-Grid. Es ist in der Lage, detaillierte Informationen über Nutzerjobs zu sammeln und diese an einen Grid-Monitoringdienst (momentan R-GMA) weiterzugeben. In dieser Hinsicht kann es als ein Sensor für R-GMA aufgefasst werden, der gegebenenfalls auch für andere Monitoringdienste erweitert werden könnte.

2.2.7 Stellaris (AstroGrid)

Stellaris [10] ist der Metadaten Management Service des Astro-Grid. Als Datenbeschreibungsformat und Anfragesprache werden RDF und SPARQL verwendet. Die Daten werden in einer relationalen Datenbank gespeichert, diese ist per HTTP abfragbar. Die derzeitige Verwendung beschränkt sich vor allem auf applikationsspezifische Metadaten, aber es eignet sich auch für Ressourcen-Monitoring. Stellaris verwendet verschiedene Komponenten als Information Provider. Es verfügt beispielsweise

über Adapter, so dass es aus MDS4 lesen kann. Auch gibt es Komponenten zum Erfassen des Zustands robotischer Teleskope. MDS4 und RDF verwenden XML-codierte Dateien, so dass eine Konvertierung leicht möglich ist.

Eine entsprechende Koppelung zwischen dem MDS4 des Kern D-Grid und Stellaris wurde bereits im AstroGrid im ersten Teil des D-Grid Integrationsprojektes umgesetzt. Für die wichtigsten Ressourcen werden hier auch die selben Kennzahlen verwendet.

2.2.8 RGID/D-GRDL (MediGRID/Instant-Grid)

Die Reliable Grid Information Database (RGID) [11] basiert auf der Beschreibungssprache D-Grid Resource Description Language (D-GRDL) [12], die für Ressourcenbeschreibungen im D-Grid gedacht ist. Derzeitige Verwendung findet vor allem für die Unterstützung von Scheduling und das Monitoring statischer Daten in MediGRID und im Instant-Grid statt. Die RGID kann sowohl als ein lokaler Ressource Sensor funktionieren, als auch andere Monitoring-Dienste (z.B. MDS4, Ganglia) abfragen. Die Informationen werden mit Testergebnissen gekoppelt, in einer nativen XML-Datenbank gespeichert (eXist) und stehen im XML Format (D-GRDL) zur Verfügung, damit eine Konvertierung leicht möglich ist. Mit der Hilfe des RGID-Portals (integrierbar in Gridsphere: JSR-168 kompatible Portlets) können die zugänglichen Hardware- und Softwareressourcen einfach visualisiert werden. Die Integration der Unterstützung anderer Monitoring/Benchmarking-Dienste (z.B. BDII, Jawari, SFT) und des neuen GLUE-Format ist in der Entwicklung.

2.2.9 GRAM-Audit-Logging (Globus)

Die in Globus 4.0.5 neu verfügbare Komponente GRAM-Audit-Logging ermöglicht ein Job-Monitoring in Globus. Die Daten werden (vermutlich aus Datenschutzgründen) jedoch nicht ins MDS publiziert sondern in einer MySQL oder PostgreSQL-Datenbank abgelegt. Insbesondere liefert die Komponente Daten, die eine Zuordnung von Nutzern zu Jobs ermöglicht (Accounting).

	MDS4	CIS	BDII	Stellaris
Anfragesprachen	XQUERY über WS-RF	XQUERY über HTTP	LDAP	SPARQL über HTTP
Datenformate	XML; GLUE 1.1; D-Grid spezifische Erweiterungen	DMTF CIM; Atom Syndication Format, geplant: GLUE2	LDAP; GLUE 1.1.	RDF; GLUE 1.1
Schnittstellen	OGSA/WS-RF; Notifikationsmechanismen (WS-N) und Trigger; ältere Globus-Dienste	XML-Datenbank; Publikation als Atom-Feed (RSS)	LDAP	relationale Datenbank, Zugriff per HTTP (REST)
Sensoren	erweiterbar um zusätzliche Sensoren (sog. Information Aggregator Sources); im D-Grid bereits eine Reihe angepasster Provider vorhanden.	Information Provider-komponente CIP; noch keine detaillierten Job- und Accounting-Daten	Sensoren von MDS2 bzw. Globus Toolkit 2	applikations-spezifische Metadaten, Ressourcen-Monitoring; Adapter zu MDS4

Tabelle 1: Zusammenfassung der Eigenschaften betrachteter Monitoringdienste

2.2.10 Zusammenfassung

Die Tabellen 1 und 2 zeigen eine Zusammenfassung der Eigenschaften der betrachteten Monitoring-Dienste hinsichtlich Schnittstellen, Datenformaten und Sensoren.

	RGID / D-GRDL	R-GMA	GRAM Audit Logging
Schnittstellen	eXist XML-Datenbank: HTTP, XML-RPC, SOAP, WebDAV	eigene Schnittstellen, Webservices	MySQL oder PostgreSQL-Datenbank
Datenformate	D-GRDL; GLUE 1.1 für wichtigste Ressourcen	SQL	Usage Records der Usage Records Working Group im Open Grid Forum: detaillierte Abrechnung (Accounting) von Jobs, Nutzern und VOs
Anfragesprachen	XPATH; XML:DB	SQL-QUERY	SQL-QUERY
Sensoren	Adapter zu MDS4	z.B. AMON	GRAM des lokalen Globus-Containers

Tabelle 2: Zusammenfassung der Eigenschaften betrachteter Monitoringdienste

Wichtig erscheint ausserdem, dass bereits in DGI-I von den Communities MediGrid und AstroGrid in Zusammenarbeit mit der DGI Fachgruppe FG2-6 Konvertierungsmodule entwickelt wurden, die in GLUE 1.1. beschriebene Monitoringdaten zwischen dem MDS4 des Kern D-Grid, Stellaris sowie RGID vermitteln. Zwischen diesen Systemen erfolgt bereits ein aktueller Datenaustausch. Hierfür werden XSL-Transformationen eingesetzt (siehe Abbildung 1).

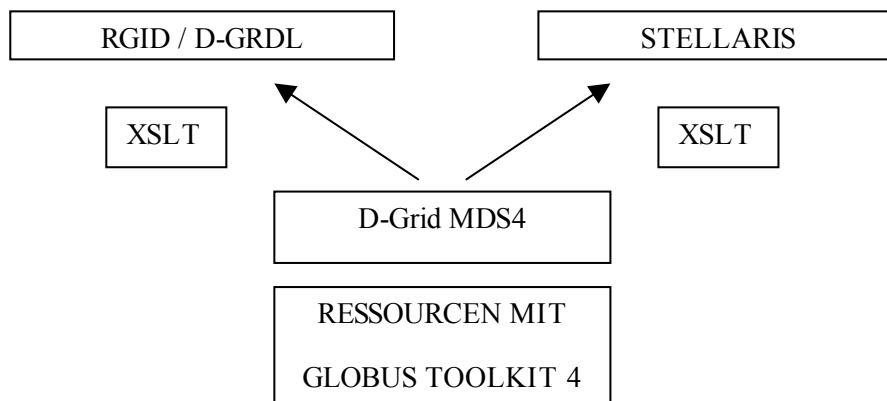


Abbildung 1: Datenaustausch zwischen D-Grid Monitoringdiensten mit Hilfe von XSLT-Transformationen

2.3 Im D-Grid verfügbare Komponenten für VO-Management

Die Komponenten des VO-Managements dienen zur Strukturierung virtueller Organisationen bzw. zu deren Aufbau. Aus der Sicht eines Monitoringsystems ermöglichen sie insbesondere eine Abbildung von Benutzern bzw. Ressourcen zu bestimmten VOs und sind somit Grundlage für Authentisierungs- und Autorisierungs-Management sowie Accounting, aber auch insbesondere für VO-spezifisches Monitoring.

Die in diesem Kapitel genannten Komponenten wurden in DGI-I bereits installiert [13] und werden produktiv in Jülich betrieben. Für das D-MON Testbed (AP 2.2.) sowie andere Projekte wird dort eine Schnittstelle zur Verfügung gestellt werden.

2.3.1 GRRS-Datenbank

Die GRRS-Datenbank (Grid Resource Registry Service) enthält Informationen zu allen im D-Grid registrierten Ressourcen. Insbesondere ermöglicht sie es, eine Ressource einer bzw. mehreren VOs zuzuordnen. Diese Information ist notwendig, um eine VO-spezifische Monitoring-Sicht auf den dynamischen Zustand der Ressourcen im D-Grid generieren zu können. Die GRRS Datenbank (Oracle) enthält die im D-Grid verfügbaren Ressourcen mit Art, Namen und Beschreibung der Ressource, Informationen über installierte Middlewares, den OUs (Operational Unit: Bezeichnung der Einrichtung, die einen Server betreibt) der Hostzertifikate, eine numerische Ressourcen-ID, VO-Zugehörigkeit, Middleware und Hostname sowie einen verantwortlichen Kontakt und wird derzeit im Wesentlichen manuell gepflegt. Es können alle Ressourcenanbieter im D-Grid ihre Ressourcen über ein Webinterface eintragen und beschreiben.

2.3.2 VOMS-Datenbank

Die VOMS-Datenbank (Virtual Organisation Membership Service), von der es auch VO-bezogen mehrere geben kann, enthält Informationen zu Mitgliedern einer VO, d.h. eine Liste der Nutzer einzelner VOs. Insbesondere lässt sich so eine Abbildung von Nutzern auf eine oder mehrere VOs realisieren. VOMS wird im D-Grid nur vereinzelt eingesetzt. Das VO-Management wird stattdessen zentral über VOMRS realisiert.

2.3.3 VOMRS-Datenbank

VOMRS Datenbanken (virtual organization membership registration service) beinhalten prinzipiell alle Informationen der VOMS-Datenbanken, bieten aber darüber hinaus weitere Konfigurationsmöglichkeiten, wie z.B. das Anlegen von Sub-VOs. Für jede VO wird in Jülich eine eigene Instanz des VOMRS betrieben. Die VOMRS Datenbank enthält Daten über alle registrierten Nutzer im D-Grid. Neben Kontaktinformationen und dem OU des Nutzerzertifikates, sind auch Mitgliedschaften in VOs inklusive deren Untergruppen gespeichert. Eine Zuordnung von Nutzern zu VOs sowie von VOs zu Nutzern ist möglich. Der Zugriff erfolgt über SQL.

2.4 Im D-Grid verfügbare Benutzerschnittstellen

Zur Anzeige der Daten aus den verschiedenen Monitoringdiensten sind Benutzerschnittstellen notwendig. Häufig bringen Monitoringdienste bereits Benutzeroberflächen mit oder es können bestimmte Benutzeroberflächen nur mit bestimmten Monitoringdiensten arbeiten.

Dies führt oft zu Unklarheiten über die unterliegenden Fähigkeiten der einzelnen Systeme und einer unscharfen Auswahl der zu verwendenden Komponenten, wenn nur Kriterien bezüglich Optik und Nutzerschnittstellenfunktionalität angelegt werden. Da auf dieser Basis mit dem Nutzerinterface der gesamte unterliegende Monitoringdienst ausgewählt wird, wird eine Entkoppelung der unterliegenden Monitoringdienste begünstigt, was zu Infrastrukturinseln führen kann und einer interoperablen D-Grid Infrastruktur zum gemeinsamen Datenaustausch entgegen wirkt. Deshalb sollten vor der Auswahl einer Benutzeroberfläche die verwendeten Monitoringdienste und deren Schnittstellen klar spezifiziert und miteinander operabel sein.

Im Folgenden werden als Teil der Bestandsaufnahme einige der im D-Grid eingesetzten Benutzerschnittstellen vorgestellt.

2.4.1 web-basierte Benutzerschnittstellen

Gridsphere:

Bei Gridsphere [14] handelt es sich um ein javabasiertes Open Source Portal Framework, das insbesondere für Grid Applikationen entwickelt wurde.

Gridsphere erlaubt für spezifische Anwendungen die Verwendung verschiedener sogenannter Portlets, welche unterschiedliche Nutzeroberflächen realisieren. So können für verschiedene Monitoringdienste auch verschiedene Portlets (z.B. Resource Browser, RGID, Stellaris, MDS2) eingepflegt werden.

Die Authentifizierung von Benutzern ist durch „Hashed password based authentication“ gegeben, das auf eine eigenen Datenbank zugreift oder JAAS (Java Authentication and Authorization Framework) nutzt. Über das Role Based Access Control (RBAC), können den Benutzern des Portals Rollen zugeordnet werden.

Die Präsentation der Daten ist möglich in Tabellenform, als reines XML, durch GeoMaint oder eigene Komponenten in Form von Koordinaten einer Landkarte oder als Chart.

Gridsphere ist in der Lage mit fast allen gängigen Datenbanksystemen (HSQLDB, Oracle, MySQL, ...) zu kommunizieren. Die Integration eines LDAP-Servers ist ebenfalls möglich, genau so wie das Ansprechen von Services, die nach dem WS-RF Standard erstellt wurden.

WebMDS:

Bei WebMDS [1] handelt es sich um ein javabasiertes Tool zum Browsen von XML-basierten Informationen über Grid Services. Es ist Teil des Globus Toolkits und wurde auch speziell für das hier verwendete Datenformat entwickelt.

Die Authentifizierung von Benutzern wird bei WebMDS nicht mitgeliefert, ist aber durch ein Plugin möglich.

Die formatierte Darstellung der Daten erfolgt ueber XSL-Transformationen. Als Darstellungsmöglichkeiten hat der Benutzer die Wahl zwischen Darstellung der Daten in Tabellenform oder als reines XML.

WebMDS verfügt bis dato über keine eigene Datenbank.

GridCSM:

Bei GridCSM [15] handelt es sich um ein für das Kern D-Grid angepasstes und erweitertes WebMDS. GridCSM stellt VO-bezogene tabelarische und geographische Sichten auf Ressourcen der Globus Toolkit 4 Infrastruktur im D-Grid bereit, z.B. Ansicht der Rechenqueues, Clusteransicht, Wartungsmeldungen sowie eine Drill-Down-Funktionalität. GridCSM enthält alle verfügbaren Daten aus den Globus Toolkit Infrastrukturen der VOs im D-Grid. Es ist noch nicht in der Lage, die Informationen aus Unicore und gLite darzustellen und einzusammeln, da es bisher nur Informationen aus dem MDS4 verarbeitet.

D-Grid Webseite mit dem Typo 3 Content Management System:

Bei der D-Grid Webseite handelt es sich um einen, mit dem Content Management System (CMS) Typo3 erstellten Internetauftritt. Die Konfigurationssprache dieses CMS ist TypoScript, ansonsten basiert es auf PHP. Die Webseite enthält Links auf diverse Monitoringsysteme im D-Grid sowie auf manuell geführte Listen und Übersichten von Ressourcen. Sie erfüllt damit den Zweck einer Oberflächenintegration für öffentlich von den Nutzern einsehbare Daten über den Zustand von Grid-Ressourcen und muss deshalb als potentielle Benutzerschnittstelle für Monitoringinformationen mit genannt werden.

Ferner findet über diese Plattform die Verteilung der Dokumentation und der Installationsanweisungen für die Monitoringdienste statt, was für ein D-Grid weites Roll-Out der Standardiensten zum Monitoring im D-Grid elementar wichtig ist. Beispielsweise wird hier die Installation der für das Kern D-Grid erweiterten Sensoren (GeoMaint) sowie des Kern-Monitoringsystems beschrieben.

Zu den von Typo3 bereitgestellten Funktionalitäten gehören unter anderem zeitliche Steuerung zum Einblenden und Entfernen von Inhalten, Rollen- und Rechtssystem, Suchfunktion für statische und dynamische Inhalte, automatische Sitemap, Druckansicht, Nutzerverwaltung, Mehrsprachigkeit, suchmaschinenfreundliche URLs, die sich über verschiedene Zusatzmodule (sogenannte Extensions), wie z. B. Gästebücher, Foren, Newsletter, Statistikmodule etc. erweitern lassen. Bilder, Texte, Tabellen, Animationen sowie externe Datenquellen können ebenfalls verwaltet werden. Seit der Version 4.1 wird auch Ajax verwendet, was eine hohe Dynamik (z.B. durch dynamisches Nachladen von Inhalten) in der Präsentation der Daten ermöglicht.

Typo3 verfügt über eine eigene Datenbank, in der die eingegebenen Inhalte gespeichert werden.

Die auf der Seite bereitgestellten Komponenten ermöglichen die Darstellung von Daten in Tabellenform oder als reines XML.

Die Integration eines LDAP Servers über Typo3 ist möglich. Ebenso verhält es sich mit der Kommunikation mit der Datenbank. Allerdings ist die Kommunikation mit WSRF-Services über Typo3 nicht möglich, es werden auch keine dynamischen Monitoringdaten automatisiert angezeigt.

GridIce:

Bei GridIce [16] handelt es sich um ein vollwertiges Monitoring Tool für Grids. Die Architektur besteht aus den GridICE Sensoren, die Monitoring Informationen beschaffen und dem GridICE Server, der für die Darstellung verantwortlich ist. An dieser Stelle soll jedoch nur die Weboberfläche von GridICE Erwähnung finden, da der unterliegende Dienst weitgehend auf BDII basiert.

GridICE ist VO- und GridService orientiert und verfügt über ein rollenbasiertes System, das sich an den typischen Benutzertypen (VO, GOC, Site) von Grid Applikationen orientiert und daran orientierten Levels organisiert ist (VO Level, Site Administration Level, End-User Level).

Es verfügt über eine eigene Datenbank (PostgreSQL) und ermöglicht somit persistente Datenhaltung. Die Präsentation der Daten ist in Tabellenform möglich, als reines XML, als Koordinate in GeoView oder als Chart.

Als einziges der hier evaluierten Systeme bietet GridICE bereits standardmässig eine VO View auf die Daten aus dem BDII an, auf der VO bezogene Informationen in Tabellenform oder als Chart abgerufen werden können.

Google Maps:

Verschiedene neuere webbasierte Nutzerschnittstellen für das Monitoring verwenden Google Maps als Basis zum Anzeigen geographischer Karten. Google Maps wird in der Regel mit Hilfe von Javascript-Routinen angepasst und erhält auch so die Monitoringdaten von den jeweiligen Monitoringdiensten.

2.4.2 Nicht webbasierte Benutzerschnittstellen

Die UNICORE Klienten:

1. Der UNICORE GPE Client:

Der UNICORE Client ist eine Java Desktop Applikation der UNICORE Grid Middleware. Er wurde nach dem OGF und OASIS Standards entwickelt. Die Oberfläche des Clients ist mit der Swing API erstellt worden, als Workflow Language wird BPEL verwendet.

2. Der UNICORE Rich Client

Neben dem GPE Client bietet das WS-RF konforme UNICORE 6 weitere Klienten, unter anderem einen Kommando-Zeilen Klienten und einen Rich Client, der als Plugin zu Eclipse realisiert ist.

Workflows werden in einem eigenen XML-Format beschrieben, die einzelnen Jobs innerhalb eines Workflows in JSDL (Job Submission Description Language).

3. Allgemein

In UNICORE 6 ist jedem Benutzer eine Rolle zugeordnet, eine explizite Zuordnung von Benutzern zu VOs ist aber nicht realisiert. Der UNICORE Client verfügt über keine eigene Datenbank, da er sich direkt mit den site-spezifischen UNICORE Datenbank verbindet. Job-Monitoring Informationen werden dem Besitzer der Jobs im Klienten angezeigt. Hier besteht direkter Zugriff auf die Ausgabedateien.

2.5 Zugrundeliegende Datenformate

Sollen all die genannten Systeme koordiniert zusammenarbeiten, so ist es notwendig, dass die verwendeten Daten und Datenformate der Teilsysteme kompatibel sind, mit einem gemeinsamen Informationsmodell als Brücke. Im Folgenden wird daher ein kurzer Überblick über die wichtigsten existierenden bzw. im D-Grid eingesetzten Datenformate gegeben.

2.5.1 GLUE

GLUE (Grid Laboratory Uniform Environment) [17] ist ein Standard zur Beschreibung von Grid-Umgebungen, der vom DataTag-Projekt entwickelt wurde und in vielen der heute existierenden Grid-Middlewares - speziell in deren Monitoringsystemen – eingesetzt wird. Die Versionen 1.1 bis 1.3 sind sowohl syntaktisch als auch semantisch weitgehend identisch. Version 2.0 ist zurzeit noch in Entwicklung, wird aber aller Voraussicht nach strukturell und inhaltlich verschieden zu den Versionen 1.x implementiert.

2.5.2 D-GRDL

Die D-GRDL (D-Grid Ressource Description Language) [12] ist eine im D-Grid während des DGI-I Projektes entwickelte Sprache zur Ressourcenbeschreibung. Die D-GRDL ist XML-basiert und sehr generisch, so dass eigentlich alle Grid-Objekte mit ihr beschrieben werden können. Ein Nachteil dieses Ansatzes ist es jedoch, dass man eine konkrete Instanz der Sprache selbst definieren muss, indem man für jeden Eintrag die Werte „Name“, „Wert“ und „Einheit“ setzt. Letztendlich definiert die D-GRDL zunächst nur Syntax aber keine Semantik. Zwei verschiedene Ausprägungen der D-GRDL müssen damit nicht zwangsweise zueinander kompatibel sein. Für die D-GRDL existieren bereit verschiedene Ausprägungen welche auch Attributdefinitionen für die im D-Grid bisher verwendeten GLUE-Dialekte (V1.1./ V1.3) anbieten.

2.5.3 CIM

CIM (Common Information Model) ist ein objektorientierter Standard zur Beschreibung von Infrastrukturen, der von der DMTF (Distributed Management Task Force) entwickelt wurde. Er wird in einer modifizierten Version unter anderem in Windows zur Überwachung des Systemzustands verwendet. Diese Tatsache verhilft CIM neben dessen Standardisierung und generalisierten Einsetzbarkeit zu einer recht großen Verbreitung, jedoch ist das verwendete Objektmodell, das viel Gebrauch von Vererbung macht, recht komplex.

2.5.4 Einschätzung

Viele Monitoringsysteme verwenden nicht ausschließlich einen dieser ursprünglichen Standards, sondern haben im Hinblick auf ihren Einsatzzweck Änderungen bzw. Ergänzungen am Standard vorgenommen. Dies bedeutet, dass zwei Systeme, die z.B. auf derselben GLUE-Version basieren nicht unbedingt vollständig kompatibel sind. Die im Rahmen von D-MON zu integrierenden Schemata, können daher unter Umständen nicht verlustfrei ineinander übersetzt werden.

3 Evaluierung zu integrierender Komponenten

Im Folgenden soll geklärt werden, welche Eigenschaften die im D-MON Projekt zu integrierenden Komponenten haben, welche dieser Komponenten als Grundlage für ein Gesamtsystem verwendet werden können und wo diese erweitert werden müssen.

Hierfür werden zunächst Kriterien definiert, die aus der Anforderungsanalyse aus AP 1.1. sowie dem Projektplan resultieren. Anschließend werden diese gegen die nativen („In-Band“) Middleware-Monitoringdienste, insbesondere die im Projektantrag genannten Systeme UNICORE 6, MDS4 und gLite BDII, evaluiert. In den Folgenden Unterkapiteln folgen dann Aussagen zur Eignung der vorgestellten VO-Managementsysteme, Nutzerschnittstellen und Datenformate als Komponenten für das D-MON Projekt.

3.1 Kriterien

Die Komponenten sollten eine Vielzahl der in AP 1.1. geforderten Anforderungen erfüllen. Insbesondere sollen sie in der Lage sein, in einem Gesamtsystem auf eine standardisierte Art und Weise zusammenzuarbeiten. Darüber hinaus müssen sie den in den Anforderungen formulierten Rollen (Nutzer, VO-Repräsentant, Dienst und GOC) die notwendigen Daten liefern.

Grundlage für die folgende Kriterienliste sind die in AP 1.1 gesammelten Anforderungen, sowie einige weitere Aspekte, die im Laufe der Komponentensichtung als relevant eingestuft wurden.

1. OGSA/WS-RF Kompatibilität

Ein wichtiges Kriterium in D-MON ist, dass die verwendeten Komponenten konform zum Ressourcenmanagement der Open Grid Service Architecture (OGSA) arbeiten sollen, welche im WS-RF Basic Profile die Unterstützung des Web Service Resource Frameworks empfiehlt. Dieses Kriterium wurde bereits im Projektantrag spezifiziert, um den Bemühungen für ein standardisiertes D-Grid entgegenzukommen.

2. Art und Umfang der bereitgestellten Daten

Die Informationsdienste und Sensoren sollen möglichst viele der in AP 1.1. definierten, für verschiedene Rollen notwendigen Daten bereitstellen (siehe AP 1.1. – Szenarien): Nutzer, VO-Repräsentant, Infrastrukturdienst, GoC.

3. verwendete Datenformate

Werden Daten aus verschiedenen Systemen verwendet, ist es notwendig, die verwendeten Datenformate zu kennen. Dies ist darüber hinaus die Grundlage für eine Zusammenführung (Datenintegration) zu einem gemeinsamen Datenschema (siehe auch AP 1.1. – horizontale Integration / Übersetzer).

4. Zuordnungsmechanismus der Daten zu VOs

Für die vertikale Integration sind VO spezifische Sichten auf die Monitoringdaten notwendig (siehe AP 1.1. – vertikale Integration / „VO-aware“ Monitoringdienste). Um dies zu ermöglichen, muss ein Zuordnungsmechanismus vorhanden sein, der spezifische Daten (Ressourcen, Nutzer, Jobs) spezifischen virtuellen Organisationen zuordnet.

5. mögliche Stapelbarkeit / skalierbare Verteilung des Dienstes

Stapelbarkeit und Verteilbarkeit sind Voraussetzung für ein hoch skalierbares und flexibles Gesamtsystem (siehe AP 1.1. - vertikale Integration).

6. Existenz von Adaptern zu anderen Systemen

Bereits existierende Übersetzer und Adaptern zwischen den Systemen können wo möglich wieder verwendet werden (siehe AP 1.1. – horizontale Integration).

7. Authentifizierungsmethoden

Vorraussetzung VO-spezifischer Sichten für bestimmte Nutzer ist deren Authentifizierung (siehe AP 1.1. – vertikale Integration / „VO-aware“ Monitoringdienste)

8. Komplexität der Zusammenführung von Tabellen (JOIN)
Um VO-basierte Sichten erzeugen zu können, ist es notwendig, die Daten, welche über die VO-Mitgliedschaft Auskunft geben, mit den Monitoringdaten zusammenzuführen. Diese Zusammenführung verschiedener Tabellen und Datensätze sollte möglichst performant vor sich gehen. Deshalb spielt die Komplexität der JOIN-Operationen eine Rolle.
9. persistente Datenhaltung
Persistente Datenhaltung wird für caching-Funktionalität und historische Daten benötigt.

3.2 Evaluierung der Monitoringdienste

Es wird deutlich, dass MDS4 und CIS bestens für eine Weiterentwicklung im Rahmen des OGSA/WS-RF Standards geeignet sind. Die bereitgestellten Daten je System sind jeweils unterschiedlich, in der Regel sind aber für alle Nutzergruppen Parameter verfügbar. Die Zuordnung der Daten zu VOs erfolgt bei den Systemen, welche eine solche vornehmen mit unterschiedlichen Methoden. So ist das markieren der jeweiligen Parameter mit zusätzlichen Daten, welche die VO kennzeichnen eine verbreitete Methode (in Anlehnung an VLANs: „tagged VO“). Die andere Möglichkeit (im Falle von MDS4) ist das repository-basierte Verlinken der Einzelteile eines Aggregatorframeworks („untagged VO“). Die Systeme verfügen über verschiedene Adapter und Schnittstellen, welche das Spektrum von XML, SQL, WS-RF / MDS4 und auch system-spezifische Entwicklungen einschliessen. Die Authorisierung erfolgt ebenfalls jeweils über verschiedene Methoden (zertifizatsbasiert mit SAML und/oder WS-Security oder aber über eine SQL-Datenbank). Die Komplexität von JOIN-Operationen ist abhängig davon, ob ein System auf Basis von XML oder SQL arbeitet. Darüber hinaus unterstützen die meisten Systeme persistente Datenhaltung. Die auf der Basis dieser Evaluierung ausgewählten Dienste werden in Kapitel 4.2 vorgestellt.

Tabelle 3: die Charakteristika der Monitoringdienste in Bezug auf die vorgestellten Kriterien

	DGRDL	R-GMA	AMON	GRAM Audit
OGSA/WS-RF Kompatibilität	nein	nein	nein	nein
bereitgestellte Daten	jeweils unterschiedliche, ausgewählte Parameter für alle Nutzergruppen			
verwendete Datenformate	DGRDL, GLUE 1.1	SQL-DB; GLUE 1.3	AMON	Usage Records
Zuordnung Daten zu VOs	nein	ja, tagged	nein	bedingt (User)
Stapelbarkeit/ Skalierbarkeit	ja	ja	n/a	ja
Adapter	MDS4	SQL-DB	R-GMA	nein
Authentifizierungsmethoden	über DB	Zertifikat	n/a	über DB
JOIN-Komplexität	XML: $O(n^2)$	SQL: $O(n)$	n/a	SQL: $O(n)$
persistente Datenhaltung	ja	ja	ja	ja
	MDS4	CIS	BDII	Stellaris
OGSA/WS-RF Kompatibilität	ja	ab 03/2008	nein	nein
bereitgestellte Daten	jeweils unterschiedliche, ausgewählte Parameter für alle Nutzergruppen			
verwendete Datenformate	GLUE 1.1; GLUE 1.3	CIM;künftig: GLUE 2.0	GLUE 1.2	RDF, GLUE 1.1
Zuordnung Daten zu VOs	repository-basiert, untagged	nein	ja, tagged	nein
Stapelbarkeit/ Skalierbarkeit	ja	ja	ja	ja
Adapter	Stellaris, DGRDL	realisierbar	XML	MDS4
Authentifizierungsmethoden	u.a. Zertifikat	künftig: SAML-VOMS	SASL; Zertifikat	n/a
JOIN-Komplexität	XML: $O(n^2)$	XML: $O(n^2)$	$O(n^2)$	n/a
persistente Datenhaltung	nein	ja	nein	ja

3.3 Datenspeicherung

In der Regel greifen die Monitoringdienste auf Datenbanken oder ähnliche Mechanismen zurück um persistent Daten vorzuhalten. Bei diesen Komponenten handelt es sich zwar nicht um Monitoringdienste an sich, jedoch können Datenbanken als Kernkomponenten bzw. Datenspeicher für Informationsdienste dienen, die einige Vorteile im Vergleich zu bestehenden Monitoring-Lösungen bieten können. Datenbanken erlauben es das Schema für die Daten selbst zu wählen. Es besteht daher kein Zwang ein Schema eines einzelnen Monitoringdienstes unverändert zu übernehmen. Stattdessen ist es möglich ein Schema zu etablieren, das zu allen Schemata der Informationsdienste möglichst kompatibel ist. Eine Datenbank kann in mehrere Relationen verteilt gespeichert werden um Duplikate von Daten zu vermeiden und thematisch zu sortieren z.B. Jobmonitoring, Ressourcenmonitoring. Der Betrieb mehrerer Datenbankserver mit identischem Schema ist sowohl verteilt als auch föderiert als Cluster möglich. Diese Tatsache ist von Vorteil für den Anwendungsfall Grid, da beim Ausfall einer Komponente nicht alle Monitoringknoten gleichzeitig ausfallen und ein gewisser Grunddienst daher immer verfügbar ist. Gleichzeitig könnten Monitoringdaten zunächst immer lokal an den Sites gespeichert bleiben, an denen sie angefallen sind. Dies ist insbesondere bei vertraulichen Accountingdaten sinnvoll.

Diese Daten können durch ein feingranulares Rechtemanagementsystem der Datenbanken gesichert werden, sodass eventuell alle oder ein Teil der Daten gezielt an einzelne Nutzergruppen/VOs publiziert werden können.

Ein Vorteil der Datenbanken gegenüber den existierenden Monitoringsystemen ist es, dass sie Daten persistent speichert, und damit Historienfunktionen wie z.B. Accounting ermöglicht. Dies erlauben zwar einige der Monitoringdienste bereits, jedoch erfolgt die Verwaltung der Daten in reinen Datenbanksystemen wesentlich effizienter, so dass Abfragen und Joins wesentlich performanter durchgeführt werden können. Die Durchführung von performanten Joins ist insofern interessant, als dass Monitoringdaten (Job- und Ressourcendaten) mit VO-Informationen zusammengeführt werden müssen.

Auch der Zugriff auf Datenbanken über den SQL Standard ist relativ problemlos, da für beinahe jede Programmiersprache fertige Adapter existieren.

An freien Datenbanken existieren zurzeit, MySQL und Postgres. Beide sind für das Projekt prinzipiell geeignet, MySQL bietet die Verteilung der Daten betreffend jedoch noch leichte Vorteile gegenüber Postgres, es unterstützt beispielsweise Föderation und Replikation besser.

3.4 VO-Managementkomponenten

Um entscheiden zu können, welche Ressourcen und Jobs ein Nutzer im Monitoring sehen soll bzw. darf, müssen Informationen über die Mitgliedschaften von Nutzern in VOs und über die verfügbaren Ressourcen in VOs vorhanden und zugreifbar sein. Diese Informationen werden produktiv im D-Grid momentan nur von den Komponenten zum VO-Management geliefert, welche in Kapitel 2 vorgestellt wurden. Deshalb kann eine Evaluierung verschiedener Komponenten hier entfallen, welche auch Aufgabe anderer Projekte zu den Themen AAI und VO-Management war. In Kürze werden neuere Entwicklungen aus dem IVOM Projekt vorgestellt, welche insbesondere eine gridweite Authentifizierung und Authorisierung mit Shibboleth erlauben. Die neuen IVOM-Komponenten können später mit den D-MON Komponenten verbunden werden (beispielsweise durch Entwicklung eines entsprechenden Adapters und einer Anbindung von Authentifizierungsfunktionalitäten).

3.5 Benutzerschnittstellen

3.5.1 Kriterienliste

- 1) Rollen- und Rechtesystem: Ein Rollen- und Rechtesystem mit frei definierbaren Rollen wird benötigt, um die verschiedenen Rollen (Administrator, VO Leitung, Normaler Benutzer u.a.) und deren Zugriffsrechte auf die Daten umzusetzen.
- 2) Authentifizierungsmodul: Der Benutzer soll mittels seines Grid Zertifikats eindeutig identifiziert und ihm Datensätze aus der DMON Datenbank eindeutig zugeordnet werden können.
- 3) WS-RF: Die eingesetzte Benutzerschnittstelle sollte die Möglichkeit anbieten mit WS-RF Services zu kommunizieren oder es sollte die Möglichkeit geben diese möglichst einfach um diese Fähigkeit zu erweitern.
- 4) Datenbankschnittstelle: Die Möglichkeit zur direkten Kommunikation mit der Datenbank wurde als eine der Minimalanforderungen definiert und sollte im eingesetzten Tool bereits vorhanden sein.
- 5) Integration vorhandener Komponenten: Die Integration von bereits bestehenden Komponenten, als Beispiel sei an dieser Stelle GeoMaint genannt, sollte einfach möglich sein, um nicht zusätzliche Hürden für einen Wechsel zum eingesetzten System zu schaffen.
- 6) GLUE 2.0 Standard: Da der GLUE 2.0 Standard als Basis für das gemeinsame Datenschema definiert wurde, sollte die eingesetzte Benutzerschnittstellen, dieses Datenschema „verstehen“ können oder die Möglichkeit diesen Standard zu unterstützen sollte verhältnismässig einfach implementierbar sein.
- 7) Auswahl bestimmter Sichten: Die Auswahl bestimmter Sichten auf die ermittelten Daten sollte implementierbar sein. Mit Sichten, sind in diesem Kontext das Filtern der Daten nach gemeinsamen Merkmalen (z.B. alle Nutzer einer bestimmten VO), das Anzeigen von Daten die logisch zu einer Gruppe gehören (z.B. alle Ressourcen oder alle Computing Elemente) oder auch Mischformen (z.B. alle Computing Elemente einer Resource einer bestimmten VO) gemeint.

	Rollen- und Rechtesystem	Authentifizierungsmodul	WS-RF	Datenbankschnittstelle	Integration vorhandener Komponenten	GLUE 2.0	Auswahl bestimmter Sichten
Gridsphere	ja	ja	ja ¹	Ja	ja	ja ²	ja
GridCSM	ja	ja	nein	Ja	ja	ja ²	ja
GridICE	ja	ja	nein	Ja	nein	ja ²	ja
WebMDS	nein	ja ³	ja ³	nein	nein	ja ²	Ja

Tabelle 4: Evaluierung der Benutzerschnittstellen

¹ Erfolgreich getestet mit Globus Libraries

² Erweiterbar

³ Durch Plugin implementierbar

Tabelle 4 fasst die Kriterien nach denen die im DGRID eingesetzten Benutzer-Schnittstellen (siehe Kapitel 3.4.1) evaluiert wurden, zusammen.

Eines der in Bezug auf die Projektziele wichtigsten Kriterien ist die Möglichkeit mit der zentralen Datenbank kommunizieren zu können, in der die gesammelten Daten abgelegt werden. Als Minimalanforderung wurde hier die direkte Kommunikation mit der Datenbank über einen SQL-Select formuliert. Idealerweise sollte das eingesetzte Tool allerdings in der Lage sein, WSRF-Dienste anzusprechen.

Ebenso sollte es möglich sein, die erhaltenen Daten zu filtern, so dass ein Nutzer nur die Daten der VO bzw. den VOs zu sehen bekommt, denen bzw. der er auch wirklich angehört. Hierfür ist die eindeutige Authentifizierung eines Benutzers notwendig. Ein weiteres wichtiges Kriterium ist die Unterstützung von Rollen im Authentifizierungsmodul.

Die Erstellung von verschiedenen Sichten auf die Daten mit verschieden hoher Granularität wird von allen eingesetzten Oberflächen unterstützt. Allerdings bieten nur wenige mehr als die Präsentation in Tabellenform und als reines XML an.

Wie der Tabelle entnommen werden kann, werden alle Anforderungen an die Benutzeroberfläche bis dato nur von Gridsphere erfüllt. Aus diesem Grund scheint die Verwendung dieses Portal Frameworks als angebracht.

3.6 Datenformate

Im Folgenden werden die verwendeten Datenformate im Hinblick auf ihre Verwendung in D-MON evaluiert.

3.6.1 Kriterien

Das für D-MON zu verwendende oder zu entwerfende Informationsmodell zur Modellierung der Daten sollte standardkonform zu Grid Standards, insbesondere dem im Open Grid Forum sowie dem Projekt OMII-Europe vorangetriebenen GLUE 2.0 Schema sein, da dieses auf zumindest lange Sicht in den Informationsdiensten der bekannten Middlewares eingesetzt werden wird. Als Minimalanforderung sind jedoch die im AP 1.1 definierten Informationen zu realisieren, sofern entsprechende Informations-Provider existieren. Eine besondere Anforderung ist es unter anderem, zu dokumentieren aus welcher Quelle / Middleware eine spezifische Information bezogen wurde und welcher VO sie zuzuordnen ist.

3.6.2 Evaluierung

Die Tabelle in Anhang A listet die zurzeit von den verschiedenen Informationssystemen zur Verfügung gestellten Daten gruppiert nach Job-Monitoring (Queue) und Infrastruktur-Monitoring (Host) auf.

Bei einem Vergleich der Schemata fällt auf, dass die diversen GLUE-Dialekte ähnliche Informationen enthalten. Lediglich GLUE 2.0 bietet nativ Unterstützung für VOs, unterscheidet sich aber durch seine verschachtelte Objektstruktur wesentlich von den Versionen 1.x. Sowohl die Open Grid Service Architecture OGSA, das OGF, als auch Unicore 6 und Globus Toolkit werden in zukünftigen Versionen GLUE 2.0 standardisieren bzw. verwenden. Es existieren bereits prototypische Instanzierungen für Datenbanken, die um die im vorhergehenden Abschnitt geforderten Felder entsprechend erweitert werden können.

4 Auswahl der Komponenten für das D-MON Monitoring-Testbett

Basierend auf den vorhergehenden Ausführungen, werden nun die Komponenten ausgewählt, welche für das D-MON Projekt eingesetzt werden sollen. Während dieses Projekt den Anspruch hat, die Daten aus mehreren Middlewares integriert überwachen zu können, ist es gleichzeitig notwendig, die Komplexität der durchzuführenden Arbeiten auf einen überschaubaren Rahmen zu beschränken.

Deshalb ist es in einem ersten Schritt nicht möglich, alle die vorgestellten und untersuchten Komponenten auf einmal zu integrieren und dem Kern D-Grid bereitzustellen. Vielmehr soll eine Installation geschaffen werden, welche die im D-Grid verwendeten Middlewares unterstützt und die um weitere Monitoringsysteme erweiterbar ist.

Anhand der Evaluierung in Kapitel 3 konnten geeignete Systeme ausgewählt und ferner technische Lücken identifiziert werden, die geschlossen werden müssen. Im Folgenden werden die Ergebnisse noch einmal kurz vorgestellt und eine Auswahl von Komponenten getroffen, die sich für die Verwendung und Weiterentwicklung im Rahmen des D-MON Projektes eignen.

4.1 Sensoren

Zunächst sollen die Sensoren verwendet werden, die an den ausgewählten Monitoringdiensten bereits verfügbar sind. Wenn notwendig, kann später um weitere Sensorkomponenten erweitert werden oder Ersetzungen vorgenommen werden.

4.2 Monitoringdienste

Im D-Grid werden drei verschiedene Middlewares eingesetzt: Globus Toolkit 4, UNICORE und gLite. Alle drei bringen nativ bereits Monitoringdienste mit. Um Standardkonformität mit OGSA zu erreichen, wurden die WS-RF kompatiblen Dienste GT4 / MDS, UNICORE 6 / CIS ausgewählt. Um darüber hinaus eine Abdeckung aller im D-Grid verwendeten Middlewares zu erreichen, wurde für gLite BDII hinzugenommen, welches als betriebsstabiler als R-GMA gilt und deshalb vordringlich behandelt wird. Die Evaluierung zeigt, das BDII um eine WS-RF Schnittstelle erweitert werden muss. Dies kann innerhalb eines Gesamtkonzeptes geschehen.

Für alle drei Systeme muss eine Zusammenführung der Schnittstellen erreicht werden und es müssen die Beschreibungssprachen angeglichen, beziehungsweise Adapter geschaffen werden.

Für weitere Systeme soll eine Verfahrensweise bzw. Schnittstelle zur Anbindung vorgesehen werden.

4.3 Komponenten des VO-Management

Um eine Zuordnung von Daten zu VOs zu erreichen, müssen die Monitoringdienste mit Komponenten zum VO-Management verknüpft werden. Die Daten aus den im D-Grid vorhandenen VO-Managementsystemen müssen hierzu in die Datenbank des Monitoring integriert werden und es müssen Verfahren für eine einheitliche VO-abhängige Authentifizierung geschaffen werden.

4.4 Benutzerschnittstellen

Auf Basis der in Kapitel 3.4.1 evaluierten Benutzerschnittstellen und der für diese Schnittstellen in Kapitel 4.5.1 erstellten Kriterienliste erscheint GridSphere als die am besten geeignete Benutzerschnittstelle.

Die Authentifizierung von Benutzern und das Zuordnen von Rollen ist in Gridsphere bereits integriert und kann bei Bedarf um eigene Module erweitert oder ersetzt werden.

Das Einbinden von bereits existenten Komponenten durch die flexible Möglichkeit, Portlets einzubinden, in der Vergangenheit als relativ unproblematisch und ist in Bezug auf diverse Komponenten auch schon durchgeführt worden. Als Beispiel sei an dieser Stelle der Sensor GeoMaint genannt, dessen Verwendung bereits mit Gridsphere getestet wurde. Da Portlets kleine Applikationen sind, die unabhängig voneinander innerhalb einer Portalseite laufen, sollte die Integration von bestehenden Modulen auf keinen Fall ein Problem darstellen.

Mit Hilfe von Gridsphere können alle geforderten Schnittstellen (LDAP, Datenbank, WS-RF Services) abgefragt werden. Insbesondere ist an dieser Stelle die Möglichkeit zur Abfrage von WSRF-Services zu nennen, die als eine der Kernanforderungen für die Benutzerschnittstelle definiert wurde. Dies wurde in der Vergangenheit bereits erfolgreich mit Globus Bibliotheken getestet.

Die Möglichkeit zur Auswahl von verschiedenen Sichten ist gegeben. Es können bezüglich der Präsentationen sowohl vorhandene Tools wie zum Beispiel die Charting Komponente von Gridsphere verwendet, aber auch eigene Komponenten entwickelt werden. Die Darstellung der Daten in tabellarischer Form und als XML Dump versteht sich von selbst.

Abschliessend lässt sich noch sagen, dass Gridsphere im D-Grid bereits von vielen Communities (AstroGrid, C3-Grid, etc.) eingesetzt wird und sich hier bereits bewährt hat. Denn die Einführung einer bisher unbekanntes oder nicht weit verbreiteten Benutzerschnittstelle könnte zu Schwierigkeiten in der Benutzerakzeptanz führen.

4.5 Zugrundeliegendes Datenformat

Wie bereits evaluiert, soll aus Gründen der zukünftigen Kompatibilität zu Entwicklungen der Middlewares sowie der nativen Unterstützung von VOs für den D-Grid Prototypen GLUE 2.0 eingesetzt werden. Die hierzu notwendige semantische und syntaktische Konvertierung der Daten scheint für alle beteiligten Middlewares durchführbar. Die in GLUE 2.0 nicht oder nicht mehr definierten Werte können mit Hilfe zusätzlicher Tabellen (z.B. auf Basis von GLUE 1.3) weiterverwendet werden.

5 Architektur

Die in Kapitel 4 ausgewählten Komponenten sollten nun zu einem Gesamtsystem verbunden werden. Dieses sollte möglichst viele der Fähigkeiten der vorgestellten Einzelsysteme vereinen, und darüber hinaus erweiterbar für eventuell zusätzliche Monitoringdienste sein. Die grundlegende Idee des in D-MON zu implementierenden Monitoringsystems ist es, zunächst Informationen aus den Informationssystemen der Middlewares sowie aus den VO-Managementkomponenten zu sammeln und persistent sowie in einem standardisierten Format zu speichern und an eine einheitliche Schnittstelle weiterzugeben. Insofern wirkt der D-MON Prototyp dann als flexible Brücke zwischen verschiedenen Middlewaresystemen.

Für diese Aufgabe eignen sich insbesondere Datenbanksysteme, da sie eigens für eine persistente, standardisierte Datenhaltung entwickelt wurden.

Das Einfügen der Daten erfolgt über zu entwickelnde Adapter, die aus jeweils einem spezifischen Informationsdienst der Middleware lesen und in die Datenbank schreiben können. Diese Komponenten übernehmen gleichzeitig die syntaktische und semantische Konvertierung der Daten.

An eine solche Datenbank können theoretisch bereits eine Vielzahl von bestehenden Komponenten angebunden werden. Da die wenigsten Komponenten jedoch nativ mit einer SQL Datenbank sprechen können und in Grids in der Regel andere Verfahren verwendet werden, werden zusätzliche Adapter entwickelt, über die diese Komponenten Zugriff erhalten können. Insbesondere wichtig ist in diesem Zusammenhang der Zugriff über OGSA standardisierte WS-RF Schnittstellen.

Für BDII sollen ebenfalls Adapter für die Abfrage der Daten über LDAP implementiert werden. Die Adapter für MDS und CIS verfügen beide über WS-RF kompatible Schnittstellen, so dass sie im Idealfall zu einem einzelnen Adapter verschmelzen können, sofern die Implementierungen zueinander kompatibel sind.

An den D-MON Monitoringdienst können dann mit ebenfalls zu implementierenden Adaptern wiederum Dienste im Grid z.B. Metascheduler angebunden werden. Das Gridsphere User Portal kann die Daten direkt per SQL aus der Datenbank, oder aber über einen der implementierten Adapter z.B. einen WS-RF-Adapter abfragen.

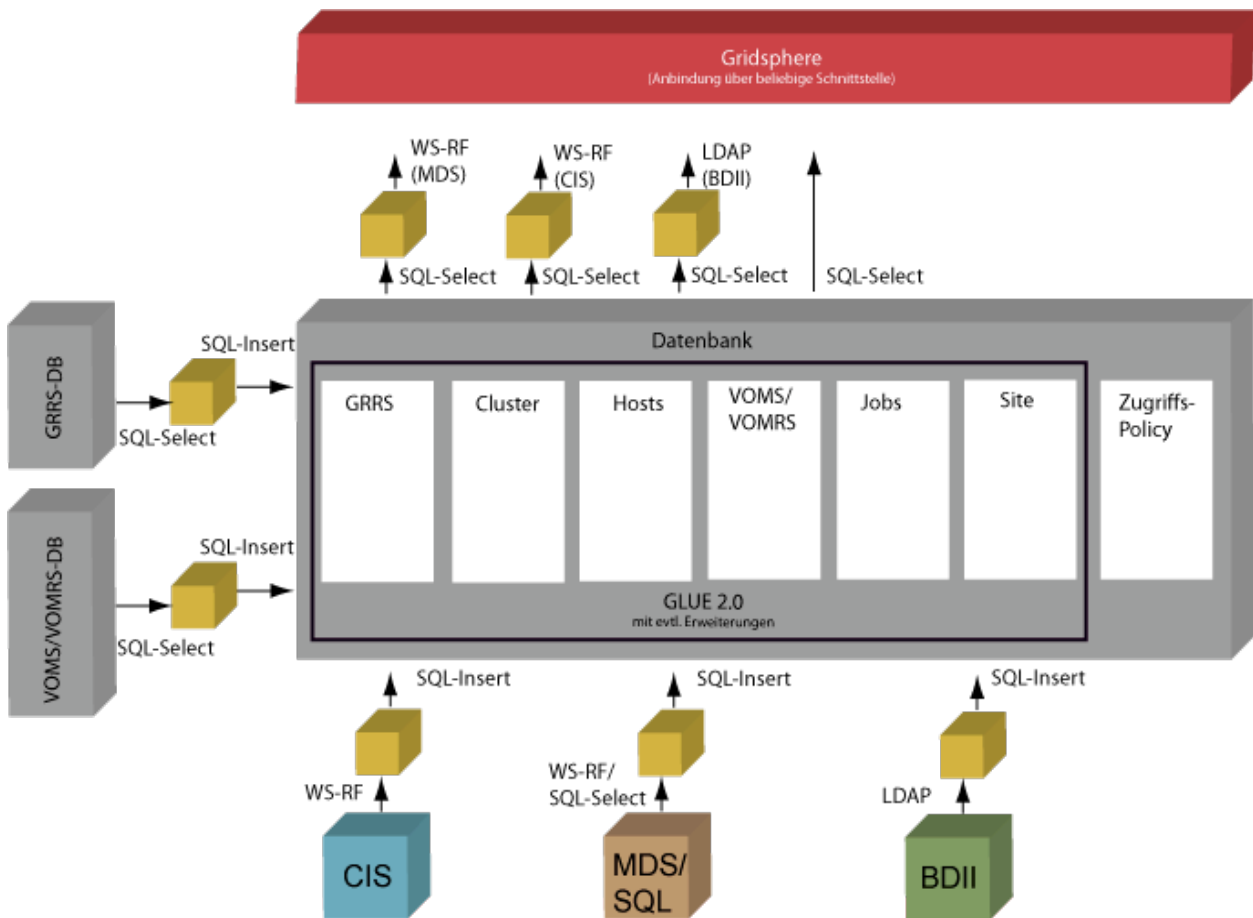


Abbildung 2: Architekturkonzept

Die Begründung für die zentrale Datenbankkomponente, liegt in einer bisher noch wenig betrachteten Anforderung, die bidirektionale Übersetzung der Daten – von jeder Middleware in jede andere – fordert. Diese Anforderung ist nur durch eine zentrale, aber föderierbare, Komponente erfüllbar, da der Aufwand der zu implementierenden Adapter im Falle einer vollständigen Vermaschung quadratisch mit der Anzahl der Middlewares steigen würde. Da gleichzeitig kein bestehender Informationsdienst als zentrale Komponente eingesetzt werden soll, um sich nicht von seiner zukünftigen Entwicklung abhängig zu machen, bleibt nur eine Eigenlösung, für deren Implementierung sich eine SQL Datenbank anbietet. Die in Abbildung 2 abgebildete Architektur versucht alle in AP1.1 gewonnenen Anforderungen sowie die in diesem Dokument evaluierten Kriterien zu berücksichtigen.

Als zentrale, aber föderierbare, Komponente dient auf Ebene einer Site eine MySQL-Datenbank in die alle Informationsdienste sowie VO-Management Komponenten mittels zu implementierender Adapter Monitoring-Informationen speisen. Diese Daten werden in einem auf GLUE 2.0 basierendem, ggf. erweitertem, SQL Schema gespeichert [18].

Auf diese Datenbank greift dann die D-MON Webapplikation, die innerhalb des Gridsphere Portalsystems läuft zu. Entweder durch die Ausführung eines SQL-Selects direkt auf der Datenbank oder idealerweise ueber den Aufruf eines WSRF-Services.

Die eindeutige Identifizierung eines Benutzers soll über sein Grid-Zertifikat erfolgen. Aus diesem sollen die Informationen Vorname, Nachname und Organisationseinheit (OU) ermittelt werden. Da diese Kombination immer nur einmal vorkommen darf, kann ein Nutzer so eindeutig identifiziert werden und nur die Daten, die für ihn und seine VO sowie seine Rolle relevant sind werden zurückgeliefert.

Die Datenbanken auf Site-Ebene sind replizierbar und föderierbar, d.h. es kann zu einem späteren Zeitpunkt eine Replikation einzelner Datenbanken zum Zwecke der Ausfallsicherheit stattfinden. Außerdem kann eine Föderation der Datenbanken realisiert werden, um VO- oder grid-weite Repositories zu schaffen.

6 Referenzen

- [1] Schopf, J. et. al., Monitoring and Discovery in a Web Services Framework: Functionality and Performance of Globus Toolkit MDS4. In: Proceedings of the 15th IEEE International Symposium on High Performance Distributed Computing, IEEE Computer Society Press, Los Alamitos (2006)
- [2] Foster, I. et. al., The Open Grid Services Architecture, Version 1.0, OGSA Working Group, Open Grid Forum (2005)
- [3] Unicore Projektwebseite, <http://www.unicore.eu/>
- [4] A.S. Memon, M.S. Memon, Ph. Wieder, B. Schuller, *CIS: An Information Service based on the Common Information Model*, Proceedings of 3rd IEEE International Conference on e-Science and Grid Computing, Bangalore, India, December, 2007, IEEE Computer Society, ISBN 0-7695-3064-8, pp. 465 - 472
- [5] Globus Toolkit MDS2, Projektwebseite, http://globus.org/toolkit/mds/#mds_gt2
- [6] gLite BDII, Projektwebseite, <http://glite.web.cern.ch/glite/packages/R3.1/deployment/glite-BDII/glite-BDII.asp>
- [7] gLite, Projektwebseite, <http://glite.web.cern.ch/glite/>
- [8] Cooke, A. et al.: The Relational Grid Monitoring Architecture: Mediating Information about the Grid. *J. Grid Comput.* 2(4): 323-339 (2004)
- [9] Pfefferkorn, R-M. et. al.: AMON – a User-Friendly Job Monitoring for the Grid. In: Proceedings of the CoreGRID Symposium, Springer US (2007)
- [10] Höggqvist, M., Röblitz, T., Reinefeld, A.: Stellaris: An RDF-based Information Service for AstroGrid-D, Max Planck Digital Library (2007), <http://edoc.mpg.de/316518>
- [11] RGID / GRDB, Projektwebseite, <http://www.instant-grid.de/?q=en/node/326S>
- [12] Wolf, A.: Spezifikation der D-Grid-Ressourcenbeschreibungssprache D-GRDL und ihrer Nutzung im Grid-Computing, Fraunhofer First, Berlin (2007)
- [13] Grimm, Chr., Pattloch, M. et al.: AAI-Prototyp für die D-Grid Infrastruktur, Fachgebiet 3-4, D-Grid Report (2007)
- [14] Gridsphere, Projektwebseite, <http://www.gridisphere.org>
- [15] Baur, T. and Saad, S.: Customer Service Management for Grid Monitoring and Accounting Data. In: Proceedings of the 18th IFIP/IEEE Workshop on Distributed Operations and Management DSOM, LNCS, San José (2007)
- [16] GridICE, Projektwebseite, <http://gridice.forge.cnaf.infn.it/>
- [17] Androzzi, S. et. al.: GLUE Specification v. 2.0 Draft 24, GLUE Schema Working Group (GLUE) Open Grid Forum (2008), <http://forge.ogf.org/sf/go/doc14639>
- [18] Ehm, F.: GLUE 2.0 SQL Schema@EGEE-Wiki (2008), <https://twiki.cern.ch/twiki/bin/view/EGEE/ExampleGlue20SQL#>