



Grid Management and Monitoring

Luděk Matyska

ludek@ics.muni.cz

Institute of Computer Science, Masaryk University Brno

Czech Republic



Overview



- Grid monitoring
 - An information source for Grid management
 - Resource-centric monitoring
 - State of the resources
 - State of the services
 - Job-centric monitoring
 - Job flow through the Grid
- Grid Management
 - Steering of the infrastructure



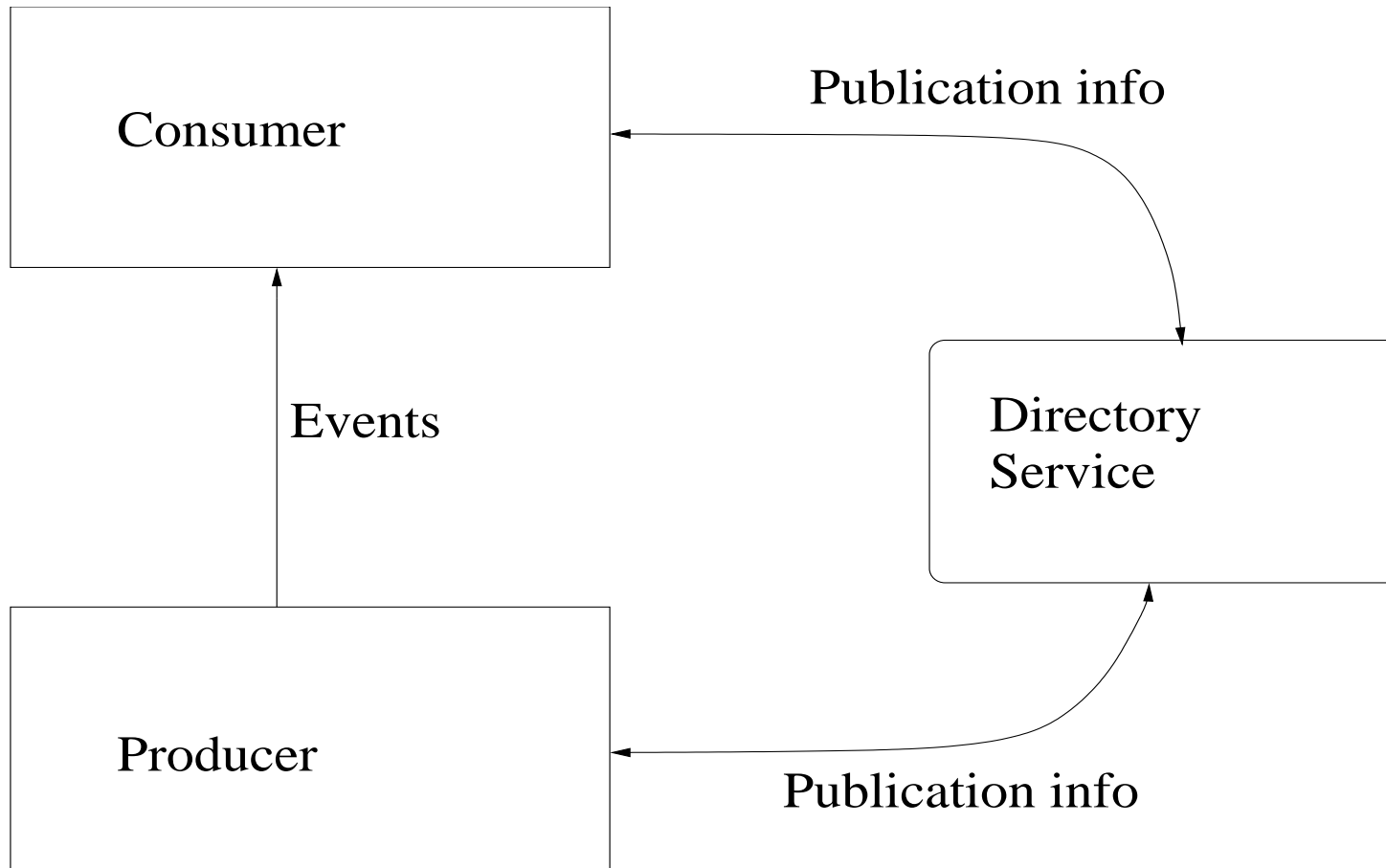
Resource Centric Grid Monitoring



Information Society
Technologies

- Grid Monitoring Architecture (GMA)
 - GGF activity
 - Basic components of GMA
 - Producers
 - Consumers
 - Directory service

GMA basic structure





GMA actual state



- Focus on transfer of monitoring info
 - Content independence
 - Usually “push” model
 - Security and persistence not primary focus (streaming data)



R-GMA



- Relational extensions of the GMA
 - Can support general SQL queries
- Developed within the DataGrid project
- Resilience through specific producers
- Java plus Tomcat/Apache based
 - C version under development
- Just basic security support



GridLab monitoring system



- Developed in SZTAKI (Hungary)
- All infrastructure components
- Focus on efficiency and scalability
- Secure transport layer



Testbed Organization for GridLab



Information Society
Technologies

- Grid Testbed Operation Center (GTOC): Masaryk University Brno
 - Testbed status monitoring
 - Bug tracking (Bugzilla server)
 - Problem escalation (only manually)
- Portal
 - User and administrative portals unified
- Information services
 - Each site runs its own GIS (MDS-2) with default schema
 - Local GISes register to GIIS in Brno (`mds.gridlab.org`)—master server



GridLab status testing



- Test availability of defined “services”
 - Components of GT2 (currently 2.2.4)
 - Application specific components
- Centrally controlled
 - Full suite of tests runs every hour
 - Results are displayed on the portal page(s)
 - Reasons of failure also provided
- Test interdependencies



Individual tests



- Specific test for each component
- Java implementation
- Tested services:
 - Gatekeeper, GIIS, GRIS
 - MDS service (GridLab)
 - Monitoring (GridLab)
 - GSIFTP, GSISSH
 - Software
 - CA
 - Jobmanagers
 - MPI availability

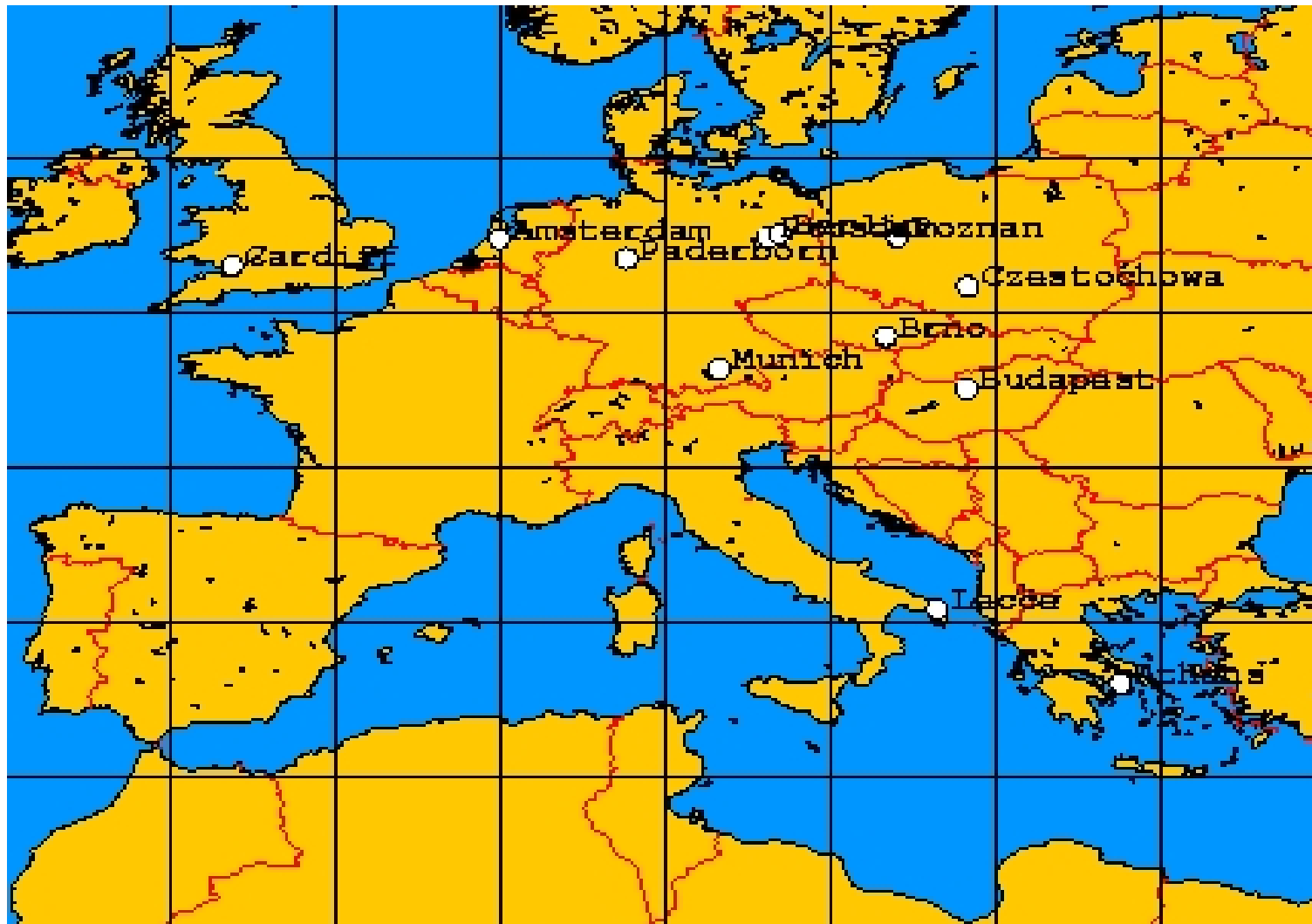


Current status



- Can be seen at <http://www.gridlab.org/WorkPackages/wp-5/testbed/notes.html>
- 18 machines/clusters
 - 9 countries
 - 13 institutions

Testbed Map



Detailed Status

File Edit View Go Bookmarks Tools Window Help

Back Forward Reload Stop file:///home/ludek/gttest.html Search Print

Home Bookmarks Confs Linux Sites Tech HPCN IS e-Pub MU CZ

Testbed status

Generated: Tuesday, October 28, 2003 9:22:26 AM UTC

Test results

no.	machine	GIS	MDS ext	GRIS	GRMS	MDS WS	monitoring	Gatekeeper	FTP	CA	mapfile	GSISSH	software	Jobmanagers	mpicc	MPI-C	mpif77	MPI-f77
1	rage1.man.poznan.pl	OK	OK	OK	OK	OK	OK	OK	OK	OK	OK	OK	OK	jobmanager-fork jobmanager-pbs	OK	jobmanager-fork jobmanager-pbs	OK	jobmanager-fork jobma
2	elforo.pcz.pl	OK	OK	OK	OK	fail	OK	OK	OK	OK	OK	OK	OK	jobmanager-fork jobmanager-condor	OK	jobmanager-fork	OK	jobmanager-fork
3	fs0.das2.cs.vu.nl	OK	OK	OK	fail	OK	OK	OK	OK	OK	OK	OK	OK	jobmanager-fork jobmanager-pbs	OK	jobmanager-fork jobmanager-pbs	OK	jobmanager-fork jobma
4	grape.man.poznan.pl	OK	OK	OK	OK	fail	OK	OK	OK	OK	OK	OK	OK	jobmanager-fork	OK	jobmanager-fork	OK	jobmanager-fork
5	hitcross.lrz-muenchen.de	OK	OK	OK	OK	OK	OK	OK	OK	OK	OK	OK	OK	jobmanager-fork	OK	jobmanager-fork	fail	prereg
6	skirtics.muni.cz	OK	OK	OK	OK	OK	OK	OK	OK	OK	OK	OK	OK	jobmanager-fork jobmanager-pbs	fail	prereg	fail	prereg
7	n0.hpc.szfaki.hu	OK	OK	OK	fail	fail	OK	OK	OK	OK	OK	OK	OK	jobmanager-condor jobmanager-fork	OK	jobmanager-fork	OK	jobmanager-fork
8	gridentry.uni-paderborn.de	OK	OK	OK	fail	OK	OK	OK	OK	OK	OK	OK	OK	jobmanager-fork jobmanager-cc3	timeout	prereg	fail	prereg
9	onyx3.zib.de	OK	fail	OK	OK	fail	fail	OK	OK	OK	OK	OK	OK	prereg	OK	prereg	OK	prereg
10	sr8000.lrz-muenchen.de	OK	OK	OK	fail	fail	fail	OK	OK	OK	OK	fail	fail	jobmanager-fork	timeout	prereg	OK	jobmanager-fork
11	mike4.lsu.edu	OK	fail	OK	fail	OK	fail	OK	OK	OK	OK	OK	OK	prereg	OK	prereg	OK	prereg
12	sierra0.unile.it	fail	fail	fail	fail	fail	fail	OK	OK	OK	fail	OK	OK	prereg	OK	prereg	OK	prereg
13	litchi.zib.de	fail	OK	OK	fail	OK	OK	fail	OK	prereg	prereg	OK	prereg	prereg	prereg	prereg	prereg	prereg
14	bouscat.cs.cf.ac.uk	OK	fail	OK	OK	fail	fail	OK	OK	fail	fail	OK	fail	jobmanager-fork jobmanager-fork	fail	prereg	fail	prereg
15	inca.cf.ac.uk	fail	fail	fail	fail	fail	fail	OK	OK	fail	fail	fail	fail	prereg	fail	prereg	fail	prereg
16	origin.aei-potsdam.mpg.de	fail	fail	timeout	fail	fail	timeout	fail	fail	prereg	prereg	timeout	prereg	prereg	prereg	prereg	prereg	prereg
17	pclab120.telecom.ece.ntua.gr	fail	fail	timeout	fail	fail	timeout	fail	fail	prereg	prereg	timeout	prereg	prereg	prereg	prereg	prereg	prereg
18	minimike.lsu.edu	fail	fail	fail	fail	fail	timeout	fail	fail	prereg	prereg	fail	prereg	prereg	prereg	prereg	prereg	prereg

Machine list

no.	machine	admin	jobmanagers	os	cpu
1	rage1.man.poznan.pl	Bogdan Ludwiczak	jobmanager-fork jobmanager-pbs	Linux 2.4.20-8smp	2x GenuineIntel - 1396 MHz
2	elforo.pcz.pl	Konrad Karczewski	jobmanager-fork jobmanager-condor	Linux 2.4.20	2x AuthenticAMD - 1666 MHz



Status—Error messages



● eltoro.pcz.pl

● MDS WS:

; nested exception is:

java.net.ConnectException: Connection refused

● fs0.das2.cs.vu.nl

● GRMS:

; nested exception is:

java.net.SocketTimeoutException: Read timed out



Web Services Tests



- Calls a method on each service
 - OK: the SOAP call succeeded and a value was returned
 - FAIL: the SOAP call was not successful (reason is displayed)
- Needs correct WSDL to include a new service



Tested web services



Name	Namespace
Scenario broker	urn:resmgmt.gridlab.org
Adaptive service	urn:Adaptive
Metadata service	urn:StorageBoxGridServer
Replica catalog	urn:csrdms
DATA movement	urn:DATA_movement_services
DATA browsing	urn:DATA_browsing_services
Authorization	urn:as_server
Message box service	urn:service.messagebox.psnc.pl
TestbedStatus	urn:testbed:results



Data movement service



- n^2 problem
- Currently full mesh really tested
 - Not a scalable solution
- In fact many dependencies automatically tested
 - Accessibility/Firewalls
 - Credentials
 - Authentication services

File Edit View Go Bookmarks Tools Window Help

Back Forward Reload Stop file:///home/ludek/datamovement.html Search Print

Home Bookmarks Confs Linux Sites Tech HPCN IS e-Pub MU CZ

Results for DATA movement service:

Tue Oct 28 05:42:30 CET 2003

DATACopyFile	init	skirit	hitcross	sr8000	onyx3	litchi	gridentry	rage1	grape	eltoro	n0	fs0	bouscat	inca	sierra0	mike4	minimike
skirit.ics.muni.cz	OK	OK	OK	OK	OK	OK	OK	OK	OK	OK		OK	OK	OK	OK	OK	
hitcross.lrz-muenchen.de		OK	OK	OK	OK	OK	OK	OK	OK	OK		OK	OK	OK	OK	OK	
sr8000.lrz-muenchen.de		OK	OK	OK	OK	OK	OK	OK	OK	OK		OK	OK	OK	OK	OK	
onyx3.zib.de		OK	OK	OK	OK	OK	OK	OK	OK	OK		OK	OK	OK	OK	OK	
litchi.zib.de		OK	OK	OK	OK	OK	OK	OK	OK	OK		OK	OK	OK	OK	OK	
gridentry.uni-paderborn.de		OK	OK	OK	OK	OK	OK	OK	OK	OK		OK	OK	OK	OK	OK	
rage1.man.poznan.pl		OK	OK	OK	OK	OK	OK	OK	OK	OK		OK	OK	OK	OK	OK	
grape.man.poznan.pl		OK	OK	OK	OK	OK	OK	OK	OK	OK		OK	OK	OK	OK		
eltoro.pcz.pl		OK	OK	OK	OK	OK	OK	OK	OK	OK		OK	OK	OK	OK	OK	
n0.hpcc.sztaki.hu																	
fs0.das2.cs.vu.nl		OK	OK	OK	OK	OK	OK	OK	OK	OK		OK	OK	OK	OK		
bouscat.cs.cf.ac.uk		OK	OK	OK	OK	OK	OK	OK	OK	OK		OK	OK	OK	OK		
inca.cf.ac.uk		OK	OK	OK	OK	OK	OK	OK	OK	OK		OK	OK	OK	OK	OK	
sierra0.unile.it		OK	OK	OK	OK	OK	OK	OK	OK	OK		OK	OK	OK	OK	OK	
mike4.lsu.edu		OK		OK	OK	OK	OK	OK	OK	OK		OK	OK	OK	OK	OK	
minimike.lsu.edu																	

```

init->hitcross.lrz-muenchen.de
null
init->sr8000.lrz-muenchen.de
null
init->onyx3.zib.de
null
  
```



SC2002 Demo



- Largest heterogeneous Grid
- Some statistics
 - 69 sites, 14 countries, 4 continents
 - 7345 CPUs, 3469 actually available for the demo
- Status monitoring
 - Centralized
 - Simplified GridLab tests
- Manual management



Job centric monitoring



- Part of the DataGrid Workload Management System
- Logging and bookkeeping service
 - Monitors flow of a job through the Grid
 - Requires instrumentation of all components
- Independent infrastructure
 - Preceded (R-)GMA implementation
 - Strong security requirements
 - Reliability/persistence
 - Data repository part of the design



Logging and Bookkeeping Service



Information Society
Technologies

- Event based
 - Generated by individual WMS components
 - Reliably transmitted into a database
- Job state constructed on the fly
 - State automaton
- Analogous to the GMA

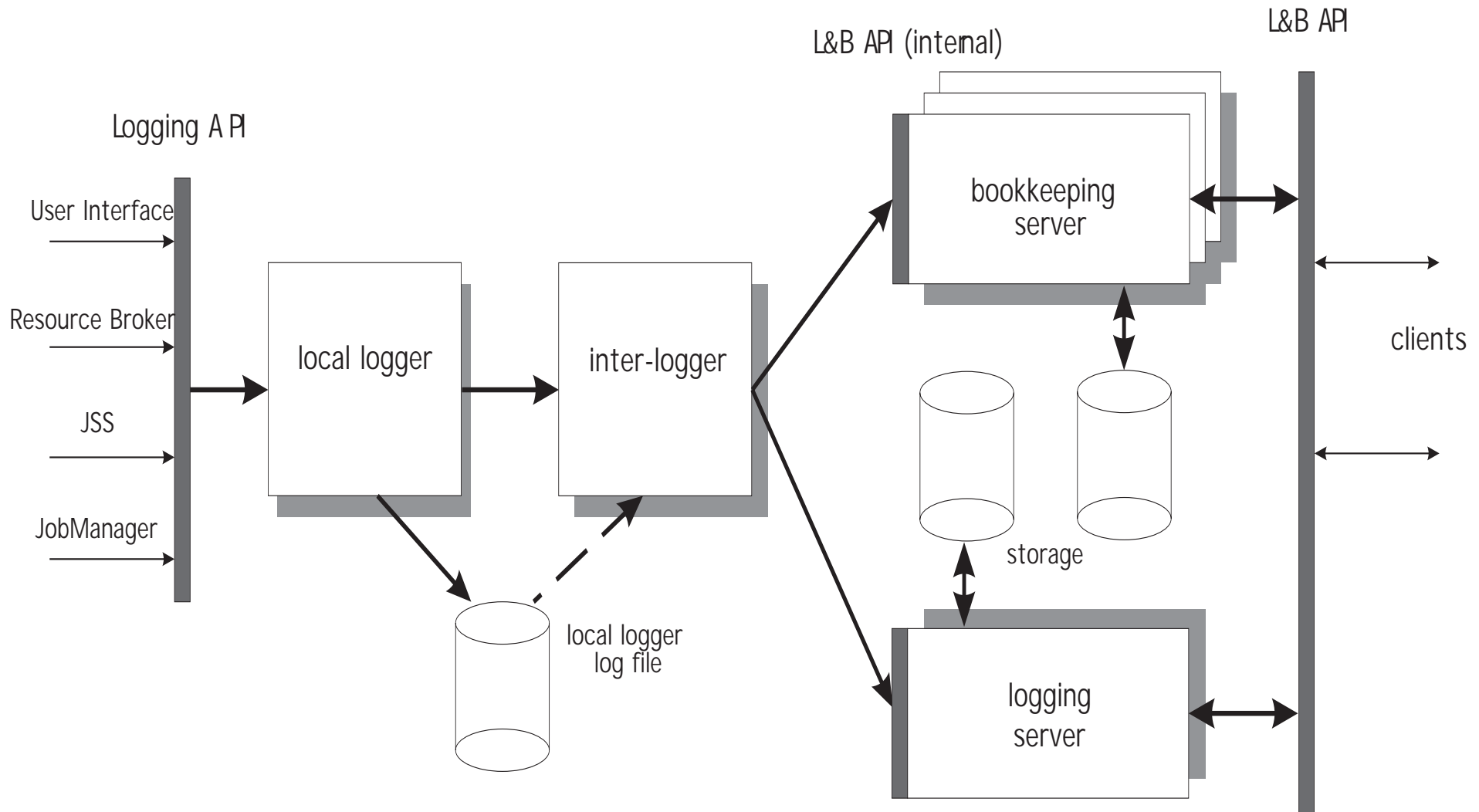


LB components



- Basic components
 - Local logger
 - Inter logger
 - Bookkeeping server
- Asynchronous delivery
 - non-blocking calls
 - persistence through local logger's files
- Synchronous (blocking) calls for special purposes

LB architecture





LB security



- All transmission encrypted
 - Extensive use of SSL
 - Sessions for efficiency improvement
- Authentication based on certificates
 - Anyone with a valid certificate can log events



Authorization control in LB



- First event defines job owner
- Only this “person” can log events for this job
- Exception—certificate expired within the WMS
 - Host certificates used to log last (abort) event(s)

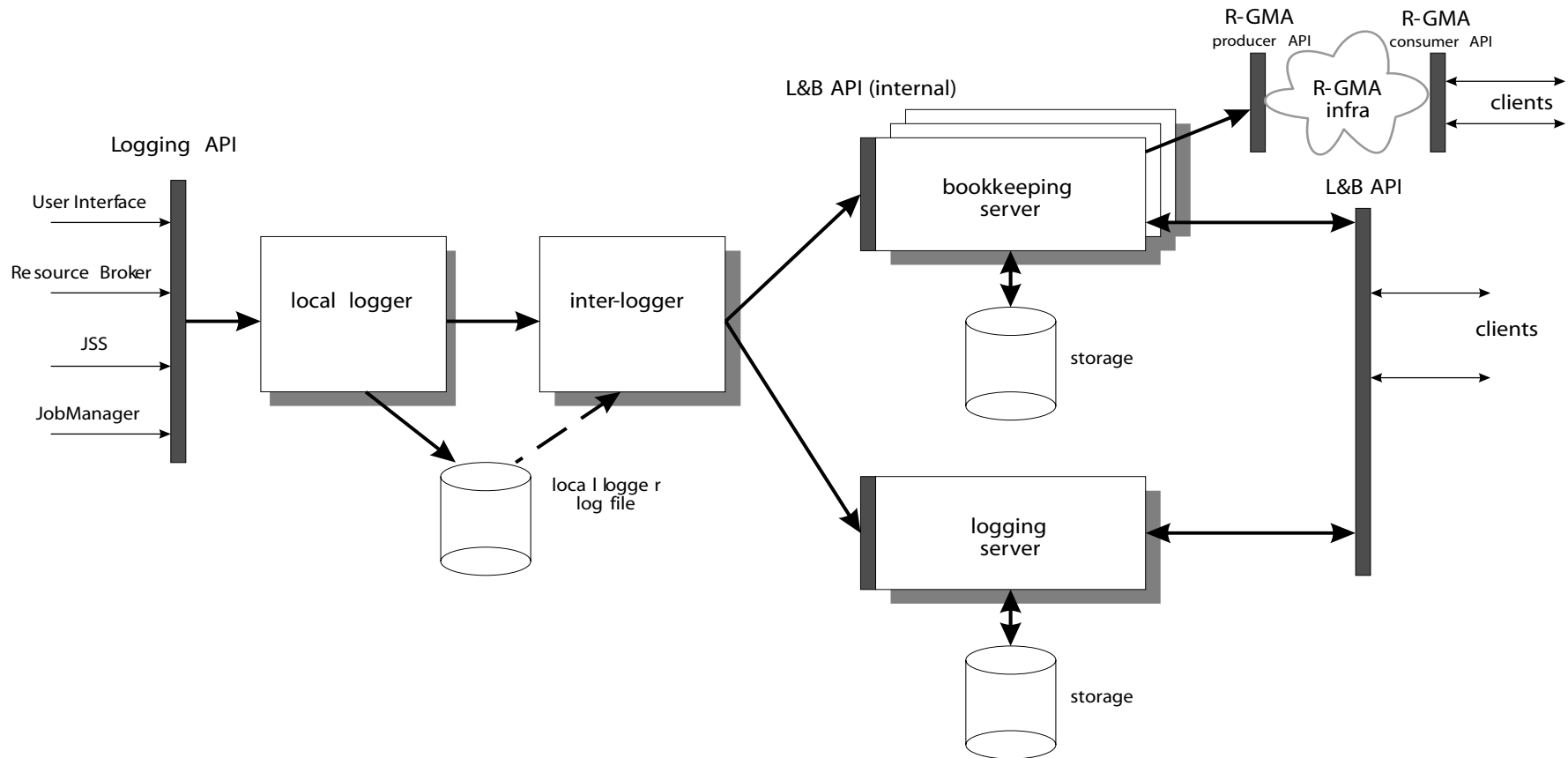


LB and R-GMA



- R-GMA not a replacement
 - Lack of security
 - Untested reliability
 - Rather heavy-weight components
- Job state delivery to users
 - Decreases load on the bookkeeping server
 - Notification support
 - Still limited use due to the security constraints

LB architecture





Open Problems



- Monitoring—Centralized approach
 - Does not scale, overloads the collecting site
 - Creates a single point of failure
 - Potentially long reaction time
- Identity of the monitoring entity
- Reliability of collected data
- Management—VO site management
 - Integration of local escalation procedures



Some solutions



- Monitoring worm
 - Check for interdependencies
 - Application-like view of the Grid
- Test service certificates
 - Special certificates
 - All tests must be able to run under any identity
 - Application instrumentation



Monitoring Worm



- An application-like monitoring tool
 - Uses the same APIs and environment as any application
 - Re-spawns itself through the resource broker
 - Migrates (randomly or using a regular pattern) through the Grid
 - “Knows” services and tries to use them
 - Filters and post-process results and sends them to the collecting site(s)
 - Checks on monitors (an independent monitoring tool)
 - If available, connects to the local monitoring infrastructure and compares its own results



Monitoring Worm—features



- Build on top of GAT (Grid Application Toolkit)
- Runs under service/test certificate or on user request (with user certificate)
 - Initiated through the portal
 - Users can trigger its run to help localize a problem (worm checks the environment on behalf of user)
- Always reports to the collecting site (even from users' runs)



Lessons Learned—Monitoring



- Cannot rely on local setup
 - Needs more independent monitoring info sources
 - Needs models and frameworks for monitoring info correlation and cross checking
- Relationship with information services
 - Some overlap may be advantageous



Lessons Learned—Security



- Essential, but often underrated
- Missing truly scalable solutions (SSL/TSL does not scale)
- Authorization
 - What a user can see?
 - Are all administrators equal?
- Virtual organizations support



Lessons Learned—Management



Information Society
Technologies

- Grid ticket tracking system missing
- Escalation procedures must be automated and integrated with local usage
- Access to local resources for developers looking for an error
 - Logs usually are not enough
 - Site setup restoration
 - How to find slight differences in setup?
 - Alternative approaches must be looked for



Lessons Learned—Other



● Firewalls

- Must find a way to integrate firewalls into Grid setup
- Firewall status monitoring—is it possible?

● Reliable repositories

- How long to store the monitoring data and managerial decisions
- Need for a Grid solution? :-)

- Hierarchical setup
 - Information processed locally
 - Only digests and anomalies sent to higher levels
 - Better test interdependency specification (semi-automatic?)
- Monitoring worm
 - Modular structure (extensibility)
 - (Semi)automatic service discovery
 - Use of monitoring APIs on services (when they will be available)

- GAT = Grid Application Toolkit
 - To “hide” Grid from the developer
- Ideal for automatic monitoring
 - Instrumentation of GAT calls
 - GAT services with monitoring API
- Use GMA to collect the produced data



Future work—Job centric monitoring



Information Society
Technologies

- Better integration with the (R-)GMA
- Full authorization
- Full notification service (with dynamic authorization)
- Collective operations (sets of jobs)
- Full logging support (i.e. long term searchable repositories)
- Grid performance monitoring



Future Work—Management



- Grid quality and performance estimation
- SLAs (Service Level Agreement) definition and monitoring
- Locality and subsidiarity, i.e. hierarchical decision support
- Better integration of monitoring and management/steering infrastructures
 - Each decision should be logged
 - Grid state roll back? (per site)



The research group



- Masaryk University (Brno)
 - Miroslav Ruda, Martin Kuba, Petr Holub, Aleš Křenek, Zdeněk Salvét, Daniel Kouřil, Jiří Škrabal
- Charles University (Prague)
 - Michal Voců
- West Bohemia University (Pilsen)
 - Jiří Sitera, Jan Pospíšil



Dziękuję



Questions?