



Tragfähigkeitsnachweis für AP 1.5

Anne Milbert (AEI)
Gevorg Poghosyan (FZK/KIT)
Tibor Kálmán (GWDG)
Tobias Lindinger (LMU)
Mathilde Romberg (FZJ)
Rebecca Breu (FZJ)

15. Juli 2009

INHALTSVERZEICHNIS

1	EINLEITUNG.....	3
2	ABGLEICH DER ANFORDERUNGEN	4
2.1	ROLLEN	4
2.2	GEFORDERTE DATEN	4
2.3	SCHNITTSTELLEN.....	25
2.4	SKALIERBARKEIT.....	25
2.5	KOMMUNIKATIONSMODELL.....	25
2.6	HISTORISCHE DATEN	25
2.7	ANFORDERUNGEN DER HORIZONTALEN INTEGRATION	25
2.8	ANFORDERUNGEN DER VERTIKALEN INTEGRATION	26
3	ZUSAMMENFASSUNG.....	29
4	REFERENZEN.....	70

1 Einleitung

Dieses Dokument realisiert einen Tragfähigkeitsnachweis für das Projekt D-MON, indem die in AP 1.1 erstellten Anforderungen mit den Ergebnissen und Implementierungen des Projektes abgeglichen werden. Dazu werden die in AP1.1 Kapitel 4 erläuterten Anforderungen themenweise überprüft.

Für die horizontale Integration sind zu überprüfen

- der Hierarchielevel, auf dem die Integration stattfindet, dieser soll möglichst niedrig, d.h. beim Ressourcenanbieter, sein;
- die Übersetzungsrichtung der entwickelten Gateways, bidirektional oder unidirektional;
- die Gatewaytypen, d.h. wird in eine gemeinsame Zielsprache übersetzt oder muss von jedem Informationsdienst in das Format jedes anderen übersetzt werden;
- die Modularität der implementierung und damit die Wiederverwendbarkeit der Teile;
- die WS-RF kompatibilität;
- die Eignung des Datenbankschemas.

Bei der vertikalen Integration sind zu überprüfen

- die Verbindung der VO-Information mit den Monitoringdaten, spätestens beim Gateway sollen sie eng verknüpft sein;
- wie VO-bewußt der Monitoring-Dienst ist;
- ob der Dienst stapelbar ist um Lastverteilung und Skalierbarkeit zu erreichen.

Das Monitoring Portal muss das folgende leisten:

- Die Funktionalität muss mindestens die von GridCSM sein, allerdings mit einer übersichtlicheren Darstellung;
- Benutzer müssen authentisiert und autorisiert werden;
- Die Sichten müssen VO-abhängig generiert werden;
- Die Sichten-Auswahl muß VO-abhängig erfolgen.

Bei den unterstützten Sichten ist zu überprüfen

- ob Sichten für VO-Nutzer und –Repräsentanten sowie für ein Grid Operation Centre zur Verfügung stehen;
- ob das GOC alle VO-spezifischen Sichten nachvollziehen kann;
- ob Schnittstellen für Accounting und Billing zur Verfügung stehen.

Für das integrierte Gesamtsystem ist zu überprüfen,

- ob das Gridsphere Portal für Anwender der einzige Zugang zu den Monitoring Daten ist;
- ob es sich beim Monitoring System um einenSammlung kombinierbarer Dienste mit lokalen Monitoring Daten handelt;
- welches Modell eingesetzt wird, push oder pull;
- ob das system skalierbar ist.

Die Ergebnisse dieser Überprüfungen werden in Kapitel 2 erläutert. Kapitel 3 fasst die Ergebnisse abschliessend zusammen. Im Anhang werden die Übersetzungs-Stylesheets gelistet, um die Implementierung der Übersetzer zu verdeutlichen.

2 Abgleich der Anforderungen

2.1 Rollen

Die zu Projektbeginn durchgeführte Anforderungsanalyse fordert die Unterstützung mehrerer Rollen für den Zugriff auf das Portal. Namentlich werden Rollen für ein Grid Operations Center (GOC) sowie für VO-Nutzer und Repräsentanten gefordert.

2.1.1 Grid Operations Center

Diese Rolle wurde für den Vollzugriff auf sämtliche, verfügbare Monitoring-Daten realisiert. Hierfür werden keine VO-Views der Datenbank abgefragt, sondern die Daten der GLUE 2.0 Datenbank direkt. Die Anforderung gilt als erfüllt.

2.1.2 VO-Nutzer

Es existieren sowohl im Portal als auch in den darunterliegenden Komponenten (Datenbank, OGSA-DAI) Sichten, die VO spezifische Views realisieren und Daten anderer VOs ausblenden. Mangels Spezifikation wurde keine Unterscheidung zwischen VO-Nutzern und VO-Repräsentanten implementiert; es besteht jedoch die generelle Möglichkeit im Portal weitere Rollen zu spezifizieren. Der Kontext der VO, in welchem die Monitoring Daten abgefragt bzw. im Portal präsentiert werden soll muss derzeit manuell über einen Parameter im OGSA-DAI Client bzw. eine Auswahlbox im Portal gesetzt werden. Mit der Einführung von SAML-VOMS im D-Grid lässt sich diese Aufgabe jedoch automatisieren.

2.1.3 VO-Repräsentant

Siehe Absatz 2.1.2

2.2 Geforderte Daten

Alle im Monitoring zu berücksichtigenden Daten laut Anforderungsanalyse werden prinzipiell vom GLUE 2.0 Schema unterstützt. Die folgende Tabelle gibt einen Überblick über alle Relationen und Felder der GLUE 2.0 Datenbank sowie welche Middleware welche Informationen liefert. Die Spalte „Requirements“ zeigt die in AP 1.1 geforderten Information, die ein Monitoringsystem in D-Grid bereitstellen sollte. Wie unschwer zu erkennen ist, konnten leider nicht alle Anforderungen erfüllt werden, da die Middleware spezifischen Informationssysteme keine entsprechenden Daten liefern. Die fehlenden Informationen können prinzipiell von weiteren Adaptoren ergänzt werden, sofern die Informationen an sich in anderen Systemen verfügbar sind. Auch dies ist leider nicht immer der Fall, ein Beispiel hierfür ist etwa das Feld EstimatedResponseTime, das wohl nur schwer automatisiert bereitgestellt werden kann. Eine Beschreibung der einzelnen Relationen und Attributen des GLUE 2.0 Schemas befindet sich unter www.oqf.org/documents/GFD.147.pdf.

GLUE 2.0 Tables / Attributes	Type	Null	Key	MDS	CIS	BDII	GRRS	VOMRS	Requirements
AccessPolicy									
servicID	varchar(255)	NO	PRI				✓		
localID	varchar(128)	NO	PRI				✓		
userdomainID	varchar(255)	NO	MUL				✓		
endpointID	varchar(255)	NO	MUL	✓	✓	✓			
scheme	varchar(30)	NO							
informationProvider	varchar(10)	NO	PRI	✓	✓	✓			
sourceAddr	varchar(39)	NO		✓	✓	✓			
insertTime	timestamp	NO		✓	✓	✓			
AdminContact									
adminDomainID	varchar(255)	NO	PRI	✓ MDS:82/100	✓ CIS:/119	✓ BDII:42/70/98/127			✓
localID	varchar(128)	NO	PRI	✓ MDS:83/101	✓ CIS:/120	✓ BDII:38/66/94/123			✓
url	varchar(255)	NO		✓ MDS:84/102	✓ CIS:/121	✓ BDII:40/68/96/125			✓
type	varchar(50)	NO		✓ MDS:85/103	✓ CIS:/122	✓ BDII:39/67/95/124			✓
informationProvider	varchar(10)	NO	PRI	✓ MDS:93/104	✓ CIS:/123	✓ BDII:43/71/99/128			✓
sourceAddr	varchar(39)	NO		✓ MDS:94/105	✓ CIS:/124	✓ BDII:44/72/100/129			✓
insertTime	timestamp	NO		✓ MDS:95/106	✓ CIS:/125	✓ BDII:45/73/101/130			✓
AdminDomain									
ID	varchar(255)	NO	PRI	✓ MDS:34	✓ CIS:/29	✓ BDII:167/209			✓
name	varchar(255)	YES	MUL	✓ MDS:35	✓ CIS:/30	✓ BDII:168/210			✓
description	varchar(255)	YES		✓ MDS:36	✓ CIS:/31	✓ BDII:169/211			✓
distributed	int(1)	YES			✓ CIS:/32	✓ BDII:171/212			✓
adminDomainID	varchar(255)	YES							✓

informationProvider	varchar(10)	NO	PRI	✓	MDS:39	✓	CIS:/33	✓	BDII:172/213			✓
sourceAddr	varchar(39)	NO		✓	MDS:40	✓	CIS:/34	✓	BDII:173/214			✓
insertTime	timestamp	NO		✓	MDS:41	✓	CIS:/35	✓	BDII:174/215			✓
AdminDomainLocation												
adminDomainID	varchar(255)	NO	PRI	✓	MDS:55	✓	CIS:/76	✓	BDII:257			✓
localID	varchar(128)	NO	PRI				✓	CIS:/77	✓	BDII:241		✓
name	varchar(255)	NO		✓	MDS:56	✓	CIS:/78	✓	BDII:242			✓
address	varchar(128)	YES		✓	MDS:57	✓	CIS:/79	✓	BDII:253			✓
place	varchar(64)	YES		✓	MDS:58	✓	CIS:/80	✓	BDII:245/248			✓
country	varchar(32)	YES		✓	MDS:59	✓	CIS:/81	✓	BDII:246/250			✓
postcode	varchar(16)	YES		✓	MDS:60	✓	CIS:/82	✓	BDII:254			✓
longitude	float	YES		✓	MDS:61	✓	CIS:/83	✓	BDII:255			✓
latitude	float	YES		✓	MDS:62	✓	CIS:/84	✓	BDII:256			✓
informationProvider	varchar(10)	NO	PRI	✓	MDS:63	✓	CIS:/85	✓	BDII:258			✓
sourceAddr	varchar(39)	NO		✓	MDS:64	✓	CIS:/86	✓	BDII:259			✓
insertTime	timestamp	NO		✓	MDS:65	✓	CIS:/87	✓	BDII:260			✓
AppEnvExecEnv												
applicationEnvironmentServiceID	varchar(255)	NO	PRI									
applicationEnvironmentLocalID	varchar(255)	NO	PRI									
executionEnvironmentID	varchar(255)	NO	PRI									
informationProvider	varchar(10)	NO	PRI									
sourceAddr	varchar(39)	NO										
insertTime	timestamp	NO										
ApplicationEnvironment												
serviceID	varchar(255)	NO	PRI				✓	CIS:/398				✓
localID	varchar(255)	NO	PRI				✓	CIS:/399				✓
name	varchar(255)	NO					✓	CIS:/400				✓
version	varchar(16)	YES					✓	CIS:/401				✓
repository	varchar(255)	YES										✓
state	varchar(100)	YES										✓
removalDate	datetime	YES										✓
license	varchar(128)	YES										✓

description	varchar(255)	YES							✓
parallelSupport	varchar(16)	YES							✓
maxSlots	int(11)	YES							✓
maxJobs	int(11)	YES							✓
maxUserSeats	int(11)	YES							✓
freeSlots	int(11)	YES							✓
freeJobs	int(11)	YES							✓
freeUserSeats	int(11)	YES							✓
cManagerID	varchar(255)	NO	MUL			✓CIS:/402			✓
informationProvider	varchar(10)	NO	PRI			✓CIS:/403			✓
sourceAddr	varchar(39)	NO				✓CIS:/404			✓
insertTime	timestamp	NO				✓CIS:/405			✓
ApplicationEnvironment_MVA									
endpointID	varchar(255)	NO							
attributeType	int(11)	YES	MUL						
value	varchar(255)	NO							
informationProvider	varchar(10)	NO							
sourceAddr	varchar(39)	NO							
insertTime	timestamp	NO							
ApplicationHandle									
serviceID	varchar(255)	NO	MUL						
localID	varchar(128)	NO							
type	varchar(32)	NO							
value	varchar(255)	NO							
appEnvLocalID	varchar(128)	NO							
informationProvider	varchar(10)	NO							
sourceAddr	varchar(39)	NO							
insertTime	timestamp	NO							
AttributeTypes									
id	int(11)	NO	PRI						
name	varchar(255)	NO	UNI						
informationProvider	varchar(10)	NO	PRI						

sourceAddr	varchar(39)	NO							
insertTime	timestamp	NO							
Benchmark									
parentID	varchar(255)	NO				✓BDII:301/331			✓
parentType	int(11)	NO	MUL			✓BDII:305/335			✓
localID	varchar(128)	NO				✓BDII:304/334			✓
type	varchar(32)	NO				✓BDII:302/332			✓
value	float	NO				✓BDII:303/333			✓
informationProvider	varchar(10)	NO				✓BDII:306/336			✓
sourceAddr	varchar(39)	NO				✓BDII:307/337			✓
insertTime	timestamp	NO				✓BDII:308/338			✓
CService_SService									
localID	varchar(128)	NO							
localPath	varchar(255)	NO							
remotePath	varchar(255)	NO							
cServiceID	varchar(255)	NO	MUL						
sServiceID	varchar(255)	NO	MUL						
informationProvider	varchar(10)	NO							
sourceAddr	varchar(39)	NO							
insertTime	timestamp	NO							
CShareCapacity_MVA									
endpointID	varchar(255)	NO							
attributeType	int(11)	YES	MUL						
value	varchar(255)	NO							
informationProvider	varchar(10)	NO							
sourceAddr	varchar(39)	NO							
insertTime	timestamp	NO							
CShareExecEnv									
computingShareServiceID	varchar(255)	NO	PRI						
computingShareLocalID	varchar(128)	NO	PRI						
executionEnvironmentID	varchar(255)	NO	PRI						

informationProvider	varchar(10)	NO	PRI						
sourceAddr	varchar(39)	NO							
insertTime	timestamp	NO							
ComputingActivity									
serviceID	varchar(255)	NO	MUL						✓
ID	varchar(255)	NO	PRI						✓
name	varchar(255)	YES							✓
type	varchar(32)	YES							✓
idFromEndpoint	varchar(255)	YES							✓
localIdFromManager	varchar(255)	YES							✓
jobDescription	varchar(128)	YES							✓
state	varchar(32)	NO							✓
restartState	varchar(32)	YES							✓
exitCode	int(11)	YES							✓
computingManagerExitCode	varchar(64)	YES							✓
waititingPosition	int(11)	YES							✓
userDomain	varchar(255)	YES							✓
owner	varchar(128)	NO							✓
localOwner	varchar(128)	YES							✓
requestedTotalWallTime	int(11)	YES							✓
requestedTotalCPUTime	int(11)	YES							✓
requestedSlots	int(11)	YES							✓
stdIn	varchar(128)	YES							✓
stdOut	varchar(128)	YES							✓
stdErr	varchar(128)	YES							✓
logDir	varchar(128)	YES							✓
queue	varchar(255)	YES							✓
usedTotalWallTime	int(11)	YES							✓
usedTotalCPUTime	int(11)	YES							✓

sourceAddr	varchar(39)	NO		✓	MDS:248	✓	CIS:/266	✓	BDII:772				
insertTime	timestamp	NO		✓	MDS:249	✓	CIS:/267	✓	BDII:773				
ComputingManager													
serviceID	varchar(255)	NO	MUL	✓	MDS:167	✓	CIS:/293	✓	BDII:437				✓
ID	varchar(255)	NO	PRI	✓	MDS:157	✓	CIS:/294	✓	BDII:432				✓
name	varchar(128)	YES		✓	MDS:159	✓	CIS:/295	✓	BDII:429				✓
type	varchar(64)	NO		✓	MDS:161				BDII:433				✓
version	varchar(16)	YES		✓	MDS:163				BDII:434				✓
reservation	int(1)	YES											✓
bulkSubmission	int(1)	YES											✓
totalPhysicalCPU	int(11)	YES		✓	MDS:165	✓	CIS:/296	✓	BDII:436				✓
totalLogicalCPUs	int(11)	YES					✓	CIS:/297					✓
totalSlots	int(11)	YES											✓
slotsUsedByLocalJobs	int(11)	YES											✓
slotsUsedByGridJobs	int(11)	YES											✓
homogeneity	int(11)	YES											✓
networkInfo	varchar(255)	YES											✓
localCpuDistribution	varchar(255)	YES											✓
workingAreaShared	int(1)	YES											✓
workingAreaTotal	int(11)	YES											✓
workingAreaFree	int(11)	YES											✓
workingAreaLifeTime	int(11)	YES											✓
cacheTotal	int(11)	YES											✓
cacheFree	int(11)	YES											✓
tmpDir	varchar(255)	YES							✓	BDII:431			✓
scratchDir	varchar(255)	YES											✓
applicationDir	varchar(255)	YES							✓	BDII:430			✓
envLocalID	varchar(128)	NO											✓
applicationEnvID	int(11)	NO											✓
informationProvider	varchar(10)	NO	PRI	✓	MDS:169	✓	CIS:/298	✓	BDII:438				✓
sourceAddr	varchar(39)	NO		✓	MDS:170	✓	CIS:/299	✓	BDII:439				✓
insertTime	timestamp	NO		✓	MDS:171	✓	CIS:/300	✓	BDII:440				✓

ComputingManager_MVA									
endpointID	varchar(255)	NO							
attributeType	int(11)	YES	MUL						
value	varchar(255)	NO							
informationProvider	varchar(10)	NO							
sourceAddr	varchar(39)	NO							
insertTime	timestamp	NO							
ComputingMappingPolicy									
userDomainID	varchar(255)	NO	PRI						
serviceID	varchar(255)	NO	PRI						
shareID	varchar(128)	NO	PRI						
localID	varchar(100)	NO	PRI						
scheme	varchar(100)	YES							
rule	varchar(100)	YES							
informationProvider	varchar(10)	NO	PRI						
sourceAddr	varchar(39)	NO							
insertTime	timestamp	NO							
ComputingService									
ID	varchar(255)	NO	PRI	✓	MDS:125	✓	CIS:/156	✓	BDII:382
name	varchar(255)	YES		✓	MDS:127	✓	CIS:/157	✓	BDII:379
type	varchar(128)	NO		✓	MDS:129	✓	CIS:/158	✓	BDII:383
qualityLevel	varchar(32)	NO				✓	CIS:/159	✓	BDII:722
complexity	varchar(64)	YES						✓	BDII:723
totalJobs	int(11)	YES		✓	MDS:134	✓	CIS:/160		
runningJobs	int(11)	YES		✓	MDS:130	✓	CIS:/161		
waitingJobs	int(11)	YES		✓	MDS:132	✓	CIS:/162		
stagingJobs	int(11)	YES							
suspendedJobs	int(11)	YES							
preLRMSWaitingJobs	int(11)	YES							
domainID	varchar(255)	NO	MUL	✓	MDS:136	✓	CIS:/163	✓	BDII:384
informationProvider	varchar(10)	NO	PRI	✓	MDS:139	✓	CIS:/164	✓	BDII:385
sourceAddr	varchar(39)	NO		✓	MDS:140	✓	CIS:/165	✓	BDII:386
insertTime	timestamp	NO		✓	MDS:141	✓	CIS:/166	✓	BDII:387

ComputingShare									
serviceID	varchar(255)	NO	PRI	✓	MDS:313		✓	BDII:508	✓
localID	varchar(128)	NO	PRI	✓	MDS:315		✓	BDII:507	✓
name	varchar(255)	YES					✓	BDII:491	✓
description	varchar(255)	YES					✓	BDII:502	✓
mappingQueue	varchar(128)	YES		✓	MDS:316		✓	BDII:382	✓
maxWallTime	int(11)	YES					✓	BDII:495	✓
maxTotalWallTime	int(11)	YES							✓
minWallTime	int(11)	YES							✓
defaultWallTime	int(11)	YES							✓
maxCPUTime	int(11)	YES					✓	BDII:492	✓
maxTotalCPUTim	int(11)	YES							✓
minCPUTime	int(11)	YES							✓
defaultCPUtime	int(11)	YES							✓
maxTotalJobs	int(11)	YES					✓	BDII:494	✓
maxRunningJobs	int(11)	YES					✓	BDII:493	✓
maxWaitingJobs	int(11)	YES							✓
maxPreLRMSWaitingJobs	int(11)	YES							✓
maxUserRunningJobs	int(11)	YES							✓
maxSlotsPerJobs	int(11)	YES							✓
maxStageInStreams	int(11)	YES							✓
maxStageOutStreams	int(11)	YES							✓
schedulingPolicy	varchar(32)	YES					✓	BDII:497	✓
maxMemory	int(11)	YES							✓
maxDiskSpace	int(11)	YES							✓
preemption	int(1)	YES							✓
servingstate	varchar(32)	NO							✓
totalJobs	int(11)	YES		✓	MDS:318		✓	BDII:503	✓
runningJobs	int(11)	YES		✓	MDS:317		✓	BDII:501	✓
localRunningJobs	int(11)	YES							✓
waitingJobs	int(11)	YES		✓	MDS:319		✓	BDII:504	✓
localWaitingJobs	int(11)	YES							✓
stagingJobs	int(11)	YES							✓
suspendedJobs	int(11)	YES							✓
preLRMSWaitingJobs	int(11)	YES							✓

estimatedAverageWaitingTime	int(11)	YES				✓BDII:498			✓
estimatedWorstWaitingTime	int(11)	YES				✓BDII:505			✓
freeSlots	int(11)	YES				✓BDII:500			✓
freeSlotsWithDuration	int(11)	YES							✓
usedSlots	int(11)	YES							✓
requestedSlots	int(11)	YES							✓
reservationPolicy	varchar(64)	YES							✓
defaultSService	varchar(255)	YES	MUL			✓BDII:509			✓
informationProvider	varchar(10)	NO	PRI	✓	MDS:330	✓BDII:510			✓
sourceAddr	varchar(39)	NO		✓	MDS:331	✓BDII:511			✓
insertTime	timestamp	NO		✓	MDS:332	✓BDII:512			✓
ComputingShare_MVA									
endpointID	varchar(255)	NO							
attributeType	int(11)	YES	MUL						
value	varchar(255)	NO							
informationProvider	varchar(10)	NO							
sourceAddr	varchar(39)	NO							
insertTime	timestamp	NO							
Endpoint									
serviceID	varchar(255)	NO	MUL	✓	✓CIS:/219	✓BDII:676/735			✓
ID	varchar(255)	NO	PRI	✓	✓CIS:/220	✓BDII:671/736			✓
name	varchar(255)	YES			✓CIS:/221	✓BDII:672/737			✓
url	varchar(255)	NO			✓CIS:/222	✓BDII:673/738			✓
technology	varchar(255)	NO			✓CIS:/223	✓BDII:674/739			✓
interface	varchar(255)	NO			✓CIS:/224				✓
implementor	varchar(255)	YES			✓CIS:/225				✓
implementationName	varchar(255)	YES			✓CIS:/226				✓
implementationVersion	varchar(255)	YES			✓CIS:/227	✓BDII:675/740			✓
qualityLevel	varchar(32)	NO			✓CIS:/228				✓
healthState	varchar(32)	NO			✓CIS:/229	✓BDII:741			✓
healthStateInfo	varchar(255)	YES			✓CIS:/230	✓BDII:742			✓
servingState	varchar(255)	NO			✓CIS:/231				✓
startTime	timestamp	NO				✓BDII:743			✓

issuerCA	varchar(255)	NO							✓
downTimeAnnounce	timestamp	NO							✓
downTimeStart	timestamp	NO							✓
downTimeEnd	timestamp	NO							✓
downTimeInfo	varchar(255)	YES							✓
informationProvider	varchar(10)	NO	PRI	✓	✓ CIS:/231	✓ BDII:677/744			✓
sourceAddr	varchar(39)	NO		✓	✓ CIS:/232	✓ BDII:678/745			✓
insertTime	timestamp	NO		✓	✓ CIS:/233	✓ BDII:679/746			✓
EndpointShare_LNK									
endpointID	varchar(255)	NO	PRI						
shareID	varchar(128)	NO	PRI						
serviceID	varchar(255)	NO	PRI						
informationProvider	varchar(10)	NO	PRI						
sourceAddr	varchar(39)	NO							
insertTime	timestamp	NO							
Endpoint_MVA									
endpointID	varchar(255)	NO							
attributeType	int(11)	YES	MUL						
value	varchar(255)	NO							
informationProvider	varchar(10)	NO							
sourceAddr	varchar(39)	NO							
insertTime	timestamp	NO							
EntryTypes									
id	int(11)	NO	PRI			✓ BDII:280			
name	varchar(255)	NO	UNI						
informationProvider	varchar(10)	NO	PRI			✓ BDII:281			
sourceAddr	varchar(39)	NO				✓ BDII:281			
insertTime	timestamp	NO				✓ BDII:282			
ExecutionEnvironment									
serviceID	varchar(255)	NO		✓ MDS:272	✓ CIS:349	✓ BDII:596			✓
ID	varchar(255)	NO	PRI	✓ MDS:269	✓ CIS:350	✓ BDII:575			✓

platform	varchar(32)	NO			✓ CIS:351				✓
virtualMachine	int(11)	YES			✓ CIS:352				✓
totalInstances	int(11)	YES			✓ CIS:353				✓
usedInstances	int(11)	YES							✓
unavailableInstances	int(11)	YES							✓
physicalCPUs	int(11)	YES			✓ CIS:354	✓ BDII:594			✓
logicalCPUs	int(11)	YES			✓ CIS:355	✓ BDII:595			✓
cpuMultiplicity	varchar(128)	YES	✓	MDS:284					✓
cpuVendor	varchar(32)	YES				✓ BDII:593			✓
cpuModel	varchar(64)	YES				✓ BDII:592			✓
cpuVersion	varchar(16)	YES							✓
cpuClockSpeed	int(11)	YES	✓	MDS:274		✓ BDII:591			✓
cpuTimeScalingFactor	float	YES							✓
wallTimeScalingFactor	float	YES							✓
mainMemorySize	int(11)	NO	✓	MDS:276	✓ CIS:356	✓ BDII:576			✓
virtualMemorySize	int(11)	YES	✓	MDS:279		✓ BDII:578			✓
osFamily	varchar(32)	NO			✓ CIS:357	✓ BDII:586			✓
osName	varchar(64)	YES	✓	MDS:280	✓ CIS:358	✓ BDII:584			✓
osVersion	varchar(16)	YES	✓	MDS:282		✓ BDII:588			✓
connectivityIn	int(11)	NO				✓ BDII:580			✓
connectivityOut	int(11)	NO				✓ BDII:582			✓
cManagerID	varchar(255)	NO	MUL	✓	MDS:270	✓ CIS:359	✓ BDII:597		✓
informationProvider	varchar(10)	NO	PRI	✓	MDS:286	✓ CIS:360	✓ BDII:598		✓
sourceAddr	varchar(39)	NO		✓	MDS:287	✓ CIS:361	✓ BDII:599		✓
insertTime	timestamp	NO		✓	MDS:288	✓ CIS:362	✓ BDII:600		✓
ExecutionEnvironment_MVA									
endpointID	varchar(255)	NO							
attributeType	int(11)	YES	MUL						
value	varchar(255)	NO							
informationProvider	varchar(10)	NO							
sourceAddr	varchar(39)	NO							
insertTime	timestamp	NO							
MappingPolicy									

userDomainID	varchar(255)	NO	PRI						
shareID	varchar(128)	NO	PRI						
serviceID	varchar(255)	NO	PRI						
scheme	varchar(30)	NO	PRI						
Rule	varchar(128)	YES							
informationProvider	varchar(10)	NO	PRI						
sourceAddr	varchar(39)	NO							
insertTime	timestamp	NO							
SAccessPolicy_MVA									
endpointID	varchar(255)	NO							
attributeType	int(11)	YES	MUL						
value	varchar(255)	NO							
informationProvider	varchar(10)	NO							
sourceAddr	varchar(39)	NO							
insertTime	timestamp	NO							
SShareCapacity_MVA									
endpointID	varchar(255)	NO							
attributeType	int(11)	YES	MUL						
value	varchar(255)	NO							
informationProvider	varchar(10)	NO							
sourceAddr	varchar(39)	NO							
insertTime	timestamp	NO							
SShareSResource									
storageShareServiceID	varchar(255)	NO	PRI						
storageShareLocalID	varchar(128)	NO	PRI						
storageResourceID	varchar(255)	NO	PRI						
informationProvider	varchar(10)	NO	PRI						

sourceAddr	varchar(39)	NO							
insertTime	timestamp	NO							
StorageAccessProtocol									
serviceID	varchar(255)	NO	PRI						
localID	varchar(128)	NO	PRI						
type	varchar(16)	NO							
version	varchar(20)	YES							
maxStreams	int(11)	YES							
informationProvider	varchar(10)	NO	PRI						
sourceAddr	varchar(39)	NO							
insertTime	timestamp	NO							
StorageEndpoint									
endpointID	varchar(255)	NO	PRI			✓			✓
sServiceID	varchar(255)	NO	PRI			✓			✓
informationProvider	varchar(10)	NO	PRI						✓
sourceAddr	varchar(39)	NO							✓
insertTime	timestamp	NO							✓
StorageManager									
ID	varchar(255)	NO	PRI						✓
type	varchar(64)	NO							✓
version	varchar(16)	YES							✓
sServiceID	varchar(255)	NO	MUL						✓
informationProvider	varchar(10)	NO	PRI						✓
sourceAddr	varchar(39)	NO							✓
insertTime	timestamp	NO							✓
StorageManager_MVA									
endpointID	varchar(255)	NO							
attributeType	int(11)	YES	MUL						
value	varchar(255)	NO							
informationProvider	varchar(10)	NO							
sourceAddr	varchar(39)	NO							

value	varchar(255)	NO							
informationProvider	varchar(10)	NO							
sourceAddr	varchar(39)	NO							
insertTime	timestamp	NO							
StorageService									
ID	varchar(255)	NO	PRI			✓BDII:644			✓
name	varchar(255)	YES				✓BDII:645			✓
type	varchar(128)	NO							✓
qualityLevel	varchar(32)	NO							✓
complexity	varchar(64)	YES							✓
domainID	varchar(255)	NO	MUL			✓BDII:646			✓
locationID	int(11)	YES							✓
informationProvider	varchar(10)	NO	PRI			✓BDII:647			✓
sourceAddr	varchar(39)	NO				✓BDII:648			✓
insertTime	timestamp	NO				✓BDII:649			✓
StorageServiceCapacity									
localID	varchar(128)	NO	PRI						✓
serviceID	varchar(255)	NO	PRI						✓
type	varchar(64)	NO							✓
totalSize	int(11)	YES							✓
usedSize	int(11)	YES							✓
freeSize	int(11)	YES							✓
reservedSize	int(11)	YES							✓
informationProvider	varchar(10)	NO	PRI						✓
sourceAddr	varchar(39)	NO							✓
insertTime	timestamp	NO							✓
StorageServiceLoc									
storageServiceID	varchar(255)	NO	PRI						✓

localID	varchar(128)	NO	PRI						✓
name	varchar(255)	NO							✓
address	varchar(128)	YES							✓
place	varchar(64)	YES							✓
country	varchar(32)	YES							✓
postcode	varchar(16)	YES							✓
longitude	float	YES							✓
latitude	float	YES							✓
informationProvider	varchar(10)	NO	PRI						✓
sourceAddr	varchar(39)	NO							✓
insertTime	timestamp	NO							✓
StorageService_MVA									
endpointID	varchar(255)	NO							
attributeType	int(11)	YES	MUL						
value	varchar(255)	NO							
informationProvider	varchar(10)	NO							
sourceAddr	varchar(39)	NO							
insertTime	timestamp	NO							
StorageShare									
serviceID	varchar(255)	NO	PRI						
localID	varchar(128)	NO	PRI						
name	varchar(255)	YES							
description	varchar(255)	YES							
servingState	varchar(32)	NO							
path	varchar(255)	YES							
sharingID	varchar(128)	NO							
accessLatency	varchar(64)	NO							
defaultLifeTime	int(11)	YES							
maximumLifeTime	int(11)	YES							
tag	varchar(255)	YES							
informationProvider	varchar(10)	NO	PRI						
sourceAddr	varchar(39)	NO							
insertTime	timestamp	NO							

StorageShareCapacity									
localID	varchar(128)	NO							
serviceID	varchar(255)	NO	PRI						
shareID	varchar(255)	NO	PRI						
type	varchar(64)	NO							
totalSize	int(11)	YES							
usedSize	int(11)	YES							
freeSize	int(11)	YES							
reservedSize	int(11)	YES							
informationProvider	varchar(10)	NO	PRI						
sourceAddr	varchar(39)	NO							
insertTime	timestamp	NO							
StorageShare_MVA									
endpointID	varchar(255)	NO							
attributeType	int(11)	YES	MUL						
value	varchar(255)	NO							
informationProvider	varchar(10)	NO							
sourceAddr	varchar(39)	NO							
insertTime	timestamp	NO							
UserContact									
userDomainID	varchar(255)	NO	PRI						
localID	varchar(128)	NO	PRI						
url	varchar(255)	NO							
type	varchar(50)	NO							
informationProvider	varchar(10)	NO	PRI						
sourceAddr	varchar(39)	NO							
insertTime	timestamp	NO							
UserDomain									
ID	varchar(255)	NO	PRI				✓	✓	
name	varchar(255)	YES	MUL					✓	
description	varchar(255)	YES							

level	int(11)	YES							
userDomainID	varchar(255)	YES							
informationProvider	varchar(10)	NO	PRI				✓	✓	
sourceAddr	varchar(39)	NO					✓	✓	
insertTime	timestamp	NO					✓	✓	
							✓	✓	
UserDomainLocation									
userDomainID	varchar(255)	NO	PRI						
localID	varchar(128)	NO	PRI						
name	varchar(255)	NO							
address	varchar(128)	YES							
place	varchar(64)	YES							
country	varchar(32)	YES							
postcode	varchar(16)	YES							
longitude	float	YES							
latitude	float	YES							
informationProvider	varchar(10)	NO	PRI						
sourceAddr	varchar(39)	NO							
insertTime	timestamp	NO							
V_CManagerBenchmark									
parentID	varchar(255)	NO							
parentType	int(11)	NO							
localID	varchar(128)	NO							
type	varchar(32)	NO							
value	float	NO							
V_ExecEnvBenchmark									
parentID	varchar(255)	NO							
parentType	int(11)	NO							
localID	varchar(128)	NO							
type	varchar(32)	NO							
value	float	NO							

2.3 Schnittstellen

Das System besitzt eine WS-RF fähige Schnittstelle (OGSA-DAI) über die alle Daten der Datenbank VO-spezifisch abgefragt werden können. Ein entsprechender OGSA-DAI Client wurde während des Projektes entwickelt und steht zur Verfügung.

2.4 Skalierbarkeit

Ein wichtiger Aspekt beim Entwurf der Monitoring Architektur war es, das System skalierbar zu gestalten. Diese Anforderung wurde dadurch erfüllt, dass die Datenbank zum einen ein Caching realisiert und kleine Verzögerungen in der Aktualität in Kauf genommen werden sowie zum anderen dadurch, dass es prinzipiell möglich ist mehrere Datenbanken hierarchisch übereinander zu stapeln. Das System ähnelt dann in seinem Aufbau der bestehenden MDS4 Hierarchie im D-Grid, kommt aber mit einer geringeren Anzahl an Hierarchiestufen aus. Die vorgehaltenen Daten sind daher aktueller als im bestehenden System.

2.5 Kommunikationsmodell

Als Kommunikationsmodell wurde wie gefordert eine Mischung aus Push- und Pull Modell implementiert. Ein zentraler Adapter speist Daten mittels eines Push-Verfahrens in die Datenbank, von wo sie bei Bedarf durch ein Pull-Verfahren abgerufen werden können. Ein Hauptgrund für diese Vorgehensweise liegt in der großen Anzahl der überwachten Systeme und der Anforderung eine möglichst flache Hierarchie zu schaffen, um Änderungen von Werten möglichst schnell propagieren zu können. Das Kommunikationsmodell ist daher implizit nötig, um Skalierbarkeit (siehe 2.4) zu gewährleisten.

2.6 Historische Daten

Die Möglichkeit Daten des Monitorings zu archivieren wurde mit der Wahl einer Datenbank geschaffen. Über einen Triggermechanismus, der zur Zeit zwar noch nicht implementiert ist aber jederzeit realisiert werden kann, können ausgewählte Daten bei Ereignissen BeforeDelete und BeforeUpdate gesichert werden. Sinnvoll erscheint dieser Mechanismus hauptsächlich bei Accounting Daten, die zur Zeit jedoch noch nicht im System enthalten sind, da keine entsprechenden Daten durch die Middleware spezifischen Monitoringsysteme bereitgestellt werden.

2.7 Anforderungen der horizontalen Integration

Ein Hauptaspekt des Projektes war die Integration des Monitorings über mehrere Middleware Systeme, ohne ein spezielles Monitoring System zu bevorzugen. Diese Anforderung wurde mit der Instanziierung einer für alle Middleware Instanzen über OGSA-DAI zugreifbaren Datenbank realisiert. Die Anforderung das System bidirektional zu implementieren konnte nicht erfüllt werden. Der Grund hierfür lag darin, dass die Middleware spezifischen Monitoringsysteme wie auch deren Administratoren in den verschiedenen Systemen syntaktisch unterschiedliche Bezeichner für dieselben Ressourcen verwenden, so dass eine automatisierte Zuordnung und Rückübersetzung derzeit nicht möglich ist. Ein Beispiel hierfür ist etwa die unterschiedliche Benennung von Services in den Middlewares, die aufgrund ihrer Verschattung durch einen middleware spezifischen Zugriffspunkt und unterschiedliche Namen nicht als ein einzelner Service identifiziert werden können, sondern jeweils als eigenständiger Service mit eigenen Ressourcen. Die Anforderung der Rückübersetzung könnte jedoch verhältnismäßig einfach durch zusätzliche Adaptern realisiert

werden, wenn die genannten Probleme durch eine D-Grid weite Richtlinie zur Bezeichnung von Ressourcen und Diensten, welche auf dem Papier existiert, in der Praxis realisiert wurde. Auch die Anforderung keine vollständig besetzte Matrix aus Gateways und Adaptoren zu implementieren und damit die Wartbarkeit des Systems zu erhöhen wurde mit der Wahl einer Datenbank mit GLUE 2.0 Schema als Ziel der Übersetzung erfüllt.

2.8 Anforderungen der vertikalen Integration

Der zweite Aspekt des D-MON Projektes ist die Bildung von VO-spezifischen Sichten auf das D-Grid Monitoring. Diese Anforderung wurde erfüllt, lediglich die in 2.1.2 diskutierte Unterscheidung von VO-Nutzern und VO-Repräsentanten wurde mangels einer Spezifikation dieser Sichten und fehlenden Daten im Monitoring verzichtet. Prinzipiell ist das Portal jedoch in der Lage Sichten für gewisse Rollen einzuschränken.

2.9 Anforderungen an das Monitoring Portal

Das Monitoring Portal des D-MON Projekts sollte neben der Implementierung der in Abschnitt x.y bereits ausführlich behandelten Rollen eines jeden Benutzers, mindestens die Funktionalität der Webseite GridCSM¹ erreichen. Die Struktur und die Uebersichtlichkeit des D-MON Portals sollte allerdings die von GridCSM uebersteigen.

GridCSM bietet unter dem Menüpunkt Topologie eine mithilfe des Informationsproviders GeoMaint dynamisch generierte Landkarte (GoogleMap) an, auf der alle Ressourcenanbieter mit dem derzeitigen Wartungszustand der jeweiligen Resource abgebildet sind. Die auf der Karte abgebildeten Marker sind klickbar und bieten bei Mausclick Detailinformationen zu den Ressourcen an. Der Menüpunkt Drill-Down bietet eine tabellarische Uebersicht ueber Web-Services und/oder WS-Ressourcen im D-Grid an, die Mitglieder der WS-Service Group sind. Es gibt die Moeglichkeit zum Abruf von Detailinformationen zu jedem Menüpunkt.

Diese Funktionalität wurde im Rahmen des D-MON Monitoring Portals vollstaendig implementiert und um diverse Aspekte erweitert. Bei Auswahl einer Tabelle im Portlet „D-MON GLUE 2.0 Monitoring Information Database“, die eine assoziierte sogenannte Location Tabelle hat, werden dynamisch die Informationen aus dieser Tabelle aus der Datenbank geladen und die entsprechenden Marker auf der Landkarte angezeigt. Momentan gibt es damit die Moeglichkeit Informationen aus den folgenden Tabellen anzuzeigen: AdminDomainLocation, ComputingServiceLoc, StorageServiceLoc und UserDomainLocation. Sollten in der Zukunft neue Tabellen hinzukommen wuerden auch diese dynamisch in die Landkarte geladen, ohne dass die Logik der Web-Applikation veraendert werden muss. Vorausgesetzt diese folgen den vereinbarten Namenskonventionen (Tabellenname + „Loc“ bzw. Tabellenname + „Location“). Zudem wird bei Klick auf eines der verlinkten Tabellenfelder im Portlet „D-MON GLUE 2.0 Monitoring Information Database“ automatisch ein Hereinzoomen auf den passenden Marker durchgefuehrt, falls dieser vorhanden ist und die Detailinformationen angezeigt.

GridCSM bietet ferner die Moeglichkeit einer tabellarischen Uebersicht ueber alle verfuegbaren Computingelemente, Job Queues und die Anzahl der vom jeweiligen Batchsystem gemeldeten Jobs an. Zusaetzlich wird durch eine farbliche Hervorhebung (CE hat Jobs in der Queue, CE ist bereit, CE ist abgeschaltet oder in Wartung) die Uebersichtlichkeit der Datensatze gesteigert.

¹ <http://webmds.lrz-muenchen.de:8080/webmds/xslfiles/csm/>

Ferner bietet es eine tabellarische Uebersicht von Clustern (Clusterliste) an, die alle Cluster und Workernodes zeigt, die Daten an das Monitoring melden.

Das D-MON Monitoring Portal hat diese Funktionalitaet, abgesehen von der Verwendung der farblichen Hervorhebung der Datensaeetze uebernommen. Insofern die Daten in der Datenbank vorhanden sind, koennen sie durch das Portal auch angezeigt werden.

Die Uebersichtlichkeit wurde der Daten durch die folgenden Features erheblich erhoehrt. Das Portlet „D-MON GLUE 2.0 Monitoring Information Database“ enthaelt eine Auswahlliste aller in der D-MON Datenbank enthaltenen Daten². Bei Auswahl einer Tabelle aus dieser Liste werden die in dieser Tabelle enthaltenen Daten dynamisch in die sich darunter befindende Tabelle geladen. Tabellenspalten, die nur NULL-Values enthalten werden automatisch herausgefiltert. Die Spalten der generierten Tabelle sind sortierbar, die Zeilen klickbar und werden bei Mausklick farblich hervorgehoben.

Der Zugriff auf Detailinformationen zu denen in diesem Portlet angezeigten Daten ist durch die Verlinkung von diversen Feldinhalten moeglich. Beim Klick auf eines dieser verlinkten Felder einer Tabellenzeile, Detailinformationen fuer diesen Feldwert in das Portlet „Details“ geladen. Diese Portlet verfuegt abgesehen von der Verlinkung von Feldwerten ueber dieselben Funktionalitaeten wie das Portlet „D-MON GLUE 2.0 Monitoring Information Database“ (Highlighting von Tabellenzeilen, Sortierbarkeit der Spalten, Auswahlliste von Tabellen die Detailinformationen zu dem entsprechenden Feldwert enthalten).

Der Menüpunkt Jawari Benchmarking auf der GridCSM Seite integriert den Jawari Benchmarking Service innerhalb eines Framesets.

Diese Funktionalitaet wird in Ermangelung entsprechender Informationen nur teilweise abgebildet. Die D-MON Datenbank verfuegt zwar ueber eine Tabelle „Benchmark“, diese enthaelt jedoch nur anteilig die Informationen, die durch das bei GridCSM verwendete Tool aggregiert werden.

Der Benutzer der Seite, abgesehen vom Benutzer mit der Rolle GUEST (nicht authentifizierbar bzw. nicht angemeldet) sollte authentifiziert (AuthN) und autorisiert (AuthZ) werden. Die ursprünglich im Projekt-Arbeitsplan im Arbeitspaket AP-1.2 „*Middleware-übergreifendes Monitoring: Evaluierung und Auswahl von Komponenten*“ angedachte Integration eines SAML-VOMS (Security Assertion Markup Language – Virtual Organisation Management System) Authentifizierungsmoduls fuer Gridsphere konnte aus folgenden Gruenden nicht durchgefuehrt werden: Der Benutzer sollte mittels seines Grid Zertifikats eindeutig identifiziert und ihm Datensätze aus der dmon Datenbank eindeutig zugeordnet werden können. Diese Funktionalitaet konnte nicht bereitgestellt werden, da bis dato im D-Grid die benutzerbezogenen Daten aus verschiedenen Gruenden nicht durch die Middlewares gesammelt werden. Der in AP-1.3 „*Softwaretechnische Analyse und Design*“ definierte rollenbasierten Zugriff auf die Daten konnte durch die Implementierung eines solchen Authentifizierungsmoduls ebenfalls nicht gewaehrleistet werden, da diese derzeit weder durch SAML-VOMS erfasst, noch im Grid-Zertifikat abgespeichert werden. Aus diesem Grund wurde die Umsetzung des Authentifizierungsmoduls vorerst auf Eis gelegt bis die Rahmenbedingungen dessen Einsatz moeglich machen.

Aus den im vorhergehenden Absatz angefuehrten Gruenden wurde die Authentifizierung und Autorisierung der Nutzer ueber den Core des Gridsphere Frameworks realisiert, sprich ein einfacher Login mit Benutzername und Passwort realisiert. Die Moeglichkeit zur Registrierung

² Die angezeigten Tabellen sind abhaengig von der Rolle des jeweiligen Benutzers

neuer Nutzer ist durch eine built-in Registrierungsfunctionalitaet von Gridsphere implementiert worden. Die Assoziierung der Rollen zu den jeweiligen Benutzern wurde ueber die Gridsphere Administrationsoberflaeche durchgefuehrt.

Eine der Kernfunctionalitaeten der Portal Seiten sollte die Moeglichkeit des Zugriffs auf die von der Datenbank uebermittelten VO-basierten Sichten auf die aggregierten Daten sein. Diese Funktionalitaet sollte bei Mitgliedschaft eines Nutzers in mehr als einer VO auch die Moeglichkeit zum Wechsel von einer VO-View in die andere, ohne erneuten Login des Benutzers gewaehrleisten. Die Informationen zur VO-Zugehoerigkeit eines Nutzers werden im Portlet „User Information“ zusammen mit dessen Benutzerdaten in einer Auswahlliste angezeigt.

Die von GridCSM angebotene Moeglichkeit eines XML-Dumps wurde in Absprache mit den anderen Projekt-Mitgliedern nicht realisiert, da sie als obsolet angesehen wurde.

3 Zusammenfassung

Aufgrund der in Absatz 2.7 genannten Probleme, die ein automatisches Zurückführen der horizontal integrierten Monitoring Daten in die nativen Monitoring Systeme verhindern, musste der ursprüngliche Projektplan, welcher nur Übersetzer zwischen den Middleware Monitoring Systemen vorsah, geändert werden und ein separater Dienst zum Sammeln implementiert werden. Funktional sind jedoch alle wesentlichen Punkte des Projektantrages sowie der Anforderungen aus AP 1.1 erfüllt. Kleinere nicht erfüllte Aspekte wurden in diesem Dokument explizit erwähnt und stellen keine Beeinträchtigung im für den produktiven Einsatz dar.

4 Anhang

Die folgenden Stylesheets realisieren die Abbildung der Information in den Grid Middleware Informationssystemen in die GLUE 2.0 Datenbank. Da aufgrund der komplex geschachtelten Inhalte eine Darstellung in Tabellenform nicht oder nur sehr unzureichend möglich wäre, wird das Mapping der Informationen im XSLT –Format dokumentiert, so wie diese in den Adaptoren realisiert wurden.

4.1 Globus MDS4 Stylesheet

```
1 <?xml version="1.0" encoding="iso-8859-1"?>
2 <xsl:stylesheet version="1.0"
3 xmlns:xsl="http://www.w3.org/1999/XSL/Transform"
4 xmlns:html="http://www.w3.org/1999/xhtml"
5 xmlns="http://www.w3.org/1999/xhtml"
6 xmlns:ns0="http://mds.globus.org/index"
7 xmlns:ns1="http://mds.globus.org/glue/ce/1.1"
8 xmlns:ns11="http://docs.oasis-open.org/wsrp/2004/06/wsrp-WS-ResourceLifetime-
9 1.2-draft-01.xsd"
10 xmlns:geomaint="http://infonforge.cnaf.infn.it/glueinfomodel/Spec/V12/R2"
11 xmlns:ns="http://infonforge.cnaf.infn.it/glueinfomodel/Spec/V12/R2" exclude-
12 result-prefixes="html"
13 >
14
15 <xsl:output method="text" encoding="ISO-8859-1"/>
16
17 <xsl:variable name="informationProvider">MDS4</xsl:variable>
18 <xsl:param name="sourceAddress" select="'wsrfclient'"/>
19 <xsl:variable name="insertTime">CURRENT_TIMESTAMP</xsl:variable>
20
21 <xsl:template match="/">
22     <xsl:apply-templates select="//ns:Site"/>
23     <xsl:apply-templates select="//ns1:GLUECE"/>
24     <xsl:apply-templates select="//ns1:SubCluster/ns1:Host"/>
25     <xsl:apply-templates select="//ns1:ComputingElement"/>
26 </xsl:template>
27
28
29 <xsl:template match="//ns:Site">
30
31     INSERT INTO
32 GLUE20.AdminDomain(ID,name,description,distributed,adminDomainID,informationP
33 rovider,sourceAddr,insertTime) VALUES (
34     "<xsl:value-of select="./@UniqueID"/>",
35     "<xsl:value-of select="ns:Name/text()"/>",
36     "<xsl:value-of select="ns:Description/text()"/>",
37     "0",
38     "<xsl:value-of select="./@UniqueID"/>",
39     "<xsl:value-of select=".$informationProvider"/>",
40     "<xsl:value-of select=".$sourceAddress"/>",
41     <xsl:value-of select=".$insertTime"/>)
42     ON DUPLICATE KEY UPDATE
43     ID = VALUES(ID),
44     name = VALUES(name),
45     description = VALUES(description),
46     distributed = VALUES(distributed),
47     adminDomainID = VALUES (adminDomainID),
48     informationProvider = VALUES(informationProvider),
49     sourceAddr = VALUES(sourceAddr),
50     insertTime = VALUES(insertTime);
51
```

```

52         INSERT INTO GLUE20.AdminDomainLocation
53 (adminDomainID,name,place,country,address,postcode,longitude,latitude,informa
54 tionProvider,sourceAddr,insertTime) VALUES (
55     "<xsl:value-of select="./@UniqueID"/>",
56     "<xsl:value-of select="ns:Name/text()"/>",
57     "<xsl:value-of select="ns:Location/text()"/>",
58     "<xsl:value-of select="ns:Location/text()"/>",
59     "<xsl:value-of select="ns:Location/text()"/>",
60     "<xsl:value-of select="ns:Location/text()"/>",
61     "<xsl:value-of select="ns:Longitude/text()"/>",
62     "<xsl:value-of select="ns:Latitude/text()"/>",
63     "<xsl:value-of select=".$informationProvider"/>",
64     "<xsl:value-of select=".$sourceAddress"/>",
65     <xsl:value-of select=".$insertTime"/>)
66     ON DUPLICATE KEY UPDATE
67     adminDomainID = VALUES(adminDomainID),
68     name = VALUES(name),
69     place = VALUES(place),
70     country = VALUES(country),
71     address = VALUES(address),
72     postcode = VALUES(postcode),
73     longitude = VALUES(longitude),
74     latitude = VALUES(latitude),
75     informationProvider = VALUES(informationProvider),
76     sourceAddr = VALUES(sourceAddr),
77     insertTime = VALUES(insertTime);
78
79     INSERT INTO GLUE20.AdminContact
80 (adminDomainID,localID,url,type,informationProvider,sourceAddr,insertTime)
81 VALUES (
82     "<xsl:value-of select="./@UniqueID"/>",
83     "sysadmin",
84     "<xsl:value-of select="./ns:SysAdminContact/text()"/>",
85     "sysadmin",
86     "<xsl:value-of select=".$informationProvider"/>",
87     "<xsl:value-of select=".$sourceAddress"/>",<xsl:value-of
88 select=".$insertTime"/>)
89     ON DUPLICATE KEY UPDATE
90     adminDomainID = VALUES(adminDomainID),
91     localID = VALUES(localID),
92     url = VALUES(url),
93     informationProvider = VALUES(informationProvider),
94     sourceAddr = VALUES(sourceAddr),
95     insertTime = VALUES(insertTime);
96
97     INSERT INTO GLUE20.AdminContact
98 (adminDomainID,localID,url,type,informationProvider,sourceAddr,insertTime)
99 VALUES (
100     "<xsl:value-of select="./@UniqueID"/>",
101     "web",
102     "<xsl:value-of select="./ns:Web/text()"/>",
103     "web",
104     "<xsl:value-of select=".$informationProvider"/>",
105     "<xsl:value-of select=".$sourceAddress"/>",
106     <xsl:value-of select=".$insertTime"/>)
107     ON DUPLICATE KEY UPDATE
108     adminDomainID = VALUES(adminDomainID),

```

```

109         localID = VALUES(localID),
110         url = VALUES(url),
111         informationProvider = VALUES(informationProvider),
112         sourceAddr = VALUES(sourceAddr),
113         insertTime = VALUES(insertTime);
114
115 </xsl:template>
116
117
118 <xsl:template match="//ns1:GLUECE">
119
120     <xsl:if test="not(count(./node()[local-
121 name()='ComputingElement'][1]/node()[local-name()='Info']/@ns1:HostName)=0)">
122         INSERT INTO GLUE20.ComputingService
123 (ID,name,type,runningJobs,waitingJobs,totalJobs,domainID,informationProvider,
124 sourceAddr,insertTime) VALUES (
125     "<xsl:value-of select="./node()[local-
126 name()='ComputingElement'][1]/node()[local-name()='Info']/@ns1:HostName"/>",
127     "<xsl:value-of select="./node()[local-
128 name()='ComputingElement'][1]/node()[local-name()='Info']/@ns1:HostName"/>",
129     "org.teragrid.ws-gram",
130     "<xsl:value-of
131 select='sum(./ns1:ComputingElement/ns1:State/@ns1:RunningJobs)'/>",
132     "<xsl:value-of
133 select='sum(./ns1:ComputingElement/ns1:State/@ns1:WaitingJobs)'/>",
134     "<xsl:value-of
135 select='sum(./ns1:ComputingElement/ns1:State/@ns1:TotalJobs)'/>",
136     "<xsl:value-of select='./ancestor::node()[local-
137 name()='Entry']/following-sibling::node()[local-
138 name()='Site'][1]/@UniqueID"/>",
139     "<xsl:value-of select='$informationProvider'"/>",
140     "<xsl:value-of select='$sourceAddress'"/>",
141     "<xsl:value-of select='$insertTime'"/>)
142     ON DUPLICATE KEY UPDATE
143     ID = VALUES(ID),
144     runningJobs = VALUES(runningJobs),
145     waitingJobs =VALUES(WaitingJobs),
146     totalJobs = VALUES(totalJobs),
147     name = VALUES(ID),
148     type = VALUES(type),
149     domainID = VALUES(domainID),
150     informationProvider = VALUES(informationProvider),
151     sourceAddr = VALUES(sourceAddr),
152     insertTime = VALUES(insertTime);
153
154     INSERT INTO GLUE20.ComputingManager
155 (ID,name,type,version,totalPhysicalCPU,serviceID,informationProvider,sourceAd
156 dr,insertTime) VALUES (
157     "<xsl:value-of select="./node()[local-
158 name()='ComputingElement'][1]/node()[local-name()='Info']/@ns1:HostName"/>",
159     "<xsl:value-of select="./node()[local-
160 name()='ComputingElement'][1]/node()[local-name()='Info']/@ns1:HostName"/>",
161     "<xsl:value-of
162 select='./ns1:ComputingElement[1]/ns1:Info/@ns1:LRMSType'"/>",
163     "<xsl:value-of
164 select='./ns1:ComputingElement[1]/ns1:Info/@ns1:LRMSVersion'"/>",

```

```

165         <xsl:value-of
166 select=" ./ns1:ComputingElement[1]/ns1:Info/@ns1:TotalCPUs"/> ",
167         "<xsl:value-of select=" ./node()[local-
168 name()='ComputingElement'][1]/node()[local-name()='Info']/@ns1:HostName"/> ",
169         "<xsl:value-of select=\"$informationProvider\"/> ",
170         "<xsl:value-of select=\"$sourceAddress\"/> ",
171         <xsl:value-of select=\"$insertTime\"/>)
172 ON DUPLICATE KEY UPDATE
173 ID = VALUES(ID),
174 name = VALUES(name),
175 type = VALUES(type),
176 version = VALUES(version),
177 totalPhysicalCPU = VALUES(totalPhysicalCPU),
178 serviceID = VALUES(serviceID),
179 informationProvider = VALUES(informationProvider),
180 sourceAddr = VALUES(sourceAddr),
181 insertTime = VALUES(insertTime);
182
183     INSERT INTO GLUE20.ComputingServiceLoc
184 (computingServiceID,name,place,country,address,postcode,longitude,latitude,in
185 formationProvider,sourceAddr,insertTime) VALUES (
186         <xsl:value-of select=" ./node()[local-
187 name()='ComputingElement'][1]/node()[local-name()='Info']/@ns1:HostName"/> ",
188         <xsl:value-of select=" ./ancestor::node()[local-
189 name()='Entry']/following-sibling::node()[local-
190 name()='Site'][1]/@UniqueID"/> ",
191         <xsl:value-of select=" ./ancestor::node()[local-
192 name()='Entry']/following-sibling::node()[local-
193 name()='Site'][1]/node()[local-name()='Location']/text()"/> ",
194         <xsl:value-of select=" ./ancestor::node()[local-
195 name()='Entry']/following-sibling::node()[local-
196 name()='Site'][1]/node()[local-name()='Location']/text()"/> ",
197         <xsl:value-of select=" ./ancestor::node()[local-
198 name()='Entry']/following-sibling::node()[local-
199 name()='Site'][1]/node()[local-name()='Location']/text()"/> ",
200         <xsl:value-of select=" ./ancestor::node()[local-
201 name()='Entry']/following-sibling::node()[local-
202 name()='Site'][1]/node()[local-name()='Location']/text()"/> ",
203         <xsl:value-of select=" ./ancestor::node()[local-
204 name()='Entry']/following-sibling::node()[local-
205 name()='Site'][1]/node()[local-name()='Longitude']/text()"/> ",
206         <xsl:value-of select=" ./ancestor::node()[local-
207 name()='Entry']/following-sibling::node()[local-
208 name()='Site'][1]/node()[local-name()='Latitude']/text()"/> ",
209         <xsl:value-of select="$informationProvider\"/> ",
210         <xsl:value-of select="$sourceAddress\"/> ",
211         <xsl:value-of select="$insertTime\"/>)
212 ON DUPLICATE KEY UPDATE
213 computingServiceID = VALUES(computingServiceID),
214 name = VALUES(name),
215 place = VALUES(place),
216 country = VALUES(country),
217 address = VALUES(address),
218 postcode = VALUES(postcode),
219 longitude = VALUES(longitude),
220 latitude = VALUES(latitude),
221 informationProvider = VALUES(informationProvider),

```

```

222         sourceAddr = VALUES(sourceAddr),
223         insertTime = VALUES(insertTime);
224
225         INSERT INTO GLUE20.Endpoint (ID, serviceID,
226 informationProvider, sourceAddr, insertTime) VALUES (
227         "<xsl:value-of select='../../../../../../node()[local-
228 name()='MemberServiceEPR']/node()[local-name()='Address']/text()"/>",
229         "<xsl:value-of select='../../node()[local-
230 name()='ComputingElement'][1]/node()[local-name()='Info']/@ns1:HostName"/>",
231         "<xsl:value-of select='$informationProvider'"/>",
232         "<xsl:value-of select='$sourceAddress'"/>",
233         <xsl:value-of select='$insertTime'"/>)
234         ON DUPLICATE KEY UPDATE
235         ID = VALUES(ID),
236         serviceID = VALUES(serviceID),
237         informationProvider = VALUES(informationProvider),
238         sourceAddr = VALUES(sourceAddr),
239         insertTime = VALUES(insertTime);
240
241         INSERT INTO GLUE20.ComputingEndpoint (endpointID,
242 cServiceID, informationProvider, sourceAddr, insertTime) VALUES (
243         "<xsl:value-of select='../../../../../../node()[local-
244 name()='MemberServiceEPR']/node()[local-name()='Address']/text()"/>",
245         "<xsl:value-of select='../../node()[local-
246 name()='ComputingElement'][1]/node()[local-name()='Info']/@ns1:HostName"/>",
247         "<xsl:value-of select='$informationProvider'"/>",
248         "<xsl:value-of select='$sourceAddress'"/>",
249         <xsl:value-of select='$insertTime'"/>)
250         ON DUPLICATE KEY UPDATE
251         endpointID = VALUES(endpointID),
252         cServiceID = VALUES(cServiceID),
253         informationProvider = VALUES(informationProvider),
254         sourceAddr = VALUES(sourceAddr),
255         insertTime = VALUES(insertTime);
256
257     </xsl:if>
258 </xsl:template>
259
260
261 <xsl:template match="//ns1:SubCluster/ns1:Host">
262     <xsl:if test="not(count(..@ns1:Name)=0)">
263         <xsl:if test="not(count ../../../../node()[local-
264 name()='ComputingElement'][1]/node()[local-name()='Info']/@ns1:HostName)=0)">
265             INSERT INTO GLUE20.ExecutionEnvironment
266 (ID,cManagerID,serviceID,cpuClockSpeed,mainMemorySize,virtualMemorySize,osNam
267 e,osVersion,cpuMultiplicity,informationProvider,sourceAddr,insertTime) VALUES
268 (
269                 "<xsl:value-of select='../../@ns1:Name'"/>",
270                 "<xsl:value-of
271 select='../../../../../../ns1:ComputingElement[1]/ns1:Info/@ns1:HostName'"/>",
272                 "<xsl:value-of select='../../../../../../node()[local-
273 name()='ComputingElement'][1]/node()[local-name()='Info']/@ns1:HostName'"/>",
274                 "<xsl:value-of
275 select='../../ns1:Processor/@ns1:ClockSpeed'"/>",
276                 "<xsl:value-of
277 select='../../ns1:MainMemory/@ns1:RAMSize'"/>",

```

```

278         " <xsl:value-of
279 select=" ./ns1:MainMemory/@ns1:VirtualSize"/>",
280         " <xsl:value-of
281 select=" ./ns1:OperatingSystem/@ns1:Name"/>",
282         " <xsl:value-of
283 select=" ./ns1:OperatingSystem/@ns1:Release"/>",
284         " <xsl:value-of
285 select=" ./ns1:Architecture/@ns1:SMPSize"/>",
286         " <xsl:value-of select="$informationProvider"/>",
287         " <xsl:value-of select="$sourceAddress"/>",
288         <xsl:value-of select="$insertTime"/>)
289         ON DUPLICATE KEY UPDATE
290         ID = VALUES(ID),
291         cManagerID = VALUES(cManagerID),
292         serviceID = VALUES(serviceID),
293         cpuClockSpeed = VALUES(cpuClockSpeed),
294         mainMemorySize = VALUES(mainMemorySize),
295         virtualMemorySize = VALUES(virtualMemorySize),
296         osName = VALUES(osName),
297         osVersion = VALUES(osVersion),
298         cpuMultiplicity = VALUES(cpuMultiplicity),
299         informationProvider = VALUES(informationProvider),
300         sourceAddr = VALUES(sourceAddr),
301         insertTime = VALUES(insertTime);
302     </xsl:if>
303 </xsl:if>
304 </xsl:template>
305
306
307 <xsl:template match="//ns1:ComputingElement">
308     <xsl:if test="not(count(./node())[local-
309 name()='Info']/@ns1:HostName)=0)">
310         INSERT INTO GLUE20.ComputingShare
311 (serviceID,localID,mappingQueue,runningJobs,totalJobs,waitingJobs,information
312 Provider,sourceAddr,insertTime) VALUES (
313     " <xsl:value-of select=" ./node()[local-
314 name()='Info']/@ns1:HostName"/> ",
315     " <xsl:value-of select=" ./@ns1:UniqueID"/> ",
316     " <xsl:value-of select=" ./@ns1:UniqueID"/> ",
317     " <xsl:value-of select=" ./ns1:State/@ns1:RunningJobs"/> ",
318     " <xsl:value-of select=" ./ns1:State/@ns1:TotalJobs"/> ",
319     " <xsl:value-of select=" ./ns1:State/@ns1:WaitingJobs"/> ",
320     " <xsl:value-of select="$informationProvider"/> ",
321     " <xsl:value-of select="$sourceAddress"/> ",
322     <xsl:value-of select="$insertTime"/>)
323     ON DUPLICATE KEY UPDATE
324     serviceID = VALUES(serviceID),
325     localID = VALUES(localID),
326     mappingQueue = VALUES(mappingQueue),
327     runningJobs = VALUES(runningJobs),
328     totalJobs = VALUES(totalJobs),
329     waitingJobs = VALUES(waitingJobs),
330     informationProvider = VALUES(informationProvider),
331     sourceAddr = VALUES(sourceAddr),
332     insertTime = VALUES(insertTime);
333     </xsl:if>
334 </xsl:template>

```

```
335  
336 </xsl:stylesheet>
```

4.2 Unicore CIS Stylesheet

```

1  <?xml version="1.0" encoding="ISO-8859-1"?>
2  <xsl:stylesheet version="1.0"
3      xmlns:xsl="http://www.w3.org/1999/XSL/Transform"
4      xmlns:v29="http://schemas.opengroup.org/glue/2008/03/spec/2/0/v29"
5      xmlns:inf="http://www.unicore.eu/unicore6/cis/infoprovider"
6      xmlns:xsd="http://www.w3.org/2001/XMLSchema"
7      xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
8      xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" >
9
10 <xsl:output method="text" encoding="ISO-885-1" />
11
12 <xsl:variable name="provider">CIS</xsl:variable>
13 <xsl:param name="sourceAddress" select="cisclient" />
14 <xsl:variable name="insertTime">CURRENT_TIMESTAMP</xsl:variable>
15
16 <xsl:template match="AdminDomain">
17 INSERT GLUE20.AdminDomain (
18     ID,
19     name,
20     description,
21     distributed,
22     informationProvider,
23     sourceAddr,
24     insertTime
25 )
26 VALUES (
27     "<xsl:value-of select="ID" />",
28     "<xsl:value-of select="Name" />",
29     "<xsl:value-of select="Description" />",
30     IF("<xsl:value-of select="Distributed" />"="false", 0, 1),
31     "<xsl:value-of select="provider" />",
32     "<xsl:value-of select="sourceAddress" />",
33     "<xsl:value-of select="insertTime" />"
34 )
35 ON DUPLICATE KEY UPDATE
36     ID = values(ID),
37     name = values(name),
38     description = values(description),
39     distributed = values(distributed),
40     informationProvider = values(informationProvider),
41     sourceAddr = values(sourceAddr),
42     insertTime = values(insertTime);
43
44     <xsl:apply-templates select="Contact">
45         <xsl:with-param name="domainID" select="ID"/>

```

```

46         </xsl:apply-templates>
47
48         <xsl:apply-templates select="Services/ComputingService">
49             <xsl:with-param name="domainID" select="ID"/>
50         </xsl:apply-templates>
51
52 </xsl:template>
53
54
55 <xsl:template match="Location">
56     <xsl:param name="domainID" />
57
58     INSERT GLUE20.AdminDomainLocation (
59         adminDomainID,
60         localID,
61         name,
62         address,
63         place,
64         country,
65         postcode,
66         longitude,
67         latitude,
68         informationProvider,
69         sourceAddr,
70         insertTime
71     )
72     VALUES (
73         "<xsl:value-of select="$domainID" />",
74         "<xsl:value-of select="LocalID" />",
75         "<xsl:value-of select="Name" />",
76         "<xsl:value-of select="Address" />",
77         "<xsl:value-of select="Place" />",
78         "<xsl:value-of select="Country" />",
79         "<xsl:value-of select="PostCode" />",
80         "<xsl:value-of select="Longitude" />",
81         "<xsl:value-of select="Latitude" />",
82         "<xsl:value-of select="$provider" />",
83         "<xsl:value-of select="$sourceAddress" />",
84         <xsl:value-of select="$insertTime" />
85     )
86     ON DUPLICATE KEY UPDATE
87         adminDomainID = values(adminDomainID),
88         localID = values(localID),
89         name = values(name),
90         address = values(address),
91         place = values(place),

```

```

92     country = values(country),
93     postcode = values(postcode),
94     longitude = values(longitude),
95     latitude = values(latitude),
96     informationProvider = values(informationProvider),
97     sourceAddr = values(sourceAddr),
98     insertTime = values(insertTime);
99 </xsl:template>
100
101
102 <xsl:template match="Contact">
103     <xsl:param name="domainID" />
104
105     INSERT GLUE20.AdminContact (
106         adminDomainID,
107         localID,
108         url,
109         type,
110         informationProvider,
111         sourceAddr,
112         insertTime
113     )
114     VALUES (
115         "<xsl:value-of select="$domainID" />",
116         "<xsl:value-of select="LocalID" />",
117         "<xsl:value-of select="URL" />",
118         "<xsl:value-of select="Type" />",
119         "<xsl:value-of select="$provider" />",
120         "<xsl:value-of select="$sourceAddress" />",
121         <xsl:value-of select="$insertTime" />
122     )
123     ON DUPLICATE KEY UPDATE
124         adminDomainID = values(adminDomainID),
125         localID = values(localID),
126         url = values(url),
127         type = values(type),
128         informationProvider = values(informationProvider),
129         sourceAddr = values(sourceAddr),
130         insertTime = values(insertTime);
131 </xsl:template>
132
133
134 <xsl:template match="ComputingService">
135     <xsl:param name="domainID" />
136
137     INSERT GLUE20.ComputingService (

```

```

138         ID,
139         name,
140         type,
141         qualityLevel,
142         totalJobs,
143         runningJobs,
144         waitingJobs,
145         domainID,
146         informationProvider,
147         sourceAddr,
148         insertTime
149     )
150 VALUES (
151     "<xsl:value-of select="ID" />",
152     "<xsl:value-of select="Name" />",
153     "<xsl:value-of select="Type" />",
154     "<xsl:value-of select="QualityLevel" />",
155     "<xsl:value-of select="TotalJobs" />",
156     "<xsl:value-of select="RunningJobs" />",
157     "<xsl:value-of select="WaitingJobs" />",
158     "<xsl:value-of select="&#36;domainID" />",
159     "<xsl:value-of select="&#36;provider" />",
160     "<xsl:value-of select="&#36;sourceAddress" />",
161     "<xsl:value-of select="&#36;insertTime" />"
162 )
163 ON DUPLICATE KEY UPDATE
164     ID = values(ID),
165     name = values(name),
166     type = values(type),
167     qualityLevel = values(qualityLevel),
168     totalJobs = values(totalJobs),
169     runningJobs = values(runningJobs),
170     waitingJobs = values(waitingJobs),
171     domainID = values(domainID),
172     informationProvider = values(informationProvider),
173     sourceAddr = values(sourceAddr),
174     insertTime = values(insertTime);
175
176     <xsl:apply-templates select="ComputingEndpoint">
177         <xsl:with-param name="serviceID" select="ID"/>
178     </xsl:apply-templates>
179
180     <xsl:apply-templates select="ComputingManager">
181         <xsl:with-param name="serviceID" select="ID"/>
182     </xsl:apply-templates>
183

```

```

184         <xsl:apply-templates select="Location">
185             <xsl:with-param name="domainID" select="$domainID"/>
186         </xsl:apply-templates>
187
188     </xsl:template>
189
190
191     <xsl:template match="ComputingEndpoint">
192         <xsl:param name="serviceID" />
193
194     INSERT GLUE20.Endpoint (
195         serviceID,
196         ID,
197         name,
198         url,
199         technology,
200         interface,
201         implementor,
202         implementationName,
203         implementationVersion,
204         qualityLevel,
205         healthState,
206         healthStateInfo,
207         servingState,
208         informationProvider,
209         sourceAddr,
210         insertTime
211     )
212     VALUES (
213         "<xsl:value-of select="$serviceID" />",
214         "<xsl:value-of select="ID" />",
215         "<xsl:value-of select="Name" />",
216         "<xsl:value-of select="URL" />",
217         "<xsl:value-of select="Technology" />",
218         "<xsl:value-of select="InterfaceName" />",
219         "<xsl:value-of select="Implementor" />",
220         "<xsl:value-of select="ImplementationName" />",
221         "<xsl:value-of select="ImplementationVersion" />",
222         "<xsl:value-of select="QualityLevel" />",
223         "<xsl:value-of select="HealthState" />",
224         "<xsl:value-of select="HealthStateInfo" />",
225         "<xsl:value-of select="ServingState" />",
226         "<xsl:value-of select="$provider" />",
227         "<xsl:value-of select="$sourceAddress" />",
228         <xsl:value-of select="$insertTime" />
229     )

```

```

230 ON DUPLICATE KEY UPDATE
231     serviceID = values(serviceID),
232     ID = values(ID),
233     name = values(name),
234     technology = values(technology),
235     interface = values(interface),
236     implementor = values(implementor),
237     implementationName = values(implementationName),
238     implementationVersion = values(implementationVersion),
239     qualityLevel = values(qualityLevel),
240     healthState = values(healthState),
241     healthStateInfo = values(healthStateInfo),
242     servingState = values(servingState),
243     informationProvider = values(informationProvider),
244     sourceAddr = values(sourceAddr),
245     insertTime = values(insertTime);
246
247
248 INSERT GLUE20.ComputingEndpoint (
249     endpointID,
250     cServiceID,
251     informationProvider,
252     sourceAddr,
253     insertTime
254 )
255 VALUES (
256     "<xsl:value-of select="ID" />",
257     "<xsl:value-of select="serviceID" />",
258     "<xsl:value-of select="provider" />",
259     "<xsl:value-of select="sourceAddress" />",
260     <xsl:value-of select="insertTime" />
261 )
262 ON DUPLICATE KEY UPDATE
263     endpointID = values(endpointID),
264     cServiceID = values(cServiceID),
265     informationProvider = values(informationProvider),
266     sourceAddr = values(sourceAddr),
267     insertTime = values(insertTime);
268 </xsl:template>
269
270
271 <xsl:template match="ComputingManager">
272     <xsl:param name="serviceID" />
273
274 INSERT GLUE20.ComputingManager (
275     serviceID,

```

```

276         ID,
277         name,
278         totalPhysicalCPU,
279         totalLogicalCPUs,
280         informationProvider,
281         sourceAddr,
282         insertTime
283     )
284 VALUES (
285     "<xsl:value-of select=\"$serviceID\" />",
286     "<xsl:value-of select=\"ID\" />",
287     "<xsl:value-of select=\"Name\" />",
288     "<xsl:value-of select=\"TotalPhysicalCPUs\" />",
289     "<xsl:value-of select=\"TotalLogicalCPUs\" />",
290     "<xsl:value-of select=\"$provider\" />",
291     "<xsl:value-of select=\"$sourceAddress\" />",
292     "<xsl:value-of select=\"$insertTime\" />"
293 )
294 ON DUPLICATE KEY UPDATE
295     serviceID = values(serviceID),
296     ID = values(ID),
297     name = values(name),
298     totalPhysicalCPU = values(totalPhysicalCPU),
299     totalLogicalCPUs = values(totalLogicalCPUs),
300     informationProvider = values(informationProvider),
301     sourceAddr = values(sourceAddr),
302     insertTime = values(insertTime);
303
304
305     <xsl:apply-templates select="ExecutionEnvironment">
306         <xsl:with-param name="cManagerID" select="ID"/>
307         <xsl:with-param name="serviceID" select="$serviceID"/>
308     </xsl:apply-templates>
309
310     <xsl:apply-templates
311     select="ApplicationEnvironments/ApplicationEnvironment">
312         <xsl:with-param name="serviceID" select="$serviceID"/>
313         <xsl:with-param name="cManagerID" select="ID"/>
314     </xsl:apply-templates>
315
316 </xsl:template>
317
318
319 <xsl:template match="ExecutionEnvironment">
320     <xsl:param name="serviceID" />
321     <xsl:param name="cManagerID" />

```

```

322
323 INSERT GLUE20.ExecutionEnvironment (
324     serviceID,
325     ID,
326     platform,
327     virtualMachine,
328     totalInstances,
329     physicalCPUs,
330     logicalCPUs,
331     mainMemorySize,
332     osFamiliy,
333     osName,
334     cManagerID,
335     informationProvider,
336     sourceAddr,
337     insertTime
338 )
339 VALUES (
340     "<xsl:value-of select=\"$serviceID\" />",
341     "<xsl:value-of select=\"ID\" />",
342     "<xsl:value-of select=\"Platform\" />",
343     "<xsl:value-of select=\"VirtualMachine\" />",
344     "<xsl:value-of select=\"TotalInstances\" />",
345     "<xsl:value-of select=\"PhysicalCPUs\" />",
346     "<xsl:value-of select=\"LogicalCPUs\" />",
347     "<xsl:value-of select=\"MainMemorySize\" />",
348     "<xsl:value-of select=\"OSFamily\" />",
349     "<xsl:value-of select=\"OSName\" />",
350     "<xsl:value-of select=\"$cManagerID\" />",
351     "<xsl:value-of select=\"$provider\" />",
352     "<xsl:value-of select=\"$sourceAddress\" />",
353     "<xsl:value-of select=\"$insertTime\" />"
354 )
355 ON DUPLICATE KEY UPDATE
356     serviceID = values(serviceID),
357     ID = values(ID),
358     platform = values(platform),
359     virtualMachine = values(virtualMachine),
360     totalInstances = values(totalInstances),
361     physicalCPUs = values(physicalCPUs),
362     logicalCPUs = values(logicalCPUs),
363     mainMemorySize = values(mainMemorySize),
364     osFamiliy = values(osFamiliy),
365     osName = values(osName),
366     cManagerID = values(cManagerID),
367     informationProvider = values(informationProvider),

```

```

368         sourceAddr = values(sourceAddr),
369         insertTime = values(insertTime);
370
371     </xsl:template>
372
373     <xsl:template match="ApplicationEnvironment">
374         <xsl:param name="serviceID" />
375         <xsl:param name="cManagerID" />
376
377         INSERT GLUE20.ApplicationEnvironment (
378             serviceID,
379             localID,
380             name,
381             version,
382             cManagerID,
383             informationProvider,
384             sourceAddr,
385             insertTime
386         )
387     VALUES (
388         "<xsl:value-of select="$serviceID" />",
389         "<xsl:value-of select="LocalID" />",
390         "<xsl:value-of select="Name" />",
391         "<xsl:value-of select="Version" />",
392         "<xsl:value-of select="$cManagerID" />",
393         "<xsl:value-of select="$provider" />",
394         "<xsl:value-of select="$sourceAddress" />",
395         <xsl:value-of select="$insertTime" />
396     )
397     ON DUPLICATE KEY UPDATE
398         serviceID = values(serviceID),
399         localID = values(localID),
400         name = values(name),
401         version = values(version),
402         cManagerID = values(cManagerID),
403         informationProvider = values(informationProvider),
404         sourceAddr = values(sourceAddr),
405         insertTime = values(insertTime);
406
407     </xsl:template>
408
409 </xsl:stylesheet>
410

```

4.3 gLite BDII Sylesheet

```

1 <xsl:stylesheet version="1.0" xmlns:xsl=http://www.w3.org/1999/XSL/Transform
2 xmlns:dsml="http://www.dsml.org/DSML">
3
4 <xsl:output method="text" encoding="ISO-8859-1" />
5
6 <xsl:variable name="informationProvider">BDII</xsl:variable>
7 <xsl:param name="sourceAddress" select="'ldapclient'" />
8 <xsl:variable name="insertTime">CURRENT_TIMESTAMP</xsl:variable>
9 <xsl:variable name="EntryTypeID">105</xsl:variable>
10 <xsl:template match="/">
11     <xsl:apply-templates select="//dsml:dsml/dsml:directory-entries/dsml:entry"/>
12 </xsl:template>
13
14 <xsl:template match="//dsml:dsml/dsml:directory-entries/dsml:entry">
15
16     <xsl:variable name="DistinguishedNameGlueSiteC"><xsl:value-of select="@dn"/></xsl:variable>
17     <xsl:choose>
18         <xsl:when test="starts-with(@dn,'GlueSiteUniqueID')">
19             <!-- ObjectClass for Filtering LDAP searches is ObjectClass=GlueSite -->
20             <xsl:variable name="DistinguishedNameLocal">
21                 <xsl:value-of select = "substring-after(substring-
22                 before($DistinguishedNameGlueSiteC,',mds-vo-name=local,o=grid'),'o-name=')"/>
23             </xsl:variable>
24             INSERT INTO
25             GLUE20.AdminContact
26                 (
27                 LocalID,
28                 Type,
29                 URL,
30                 <!-- OtherInfo,-->
31                 adminDomainID,
32                 informationProvider,
33                 sourceAddr,
34                 insertTime
35                 )
36             VALUES
37                 (

```

```

38         "<xsl:value-of select='dsml:attr[@name='GlueSiteUniqueID']/dsml:value'/>",
39         "<xsl:value-of select='dsml:attr[@name='GlueSiteDescription']/dsml:value'/>",
40         "<xsl:value-of select='dsml:attr[@name='GlueSiteWeb']/dsml:value'/>",
41         <!--<xsl:value-of select='dsml:attr[@name='GlueSiteOtherInfo']/dsml:value'/>-->
42         "<xsl:value-of select='$DistinguishedNameLocal'/>",
43         "<xsl:value-of select='$informationProvider'/>",
44         "<xsl:value-of select='$sourceAddress'/>",
45         <xsl:value-of select='$insertTime'/>)
46     ON DUPLICATE KEY UPDATE
47         Type = VALUES(Type),
48         URL = VALUES(URL),
49         LocalID = VALUES(LocalID),
50         informationProvider = VALUES(informationProvider),
51         sourceAddr = VALUES(sourceAddr),
52         insertTime = VALUES(insertTime);
53     INSERT INTO
54     GLUE20.AdminContact
55     (
56         LocalID,
57         Type,
58         URL,
59         <!-- OtherInfo,-->
60         adminDomainID,
61         informationProvider,
62         sourceAddr,insertTime
63     )
64     VALUES
65     (
66         "<xsl:value-of select='dsml:attr[@name='GlueSiteUniqueID']/dsml:value'/>-UserSupport",
67         "UserSupport",
68         "<xsl:value-of select='dsml:attr[@name='GlueSiteUserSupportContact']/dsml:value'/>",
69         <!--<xsl:value-of select='dsml:attr[@name='GlueSiteOtherInfo']/dsml:value'/>-->
70         "<xsl:value-of select='$DistinguishedNameLocal'/>",
71         "<xsl:value-of select='$informationProvider'/>",
72         "<xsl:value-of select='$sourceAddress'/>",
73         <xsl:value-of select='$insertTime'/>)
74     ON DUPLICATE KEY UPDATE
75         Type = VALUES(Type),

```

```

76         URL = VALUES(URL),
77         LocalID = VALUES(LocalID),
78         informationProvider = VALUES(informationProvider),
79         sourceAddr = VALUES(sourceAddr),
80         insertTime = VALUES(insertTime);
81     INSERT INTO
82     GLUE20.AdminContact
83     (
84         LocalID,
85         Type,
86         URL,
87         <!-- OtherInfo,-->
88         adminDomainID,
89         informationProvider,
90         sourceAddr,insertTime
91     )
92     VALUES
93     (
94         "<xsl:value-of select="dsml:attr[@name='GlueSiteUniqueID']/dsml:value"/>-SysAdmin",
95         "SysAdmin",
96         "<xsl:value-of select="dsml:attr[@name='GlueSiteSysAdminContact']/dsml:value"/>",
97         <!--"<xsl:value-of select="dsml:attr[@name='GlueSiteOtherInfo']/dsml:value"/>,"-->
98         "<xsl:value-of select="DistinguishedNameLocal"/>",
99         "<xsl:value-of select="informationProvider"/>",
100        "<xsl:value-of select="sourceAddress"/>",
101        <xsl:value-of select="insertTime"/>
102    )
103     ON DUPLICATE KEY UPDATE
104         Type = VALUES(Type),
105         URL = VALUES(URL),
106         LocalID = VALUES(LocalID),
107         informationProvider = VALUES(informationProvider),
108         sourceAddr = VALUES(sourceAddr),
109         insertTime = VALUES(insertTime);
110     INSERT INTO
111     GLUE20.AdminContact
112     (
113         LocalID,

```

```

114         Type,
115         URL,
116         <!-- OtherInfo,-->
117         adminDomainID,
118         informationProvider,
119         sourceAddr,insertTime
120     )
121     VALUES
122     (
123         "<xsl:value-of select='dsml:attr[@name='GlueSiteUniqueID']/dsml:value'/>-Security",
124         "Security",
125         "<xsl:value-of select='dsml:attr[@name='GlueSiteSecurityContact']/dsml:value'/>",
126         <!--"<xsl:value-of select='dsml:attr[@name='GlueSiteOtherInfo']/dsml:value'/>,"-->
127         "<xsl:value-of select='$DistinguishedNameLocal'/>",
128         "<xsl:value-of select='$informationProvider'/>",
129         "<xsl:value-of select='$sourceAddress'/>",
130         <xsl:value-of select='$insertTime'/>
131     )
132     ON DUPLICATE KEY UPDATE
133         Type = VALUES(Type),
134         URL = VALUES(URL),
135         LocalID = VALUES(LocalID),
136         informationProvider = VALUES(informationProvider),
137         sourceAddr = VALUES(sourceAddr),
138         insertTime = VALUES(insertTime);
139 </xsl:when>
140 </xsl:choose>
141
142 <xsl:variable name="DistinguishedNameGlueSiteA"><xsl:value-of select="@dn"/></xsl:variable>
143 <xsl:choose>
144     <xsl:when test="starts-with(@dn,'GlueSiteUniqueID')">
145         <!-- ObjectClass for Filtering LDAP searches is ObjectClass=GlueSite -->
146         <xsl:variable name="DistinguishedNameLocal">
147             <xsl:value-of select = "substring-after(substring-
148 before($DistinguishedNameGlueSiteA,'mds-vo-name=local,o=grid'),'o-name=')" />
149         </xsl:variable>
150         <xsl:variable name="Location">
151             <xsl:value-of select='dsml:attr[@name='GlueSiteLocation']/dsml:value'/>

```

```

152     </xsl:variable>
153
154     INSERT INTO
155     GLUE20.AdminDomain
156         (
157             ID,
158             Name,
159             Description,
160             Distributed,
161             informationProvider,
162             sourceAddr,
163             insertTime
164         )
165     VALUES
166         (
167             "<xsl:value-of select="$DistinguishedNameLocal"/>",
168             "<xsl:value-of select='dsml:attr[@name='GlueSiteName']/dsml:value'/>",
169             "<xsl:value-of select='dsml:attr[@name='GlueSiteDescription']/dsml:value'/>-<xsl:value-
170 of select='dsml:attr[@name='GlueSiteOtherInfo']/dsml:value'/>",
171             "0",
172             "<xsl:value-of select='$informationProvider'/>",
173             "<xsl:value-of select='$sourceAddress'/>",
174             <xsl:value-of select='$insertTime'/>)
175     ON DUPLICATE KEY UPDATE
176         Name = VALUES (Name),
177         Description = VALUES (Description),
178         Distributed = VALUES (Distributed),
179         sourceAddr = VALUES (sourceAddr),
180         insertTime = VALUES (insertTime);
181 </xsl:when>
182 </xsl:choose>
183
184 <xsl:variable name="DistinguishedNameGlueSite"><xsl:value-of select="@dn"/></xsl:variable>
185 <xsl:choose>
186     <xsl:when test="starts-with(@dn,'GlueSiteUniqueID')">
187         <!-- ObjectClass for Filtering LDAP searches is ObjectClass=GlueSite -->
188         <xsl:variable name="DistinguishedNameLocal">

```

```

189         <xsl:value-of select = "substring-after(substring-
190 before($DistinguishedNameGlueSite,' ,mds-vo-name=local,o=grid'),'o-name=')" />
191     </xsl:variable>
192     <xsl:variable name="Location">
193         <xsl:value-of select="dsml:attr[@name='GlueSiteLocation']/dsml:value"/></xsl:variable>
194         <!--<xsl:value-of select="$Place"/>
195         <xsl:value-of select="$County"/>
196         -->
197     INSERT INTO GLUE20.AdminDomain
198         (
199         ID,
200         Name,
201         Description,
202         Distributed,
203         informationProvider,
204         sourceAddr,
205         insertTime
206     )
207     VALUES
208     (
209         "<xsl:value-of select="$DistinguishedNameLocal"/>",
210         "<xsl:value-of select="dsml:attr[@name='GlueSiteName']/dsml:value"/>",
211         "<xsl:value-of select="$Location"/>",
212         "0",
213         "<xsl:value-of select="$informationProvider"/>",
214         "<xsl:value-of select="$sourceAddress"/>",
215         <xsl:value-of select="$insertTime"/>
216     )
217     ON DUPLICATE KEY UPDATE
218         Name = VALUES (Name),
219         Description = VALUES (Description),
220         Distributed = VALUES (Distributed),
221         sourceAddr = VALUES (sourceAddr),
222         insertTime = VALUES (insertTime);
223     INSERT INTO
224     GLUE20.AdminDomainLocation
225     (
226         localID,

```

```

227         name,
228         place,
229         country,
230         address
231         <!--,postcode-->,
232         longitude,
233         latitude,
234         adminDomainID,
235         informationProvider,
236         sourceAddr,
237         insertTime
238     )
239     VALUES
240     (
241         "<xsl:value-of select='dsml:attr[@name='GlueSiteUniqueID']/dsml:value'/>",
242         "<xsl:value-of select='dsml:attr[@name='GlueSiteName']/dsml:value'/>",
243         <xsl:choose>
244             <xsl:when test="contains($Location,',')">
245                 "<xsl:value-of select = 'substring-before($Location,',')' />",
246                 "<xsl:value-of select = 'substring-after($Location,',')' />",
247             </xsl:when>
248             <xsl:otherwise>
249                 "unknown",
250                 "<xsl:value-of select = '$Location' />",
251             </xsl:otherwise>
252         </xsl:choose>
253         "<xsl:value-of select='dsml:attr[@name='GlueSiteWeb']/dsml:value'/>",
254         <!--$Postalcode"-->
255         "<xsl:value-of select='dsml:attr[@name='GlueSiteLongitude']/dsml:value'/>",
256         "<xsl:value-of select='dsml:attr[@name='GlueSiteLatitude']/dsml:value'/>",
257         "<xsl:value-of select='$DistinguishedNameLocal'/>",
258         "<xsl:value-of select='$informationProvider'/>",
259         "<xsl:value-of select='$sourceAddress'/>",
260         <xsl:value-of select='$insertTime'/>
261     ON DUPLICATE KEY UPDATE
262         localID = VALUES (localID),
263         Name = VALUES (Name),
264         place = VALUES (place),

```

```

265         country = VALUES (country),
266         address = VALUES (address),
267         postcode = VALUES (postcode),
268         longitude = VALUES (longitude),
269         latitude = VALUES (latitude),
270         sourceAddr = VALUES (sourceAddr),
271         insertTime = VALUES (insertTime);
272     </xsl:when>
273 </xsl:choose>
274
275 <xsl:variable name="DistinguishedNameBENCH"><xsl:value-of select="@dn"/></xsl:variable>
276 <xsl:choose>
277     <xsl:when test="starts-with(@dn, 'GlueSubClusterUniqueID')">
278         <!-- ObjectClass for Filtering LDAP searches is ObjectClass=GlueSubCluster -->
279         INSERT IGNORE INTO GLUE20.EntryTypes
280         (ID,informationProvider,sourceAddr,insertTime) VALUES ("<xsl:value-of
281         select="$EntryTypeID"/>",<xsl:value-of select="$informationProvider"/>",<xsl:value-of
282         select="$sourceAddress"/>",<xsl:value-of select="$insertTime"/>);
283         <xsl:variable name="GlueSubClusterUniqueID">
284             <xsl:value-of select = "substring-before(substring-
285             after($DistinguishedNameBENCH,'GlueSubClusterUniqueID=',')'" />
286         </xsl:variable>
287         INSERT INTO
288         GLUE20.Benchmark
289         (
290         parentID,
291         Type,
292         Value,
293         LocalID,
294         parentType,
295         informationProvider,
296         sourceAddr,
297         insertTime
298         )
299         VALUES
300         (
301         "<xsl:value-of select="$GlueSubClusterUniqueID"/>",<br>
302         "SPECint 2000",
303         "<xsl:value-of select="dsml:attr[@name='GlueHostBenchmarkSI00']/dsml:value"/>",<br>

```

```

304         "<xsl:value-of select='dsml:attr[@name='GlueSubClusterName']/dsml:value'/>",
305         "<xsl:value-of select='$EntryTypeID'/>",
306         "<xsl:value-of select='$informationProvider'/>",
307         "<xsl:value-of select='$sourceAddress'/>",
308         "<xsl:value-of select='$insertTime'/>"
309     )
310 ON DUPLICATE KEY UPDATE
311     parentID = VALUES(parentID),
312     Type = VALUES(Type),
313     Value = VALUES(Value),
314     LocalID = VALUES(LocalID),
315     sourceAddr = VALUES(sourceAddr),
316     insertTime = VALUES(insertTime);
317 INSERT INTO
318 GLUE20.Benchmark
319     (
320     parentID,
321     Type,
322     Value,
323     LocalID,
324     parentType,
325     informationProvider,
326     sourceAddr,
327     insertTime
328     )
329 VALUES
330     (
331     "<xsl:value-of select='$GlueSubClusterUniqueID'/>",
332     "SPECfloat 2000",
333     "<xsl:value-of select='dsml:attr[@name='GlueHostBenchmarkSF00']/dsml:value'/>",
334     "<xsl:value-of select='dsml:attr[@name='GlueSubClusterName']/dsml:value'/>",
335     "<xsl:value-of select='$EntryTypeID'/>",
336     "<xsl:value-of select='$informationProvider'/>",
337     "<xsl:value-of select='$sourceAddress'/>",
338     "<xsl:value-of select='$insertTime'/>")
339 ON DUPLICATE KEY UPDATE
340     parentID = VALUES(parentID),
341     Type = VALUES(Type),

```

```

342         Value = VALUES(Value),
343         LocalID = VALUES(LocalID),
344         sourceAddr = VALUES(sourceAddr),
345         insertTime = VALUES(insertTime);
346     </xsl:when>
347 </xsl:choose>
348
349 <xsl:variable name="DistinguishedNameGlueClusterTop"><xsl:value-of select="@dn"/></xsl:variable>
350 <xsl:choose>
351     <xsl:when test="starts-with($DistinguishedNameGlueClusterTop, 'GlueClusterUniqueID')">
352         <!-- ObjectClass for Filtering LDAP searches is ObjectClass=GlueClusterTop -->
353         <xsl:variable name="DistinguishedNameLocal">
354             <xsl:value-of select = "substring-after(substring-
355 before($DistinguishedNameGlueClusterTop,',mds-vo-name=local,o=grid'),'o-name=')" />
356         </xsl:variable>
357         <xsl:variable name="JobManagerType">
358             <xsl:value-of select="dsml:attr[@name='GlueClusterService']/dsml:value"/>
359         </xsl:variable>
360         <xsl:variable name="JobManagerTypeLocal">
361             <xsl:value-of select = "substring-after($JobManagerType,'jobmanager-')" />
362         </xsl:variable>
363         INSERT IGNORE GLUE20.AdminDomain (ID,informationProvider) VALUES ("<xsl:value-of
364 select="$DistinguishedNameLocal"/>",<xsl:value-of select="$informationProvider"/>");
365         INSERT INTO
366         GLUE20.ComputingService
367         (
368             Name,
369             <!-- StatusPage, -->
370             ID,
371             Type,
372             domainID,
373             informationProvider,
374             sourceAddr,
375             insertTime
376         )
377         VALUES
378         (
379             "<xsl:value-of select="dsml:attr[@name='GlueClusterName']/dsml:value"/>,"

```

```

380         <!-- " <xsl:value-of
381 select="dsml:attr[@name='GlueInformationServiceURL']/dsml:value"/> ", -->
382         "<xsl:value-of select="dsml:attr[@name='GlueClusterUniqueID']/dsml:value"/> ",
383         "<xsl:value-of select=" $JobManagerTypeLocal"/> ",
384         "<xsl:value-of select=" $DistinguishedNameLocal"/> ",
385         "<xsl:value-of select=" $informationProvider"/> ",
386         "<xsl:value-of select=" $sourceAddress"/> ",
387         <xsl:value-of select=" $insertTime"/>
388     )
389     ON DUPLICATE KEY UPDATE
390         Name = VALUES(Name),
391         <!-- StatusPage = VALUES( StatusPage), -->
392         ID = VALUES(ID),
393         Type = VALUES(Type),
394         sourceAddr = VALUES (sourceAddr),
395         insertTime = VALUES (insertTime);
396 </xsl:when>
397 </xsl:choose>
398
399 <xsl:variable name="DistinguishedNameGlueCEM"><xsl:value-of select="@dn"/></xsl:variable>
400 <xsl:choose>
401     <xsl:when test="starts-with($DistinguishedNameGlueCEM, 'GlueCEUniqueID')">
402         <!-- ObjectClass for Filtering LDAP searches is ObjectClass=GlueCE -->
403         <xsl:variable name="DistinguishedNameLocal">
404             <xsl:value-of select = "substring-after(substring-
405 before($DistinguishedNameGlueCEM, ',mds-vo-name=local,o=grid'),'o-name=') " />
406         </xsl:variable>
407         INSERT IGNORE GLUE20.AdminDomain (ID,informationProvider) VALUES ("<xsl:value-of
408 select=" $DistinguishedNameLocal"/> ", "<xsl:value-of select=" $informationProvider"/> ");
409         INSERT IGNORE GLUE20.ComputingService (ID,domainID,informationProvider) VALUES
410 ("<xsl:value-of select=" dsml:attr[@name='GlueCEHostingCluster']/dsml:value"/> ", "<xsl:value-of
411 select=" $DistinguishedNameLocal"/> ", "<xsl:value-of select=" $informationProvider"/> ");
412         INSERT INTO
413         GLUE20.ComputingManager
414         (
415             Name,
416             ApplicationDir,
417             TmpDir,
418             ID,

```

```

419         Type,
420         Version,
421         <!--OtherInfo, TotalPhysicalCPUs, -->
422         serviceID,
423         informationProvider,
424         sourceAddr,
425         insertTime
426     )
427     VALUES
428     (
429         "<xsl:value-of select='dsml:attr[@name='GlueCEUniqueID']/dsml:value'/>",
430         "<xsl:value-of select='dsml:attr[@name='GlueCEInfoApplicationDir']/dsml:value'/>",
431         "<xsl:value-of select='dsml:attr[@name='GlueCEInfoDataDir']/dsml:value'/>",
432         "<xsl:value-of select='dsml:attr[@name='GlueCEInfoHostName']/dsml:value'/>",
433         "<xsl:value-of select='dsml:attr[@name='GlueCEInfoLRMSType']/dsml:value'/>",
434         "<xsl:value-of select='dsml:attr[@name='GlueCEInfoLRMSVersion']/dsml:value'/>",
435         <!--<xsl:value-of select='dsml:attr[@name='GlueCEInfoDefaultSE']/dsml:value'/>-->,
436         "<xsl:value-of select='dsml:attr[@name='GlueCEInfoTotalCPUs']/dsml:value'/>-->",
437         "<xsl:value-of select='dsml:attr[@name='GlueCEHostingCluster']/dsml:value'/>",
438         "<xsl:value-of select='$informationProvider'/>",
439         "<xsl:value-of select='$sourceAddress'/>",
440         <xsl:value-of select='$insertTime'/>)
441     ON DUPLICATE KEY UPDATE
442         ID = VALUES (ID),
443         ApplicationDir = VALUES (ApplicationDir ),
444         TmpDir = VALUES (TmpDir),
445         Name = VALUES (Name),
446         Type = VALUES (Type),
447         Version = VALUES (Version),
448         sourceAddr = VALUES (sourceAddr),
449         insertTime = VALUES (insertTime);
450     </xsl:when>
451 </xsl:choose>
452
453 <xsl:variable name="DistinguishedNameGlueCE"><xsl:value-of select="@dn"/></xsl:variable>
454 <xsl:choose>
455     <xsl:when test="starts-with($DistinguishedNameGlueCE, 'GlueCEUniqueID')">
456         <!-- ObjectClass for Filtering LDAP searches is ObjectClass=GlueCE -->

```

```

457     <xsl:variable name="DistinguishedNameLocal">
458         <xsl:value-of select = "substring-after(substring-
459         before($DistinguishedNameGlueCE,',mds-vo-name=local,o=grid'),'o-name=')"/>
460     </xsl:variable>
461     INSERT IGNORE GLUE20.AdminDomain (ID,informationProvider) VALUES ("<xsl:value-of
462     select="$DistinguishedNameLocal"/>",<xsl:value-of select="$informationProvider"/>");
463     INSERT IGNORE GLUE20.StorageService (ID,domainID,informationProvider) VALUES
464     ("<xsl:value-of select="dsml:attr[@name='GlueCEInfoDefaultSE']/dsml:value"/>",<xsl:value-of
465     select="$DistinguishedNameLocal"/>",<xsl:value-of select="$informationProvider"/>");
466     INSERT INTO
467     GLUE20.ComputingShare
468     (
469     Name,
470     MaxCPUTime,
471     MaxRunningJobs,
472     MaxTotalJobs,
473     MaxWallTime,
474     SchedulingPolicy,
475     EstimatedAverageWaitingTime,
476     FreeSlots,
477     RunningJobs,
478     Description,
479     TotalJobs,
480     WaitingJobs,
481     EstimatedWorstWaitingTime,
482     LocalID,
483     serviceID,
484     DefaultSService,
485     informationProvider,
486     sourceAddr,
487     insertTime
488     )
489     VALUES
490     (
491     "<xsl:value-of select="dsml:attr[@name='GlueCENAME']/dsml:value"/>",
492     "<xsl:value-of select="dsml:attr[@name='GlueCEPolicyMaxCPUTime']/dsml:value"/>",
493     "<xsl:value-of select="dsml:attr[@name='GlueCEPolicyMaxRunningJobs']/dsml:value"/>",
494     "<xsl:value-of select="dsml:attr[@name='GlueCEPolicyMaxTotalJobs']/dsml:value"/>",

```

```

495         <xsl:value-of
496 select="dsml:attr[@name='GlueCEPolicyMaxWallClockTime']/dsml:value"/>,
497         <xsl:value-of select="dsml:attr[@name='GlueCEPolicyPriority']/dsml:value"/>,
498         <xsl:value-of
499 select="dsml:attr[@name='GlueCEStateEstimatedResponseTime']/dsml:value"/>,
500         <xsl:value-of select="dsml:attr[@name='GlueCEStateFreeJobSlots']/dsml:value"/>,
501         <xsl:value-of select="dsml:attr[@name='GlueCEStateRunningJobs']/dsml:value"/>,
502         <xsl:value-of select="dsml:attr[@name='GlueCEStateStatus']/dsml:value"/>,
503         <xsl:value-of select="dsml:attr[@name='GlueCEStateTotalJobs']/dsml:value"/>,
504         <xsl:value-of select="dsml:attr[@name='GlueCEStateWaitingJobs']/dsml:value"/>,
505         <xsl:value-of
506 select="dsml:attr[@name='GlueCEStateWorstResponseTime']/dsml:value"/>,
507         <xsl:value-of select="dsml:attr[@name='GlueCEUniqueID']/dsml:value"/>,
508         <xsl:value-of select="dsml:attr[@name='GlueCEHostingCluster']/dsml:value"/>,
509         <xsl:value-of select="dsml:attr[@name='GlueCEInfoDefaultSE']/dsml:value"/>,
510         <xsl:value-of select="$informationProvider"/>,
511         <xsl:value-of select="$sourceAddress"/>,
512         <xsl:value-of select="$insertTime"/>)
513     ON DUPLICATE KEY UPDATE
514         Name = VALUES(Name),
515         MaxCPUTime = VALUES(MaxCPUTime),
516         MaxRunningJobs = VALUES(MaxRunningJobs),
517         MaxTotalJobs = VALUES(MaxTotalJobs),
518         MaxWallTime = VALUES(MaxWallTime),
519         SchedulingPolicy = VALUES(SchedulingPolicy),
520         EstimatedAverageWaitingTime = VALUES(EstimatedAverageWaitingTime),
521         FreeSlots = VALUES(FreeSlots),
522         RunningJobs = VALUES(RunningJobs),Description = VALUES(Description),
523         TotalJobs = VALUES(TotalJobs),
524         WaitingJobs = VALUES(WaitingJobs),
525         EstimatedWorstWaitingTime = VALUES(EstimatedWorstWaitingTime),
526         LocalID = VALUES(LocalID),
527         sourceAddr = VALUES (sourceAddr),
528         insertTime = VALUES (insertTime);
529     </xsl:when>
530 </xsl:choose>
531
532 <xsl:variable name="DistinguishedNameGlueSubCluster"><xsl:value-of select="@dn"/></xsl:variable>

```

```

533 <xsl:choose>
534     <xsl:when test="starts-with($DistinguishedNameGlueSubCluster, 'GlueSubClusterUniqueID')">
535         <!-- ObjectClass for Filtering LDAP searches is ObjectClass=GlueSubCluster -->
536         <xsl:variable name="DistinguishedNameLocal">
537             <xsl:value-of select = "substring-after(substring-
538 before($DistinguishedNameGlueSubCluster, ', mds-vo-name=local, o=grid'), 'o-name=')" />
539         </xsl:variable>
540         INSERT IGNORE GLUE20.AdminDomain (ID,informationProvider) VALUES ("<xsl:value-of
541 select="$DistinguishedNameLocal"/>", "<xsl:value-of select="$informationProvider"/>");
542         INSERT IGNORE GLUE20.ComputingService (ID,domainID,informationProvider) VALUES
543 ("<xsl:value-of select="dsml:attr[@name='GlueSubClusterUniqueID']/dsml:value"/>,"<xsl:value-of
544 select="$DistinguishedNameLocal"/>,"<xsl:value-of select="$informationProvider"/>");
545         INSERT IGNORE GLUE20.ComputingManager (ID, serviceID, informationProvider) VALUES
546 ("<xsl:value-of select="dsml:attr[@name='GlueSubClusterUniqueID']/dsml:value"/>,"<xsl:value-of
547 select="dsml:attr[@name='GlueSubClusterUniqueID']/dsml:value"/>,"<xsl:value-of
548 select="$informationProvider"/>");
549         INSERT INTO
550         GLUE20.ExecutionEnvironment
551         (
552         <!-- Name,-->
553         ID,
554         MainMemorySize,
555         VirtualMemorySize,
556         ConnectivityIn,
557         ConnectivityOut,
558         OSName,
559         <!--OSFamily,-->
560         OSVersion,
561         CPUClockSpeed,
562         CPUModel,
563         CPUVendor,
564         PhysicalCPUs,
565         LogicalCPUs,
566         serviceID,
567         cManagerID,
568         informationProvider,
569         sourceAddr,
570         insertTime
571         )

```

```

572     VALUES
573     (
574         <!-- "<xsl:value-of select="dsml:attr[@name='GlueSubClusterName']/dsml:value"/>"-->
575         "<xsl:value-of select="dsml:attr[@name='GlueSubClusterUniqueID']/dsml:value"/>",
576         "<xsl:value-of
577 select="dsml:attr[@name='GlueHostMainMemoryRAMSize']/dsml:value"/>",
578         "<xsl:value-of
579 select="dsml:attr[@name='GlueHostMainMemoryVirtualSize']/dsml:value"/>",
580         "<xsl:value-of
581 select="dsml:attr[@name='GlueHostNetworkAdapterInboundIP']/dsml:value"/>",
582         "<xsl:value-of
583 select="dsml:attr[@name='GlueHostNetworkAdapterOutboundIP']/dsml:value"/>",
584         "<xsl:value-of
585 select="dsml:attr[@name='GlueHostOperatingSystemName']/dsml:value"/>",
586         <!--"<xsl:value-of
587 select="dsml:attr[@name='GlueHostOperatingSystemRelease']/dsml:value"/>"-->
588         "<xsl:value-of
589 select="dsml:attr[@name='GlueHostOperatingSystemVersion']/dsml:value"/>",
590         "<xsl:value-of
591 select="dsml:attr[@name='GlueHostProcessorClockSpeed']/dsml:value"/>",
592         "<xsl:value-of select="dsml:attr[@name='GlueHostProcessorModel']/dsml:value"/>",
593         "<xsl:value-of select="dsml:attr[@name='GlueHostProcessorVendor']/dsml:value"/>",
594         "<xsl:value-of select="dsml:attr[@name='GlueSubClusterPhysicalCPUs']/dsml:value"/>",
595         "<xsl:value-of select="dsml:attr[@name='GlueSubClusterLogicalCPUs']/dsml:value"/>",
596         "<xsl:value-of select="dsml:attr[@name='GlueSubClusterUniqueID']/dsml:value"/>",
597         "<xsl:value-of select="dsml:attr[@name='GlueSubClusterUniqueID']/dsml:value"/>",
598         "<xsl:value-of select=\"$informationProvider\"/>",
599         "<xsl:value-of select=\"$sourceAddress\"/>",
600         <xsl:value-of select=\"$insertTime\"/>
601     )
602 ON DUPLICATE KEY UPDATE
603     ID = VALUES(ID),
604     MainMemorySize = VALUES(MainMemorySize),
605     VirtualMemorySize = VALUES(VirtualMemorySize),
606     ConnectivityIn = VALUES(ConnectivityIn),
607     ConnectivityOut = VALUES(ConnectivityOut),
608     OSName = VALUES(OSName),
609     <!--OSFamily = VALUES(OSFamily),-->
610     OSVersion = VALUES(OSVersion),

```

```

611         CPUClockSpeed = VALUES(CPUClockSpeed),
612         CPUModel = VALUES(CPUModel),
613         CPUVendor = VALUES(CPUVendor),
614         PhysicalCPUs = VALUES(PhysicalCPUs),
615         LogicalCPUs = VALUES(LogicalCPUs),
616         serviceID = VALUES (serviceID),
617         sourceAddr = VALUES (sourceAddr),
618         insertTime = VALUES (insertTime);
619     </xsl:when>
620 </xsl:choose>
621
622 <xsl:variable name="DistinguishedNameGlueSE"><xsl:value-of select="@dn"/></xsl:variable>
623 <xsl:choose>
624     <xsl:when test="starts-with($DistinguishedNameGlueSE, 'GlueSEUniqueID')">
625         <!-- ObjectClass for Filtering LDAP searches is ObjectClass=GlueSE -->
626         <xsl:variable name="DistinguishedNameLocal">
627             <xsl:value-of select = "substring-after(substring-
628 before($DistinguishedNameGlueSE, ',mds-vo-name=local,o=grid'),'o-name=')"/>
629         </xsl:variable>
630         INSERT IGNORE GLUE20.AdminDomain (ID,informationProvider) VALUES ("<xsl:value-of
631 select="$DistinguishedNameLocal"/>","<xsl:value-of select="$informationProvider"/>");
632     INSERT INTO
633     GLUE20.StorageService
634     (
635     ID,
636     Name,
637     domainID,
638     informationProvider,
639     sourceAddr,
640     insertTime
641     )
642     VALUES
643     (
644         "<xsl:value-of select="dsml:attr[@name='GlueSEUniqueID']/dsml:value"/>",
645         "<xsl:value-of select="dsml:attr[@name='GlueSEName']/dsml:value"/>",
646         "<xsl:value-of select="$DistinguishedNameLocal"/>",
647         "<xsl:value-of select="$informationProvider"/>",
648         "<xsl:value-of select="$sourceAddress"/>",

```

```

649         <xsl:value-of select="$insertTime"/>)
650     ON DUPLICATE KEY UPDATE
651         Name = VALUES(Name),
652         ID = VALUES(ID),
653         informationProvider = VALUES(informationProvider),
654         sourceAddr = VALUES (sourceAddr),
655         insertTime = VALUES (insertTime);
656     INSERT INTO
657     GLUE20.Endpoint
658         (
659         ID,
660         Name,
661         URL,
662         Technology,
663         ImplementationVersion,
664         serviceID,
665         informationProvider,
666         sourceAddr,
667         insertTime
668         )
669     VALUES
670         (
671         "<xsl:value-of select='dsml:attr[@name='GlueSEUniqueID']/dsml:value'/>",
672         "<xsl:value-of select='dsml:attr[@name='GlueSEName']/dsml:value'/>",
673         "<xsl:value-of select='dsml:attr[@name='GlueSEPort']/dsml:value'/>",
674         "<xsl:value-of select='dsml:attr[@name='GlueSEArchitecture']/dsml:value'/>",
675         "<xsl:value-of select='dsml:attr[@name='GlueInformationServiceURL']/dsml:value'/>",
676         "<xsl:value-of select='dsml:attr[@name='GlueSEUniqueID']/dsml:value'/>",
677         "<xsl:value-of select='$informationProvider'/>",
678         "<xsl:value-of select='$sourceAddress'/>",
679         <xsl:value-of select="$insertTime"/>
680         )
681     ON DUPLICATE KEY UPDATE
682         Name = VALUES(Name),
683         URL = VALUES(URL),
684         Technology = VALUES(Technology),
685         ImplementationVersion = VALUES(ImplementationVersion),
686         serviceID = VALUES(serviceID),

```

```

687         informationProvider = VALUES(informationProvider),
688         sourceAddr = VALUES (sourceAddr),
689         insertTime = VALUES (insertTime);
690     INSERT IGNORE GLUE20.StorageEndpoint
691 (sServiceID,EndpointID,informationProvider,sourceAddr,insertTime) VALUES ("<xsl:value-of
692 select="dsml:attr[@name='GlueSEUniqueID']/dsml:value"/>","<xsl:value-of
693 select="dsml:attr[@name='GlueSEUniqueID']/dsml:value"/>","<xsl:value-of
694 select="$informationProvider"/>","<xsl:value-of select="$sourceAddress"/>","<xsl:value-of
695 select="$insertTime"/>");
696     </xsl:when>
697 </xsl:choose>
698
699 <xsl:variable name="DistinguishedNameGlueService"><xsl:value-of select="@dn"/></xsl:variable>
700 <xsl:choose>
701     <xsl:when test="starts-with($DistinguishedNameGlueService, 'GlueServiceUniqueID')">
702     <!-- ObjectClass for Filtering LDAP searches is ObjectClass=GlueService -->
703     <xsl:variable name="DistinguishedNameLocal">
704         <xsl:value-of select = "substring-after(substring-
705         before($DistinguishedNameGlueService,'mds-vo-name=local,o=grid'),'o-name=')" />
706     </xsl:variable>
707     INSERT IGNORE GLUE20.AdminDomain (ID,informationProvider) VALUES ("<xsl:value-of
708 select="$DistinguishedNameLocal"/>","<xsl:value-of select="$informationProvider"/>");
709     INSERT IGNORE GLUE20.ComputingService
710 (ID,name,type,qualityLevel,complexity,domainID,informationProvider) VALUES ("<xsl:value-of
711 select="dsml:attr[@name='GlueServiceUniqueID']/dsml:value"/>","<xsl:value-of
712 select="dsml:attr[@name='GlueServiceName']/dsml:value"/>","<xsl:value-of
713 select="dsml:attr[@name='GlueServiceType']/dsml:value"/>","<xsl:value-of
714 select="dsml:attr[@name='GlueServiceWSDL']/dsml:value"/>","<xsl:value-of
715 select="dsml:attr[@name='GlueServiceSemantics']/dsml:value"/>","<xsl:value-of
716 select="$DistinguishedNameLocal"/>","<xsl:value-of select="$informationProvider"/>");
717     INSERT INTO
718     GLUE20.Endpoint
719     (
720     serviceID,
721     ID,
722     Name,
723     URL,
724     Technology,
725     implementationVersion,
726     healthState,

```

```

727         healthStateInfo,
728         startTime,
729         informationProvider,
730         sourceAddr,
731         insertTime
732     )
733     VALUES
734     (
735         "<xsl:value-of select='dsml:attr[@name='GlueServiceUniqueID']/dsml:value'/>",
736         "<xsl:value-of select='dsml:attr[@name='GlueServiceEndpoint']/dsml:value'/>",
737         "<xsl:value-of select='dsml:attr[@name='GlueServiceName']/dsml:value'/>",
738         "<xsl:value-of select='dsml:attr[@name='GlueServiceEndpoint']/dsml:value'/>",
739         "<xsl:value-of select='dsml:attr[@name='GlueServiceType']/dsml:value'/>",
740         "<xsl:value-of select='dsml:attr[@name='GlueServiceVersion']/dsml:value'/>",
741         "<xsl:value-of select='dsml:attr[@name='GlueServiceStatus']/dsml:value'/>",
742         "<xsl:value-of select='dsml:attr[@name='GlueServiceStatusInfo']/dsml:value'/>",
743         "<xsl:value-of select='dsml:attr[@name='GlueServiceStartTime']/dsml:value'/>",
744         "<xsl:value-of select='$informationProvider'/>",
745         "<xsl:value-of select='$sourceAddress'/>",
746         "<xsl:value-of select='$insertTime'/>"
747     )
748     ON DUPLICATE KEY UPDATE
749         Name = VALUES(Name),
750         URL = VALUES(URL),
751         Technology = VALUES(Technology),
752         ImplementationVersion = VALUES(ImplementationVersion),
753         informationProvider = VALUES(informationProvider),
754         sourceAddr = VALUES (sourceAddr),
755         insertTime = VALUES (insertTime);
756     INSERT INTO
757     GLUE20.ComputingEndpoint
758     (
759         endpointID,
760         staging,
761         cServiceID,
762         informationProvider,
763         sourceAddr,
764         insertTime

```

```

765         )
766     VALUES
767     (
768         "<xsl:value-of select='dsml:attr[@name='GlueServiceEndpoint']/dsml:value'/>",
769         "<xsl:value-of select='dsml:attr[@name='GlueServiceStatus']/dsml:value'/>",
770         "<xsl:value-of select='dsml:attr[@name='GlueServiceUniqueID']/dsml:value'/>",
771         "<xsl:value-of select='$informationProvider'/>",
772         "<xsl:value-of select='$sourceAddress'/>",
773         "<xsl:value-of select='$insertTime'/>"
774     )
775     ON DUPLICATE KEY UPDATE
776         staging = VALUES(staging),
777         endpointID = VALUES(endpointID),
778         informationProvider = VALUES(informationProvider),
779         sourceAddr = VALUES (sourceAddr),
780         insertTime = VALUES (insertTime);
781     </xsl:when>
782 </xsl:choose>
783 </xsl:template>
784 </xsl:stylesheet>

```

5 Referenzen

- [1] Softwaretechnische Analyse und Design, D-MON Project Deliverable, Oktober 2008
http://www.d-grid.de/fileadmin/user_upload/documents/D-MON/Design-AP1.3.final.pdf
- [2] Spezifikation Middlewareübergreifender Monitoring und VO-Integrations Tests, D-MON Projekt Deliverable, Dezember 2008,
http://www.d-grid.de/fileadmin/user_upload/documents/D-MON/AP2_4.pdf
- [3] Sammlung von Anforderungen, D-MON Projekt Deliverable, Januar 2008
http://www.d-grid.de/fileadmin/user_upload/documents/D-MON/Anforderungsanalyse-Version1.0.pdf