Standards-based Access to LFCs

Karolina Sarnowska, University of Virginia
Andrew Grimshaw, University of Virginia
Mark Morgan, University of Virginia
Akos Frohner, CERN
Erwin Laure, KTH

Outline

- Introduction
- Development
- Interoperability
- Performance
- Conclusion

Interoperability and Grid Standards

INTRODUCTION

Goal & Motivation

- Create a standards-based interface for accessing data resources in EGEE grids
- Read data from LFCs via standardized interface
- Enable interoperability
- Validate specifications

LFC: The LCG File Catalog

- Catalog of data files on storage elements
- Each logical file entry can correspond to multiple physical files (replicas)
- Provides mapping between logical file names and storage URLs

LFC

/grid
/atlas
/alice
/cms

Storage Elements

EGEE Grid
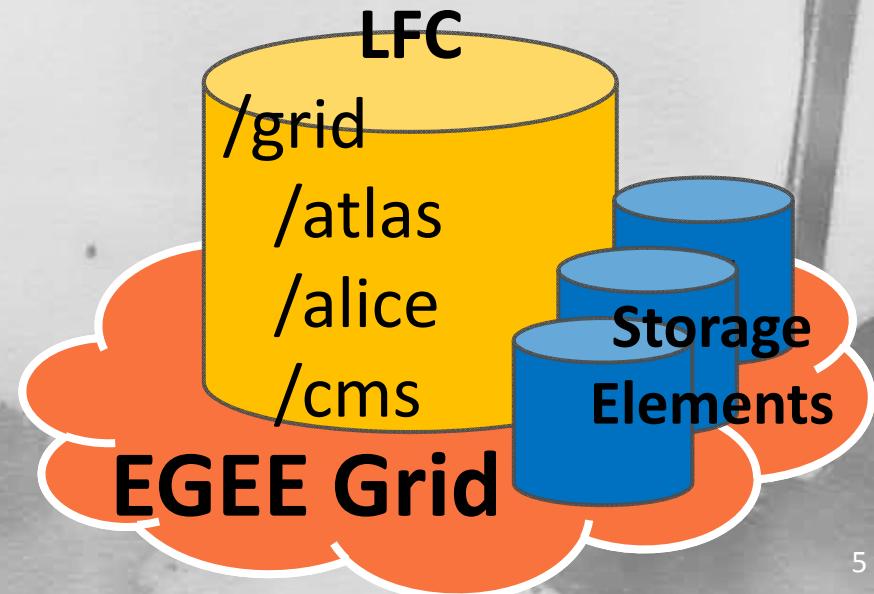
OGF Standards

- Infrastructure
  WS-Addressing, **Resource Namespace Service (RNS)**, WS-Naming
- Compute
  JSDL, OGSA-BES, HPC Basic Profile
- Data
  RNS, **OGSA-ByteIO**, gridFTP, WS-DAI
- APIs
  SAGA, DRMAA, GridRPC
- Security
  WSI-BSP, WS-Security, WS-Trust, WS-Federation, WSSecurityPolicy, WS-SecureNaming, WS-Secure Communication

Step One: Identify Resource

- Need means of referring to resources

- What's in a name?
  - human-readable
  - maps to unique resource

The RNS Specification

■ RNS: Resource Namespace Service

■ Describes standard way of mapping names to grid endpoints

■ RNS 1.0 Operations:

   Add, List, Remove, Query, Move

The SNARL Service:
An Implementation of RNS

Standards-based **N**aming for **A**ccessing **R**esources in an **L**FC

LFC

/grid
/atlas
/alice
/cms
/dtear

EGEE Grid

Another Grid

Grid Client

(1) RNS Requests

SNARL Service

(2) LFC API Calls

Storage Elements

(3) RNS Responses

9
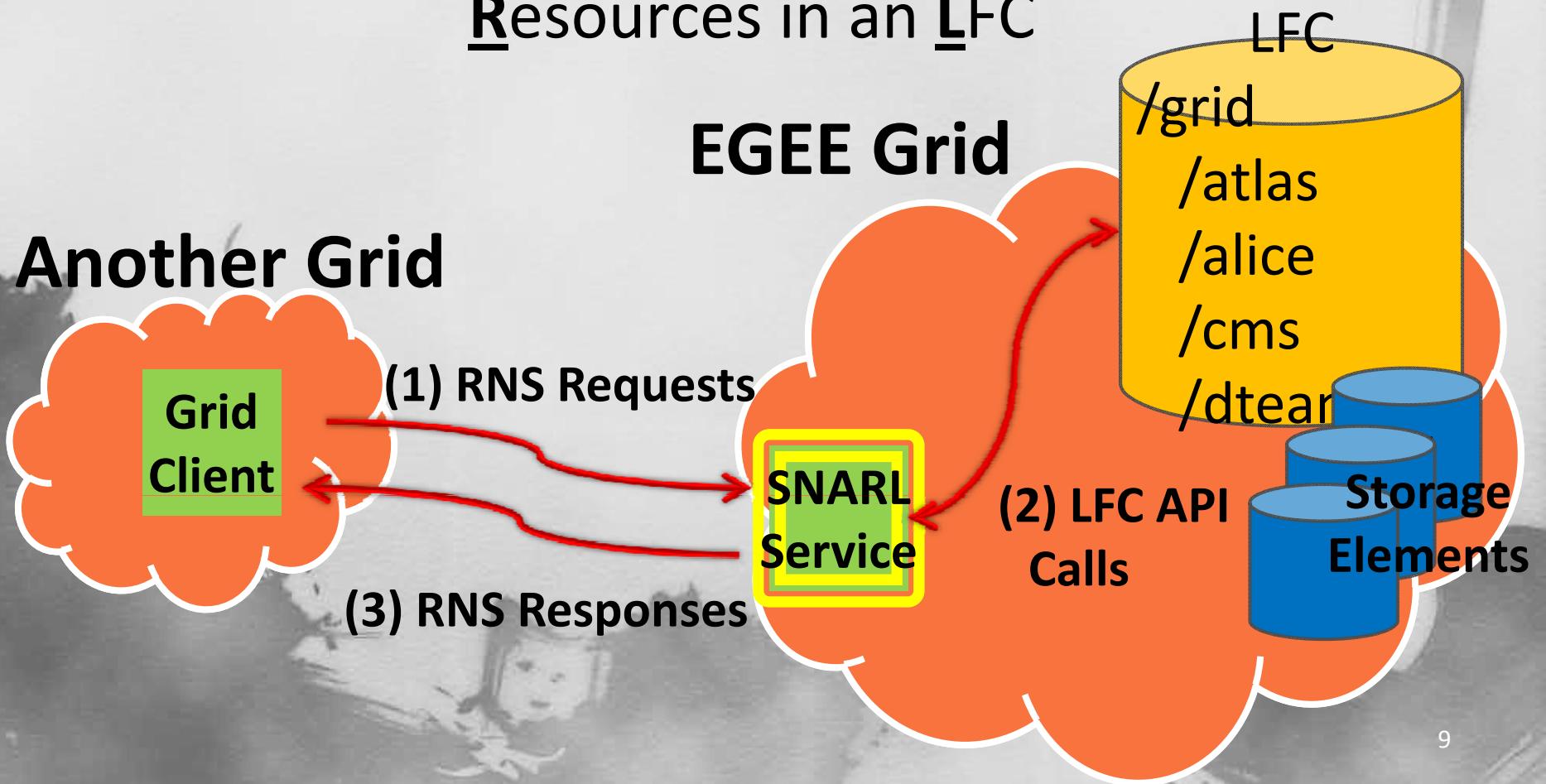
Step Two: Access Resource Data

■ Need means of accessing data associated with resources

The ByteIO Specification

- Describes standard way of handling transfer of data associated with grid entities
- POSIX-like interface
- Random, session-less or stateful session access
- Operations: Read, Write, Append, TruncAppend, SeekRead, SeekWrite

The SABLE Service:
An Implementation of ByteIO

Standards-based **A**ccess to **B**ytes of **L**FC **E**ntries

EGEE Grid

LFC

/grid

/atlas

/alice

/cms

/dteam

Another Grid

(1) ByteIO Requests

SABLE Service

Client

(3) ByteIO Responses

(2) GFAL API Calls

Storage Elements
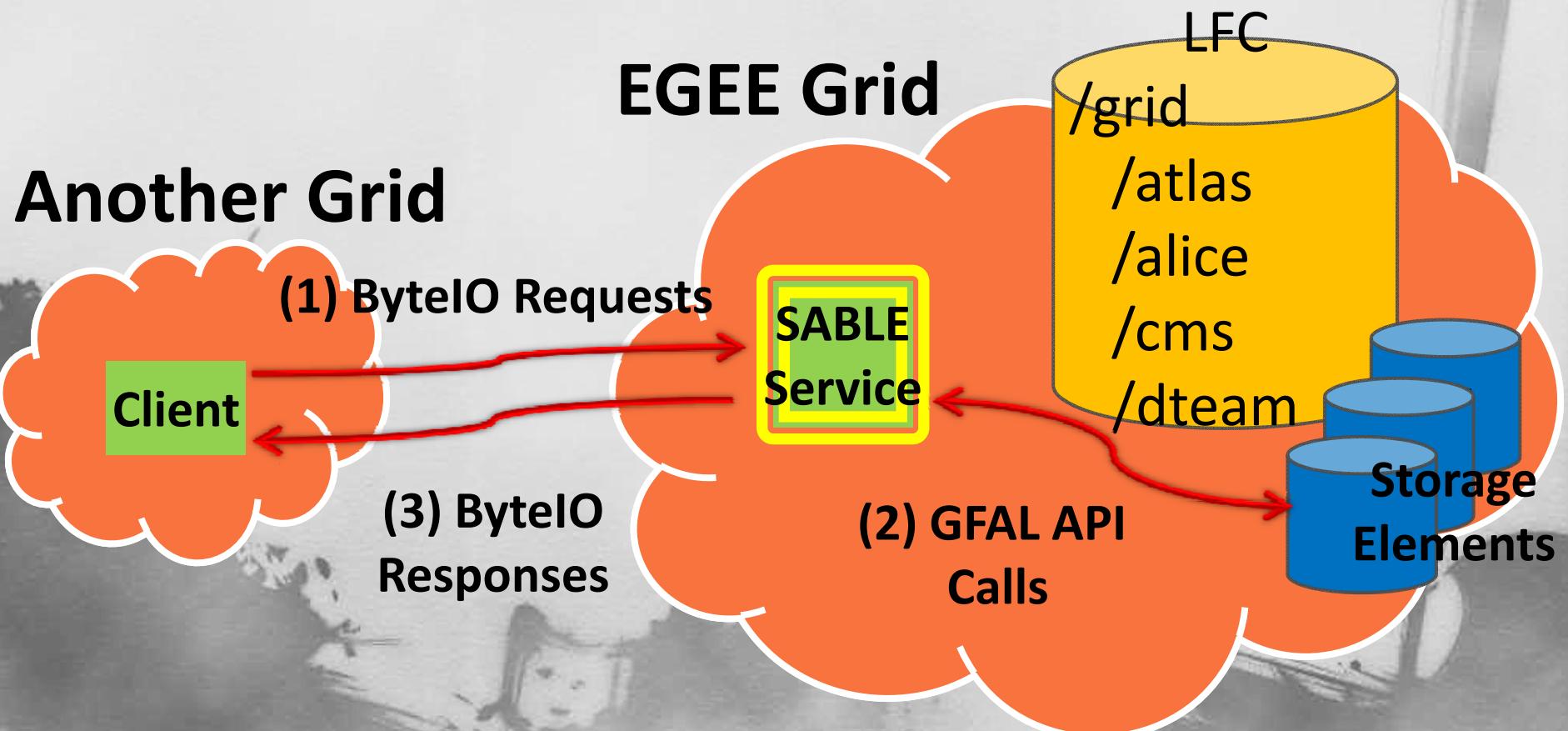
Creating web services with Axis2/C

DEVELOPMENT

Development Process

1. Setup web services framework with Axis2/C

2. Create skeleton services with WSDL2C

3. Create clients for testing

4. **Implement RNS and ByteIO Operations using gLite APIs**

Why Apache Axis2/C?

- Web services engine implemented in C

- Handles SOAP message processing

- Built-in support includes:
  - WS-Addressing
  - WS-Policy
  - WS-SecurityPolicy

Service Skeleton Generation with WSDL2C Tool

1. Specify web service in WSDL file

2. Use tool to generate C web service skeleton

Resulting Service Skeleton Size:

SNARL: 89k LOC

SABLE :   77k LOC

Test Client Creation

■ Example: RNS Add Request

```
<soap:Envelope>
 <soap:Header>
   <wsa:Action>http://schemas.ggf.org/rns/2006/03/rns/add</wsa:Action>
 </soap:Header>
<soap:Body>
  <rns:add xmlns:rns='http://schemas.ggf.org/rns/2006/05/rns'>
   <rns:entry_name> foo </rns:entry_name>
  </rns:add>
 </soap:Body>
</soap:Envelope>
```

Implementing RNS Ops

- Translating RNS operations to API calls

RNS Operation	LFC API Call
Add	lfc_creat
List	lfc_readdir
Remove	lfc_unlink
Query	lfc_stat
Move	lfc_unlink + lfc_creat

- Should replicas be visible to users?

Revealing Logical Files vs. Replicas

(1) RNS List

Grid Client

SNARL Service

(3) RNS List Response

(2) LFC API Calls

SE •fooReplicaA

SE •fooReplicaB

LFC •/grid/foo

19
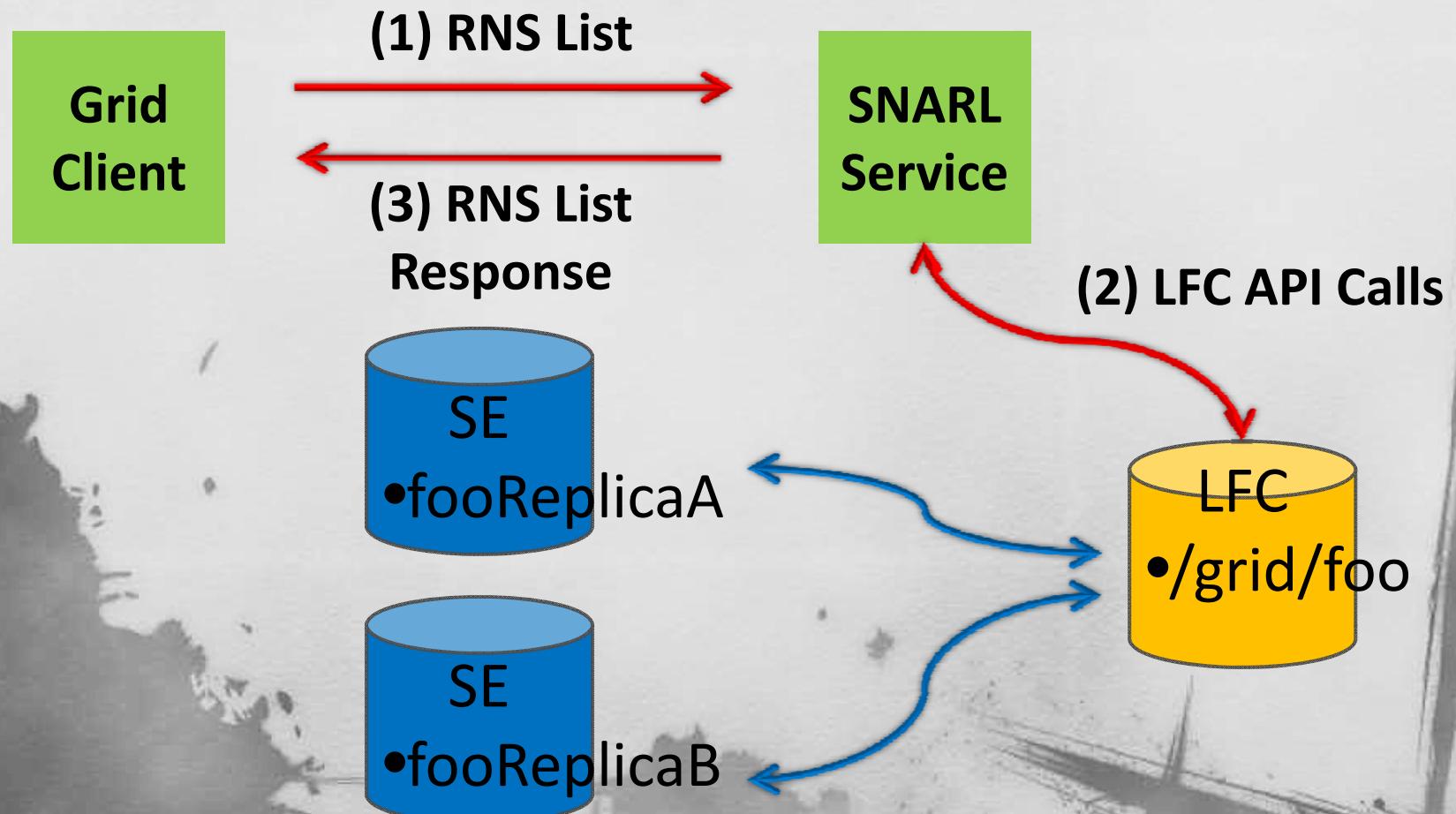
Implementing ByteIO Ops

- Translating ByteIO operations to API calls

ByteIO Operation	API Call
All	gfal_open
Read	gfal_lseek + gfal_read
Write/Append	gfal_lseek + gfal_write
TruncAppend	gfal_creat + Append

- Should writes be allowed?

INTEROPERABILITY TESTS

Spec Adoption

Project/Spec	RNS	ByteIO
Genesis II	X	X
Unicore 6		X
Fujitsu USMT		X
Microsoft CCS		
Globus		
OMII-UK		X
GridSAM		
Crown		
Platform		
OGSA-DAI	X	X
NAREGI		
gFarm	X	

# Interoperability Test Setup

1. Using RNS list request, choose LFC entry
2. Using ByteIO read request, read data resource associated with LFC entry

Step One:
List LFC Entries via SNARL

GenesisII Grid

EGEE Grid

Grid Client

LFC

/grid
/atlas
/alice
/cms
/dteam

(1) RNS List Request

(2) LFC API Calls

(3) RNS List Response

SNARL Service

Storage Elements

24
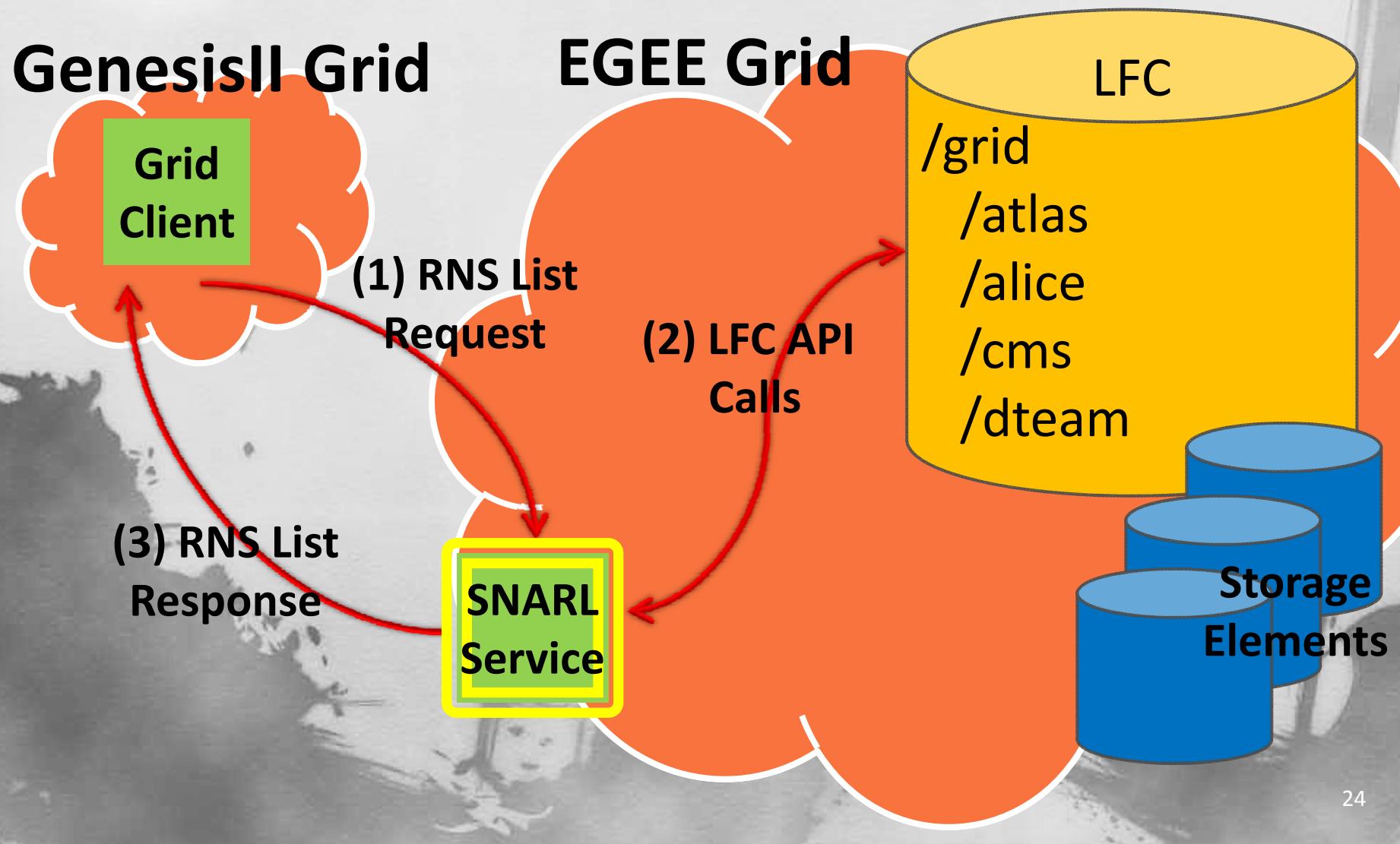
Step Two:
Read Resource Data via SABLE

GenesisII Grid

EGEE Grid

LFC

/grid
/atlas
/alice
/cms
/dteam

Grid Client
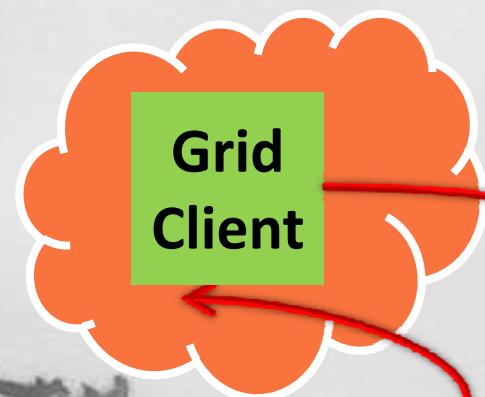
(1) ByteIO Read Request

SABLE Service

(3) ByteIO Read Response

(2) GFAL API Calls

Storage Elements
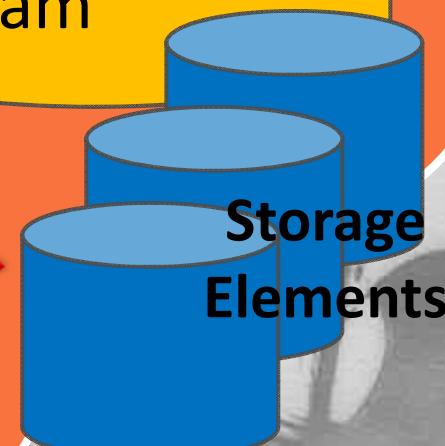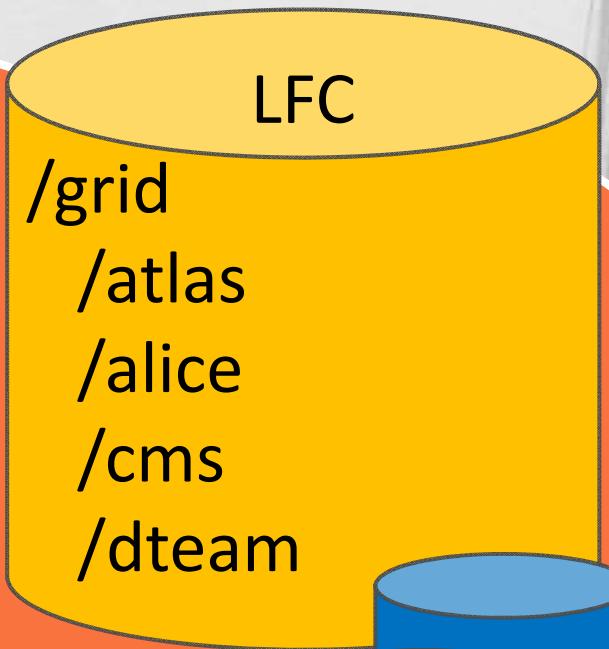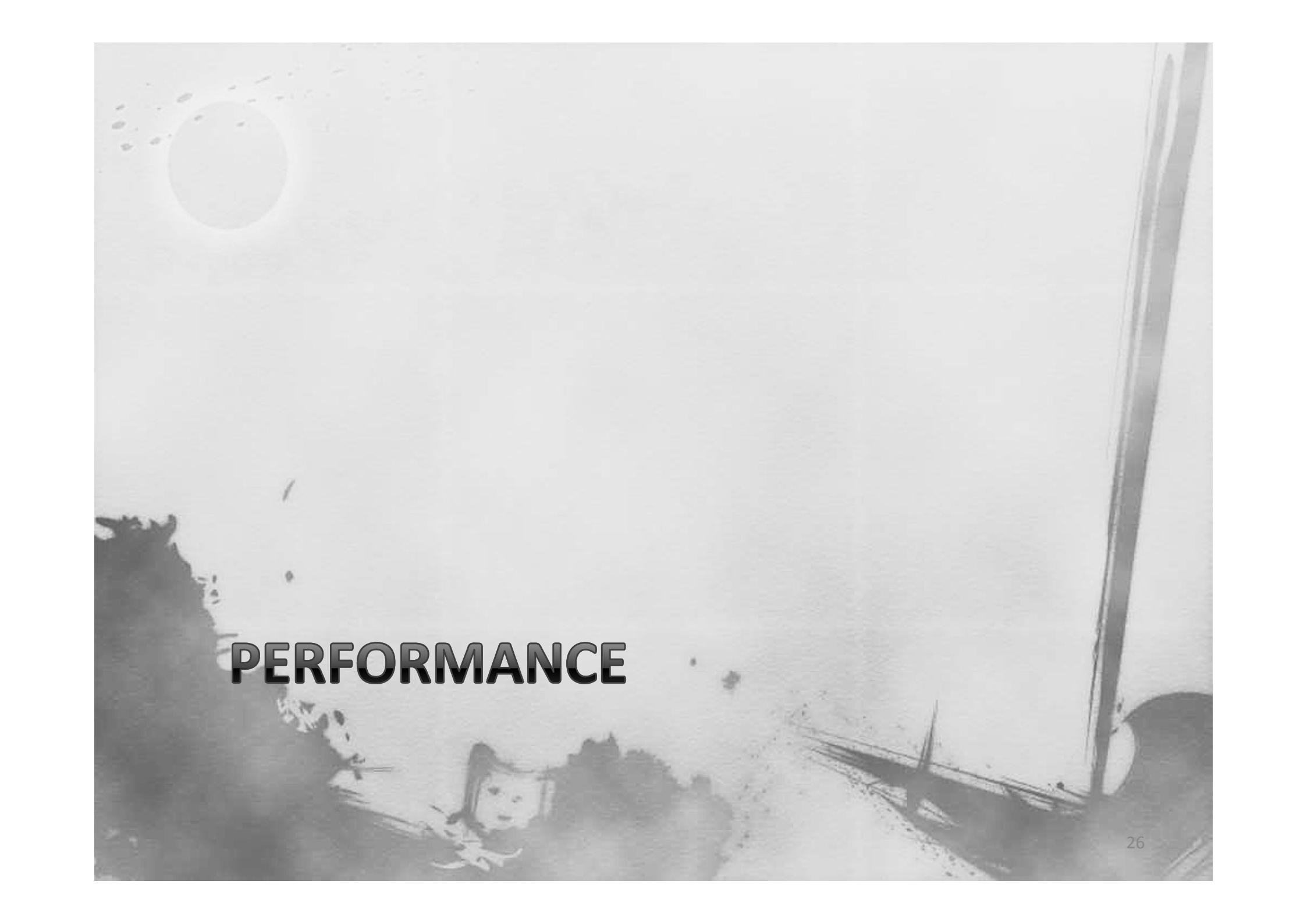
PERFORMANCE

Setup: Machine Specs

- Scientific CERN Linux 4
- Intel Pentium 4 2.8 GHz CPU
- 1 GB memory

SNARL Performance

Dir Size:	*lfc-ls* Cmdline	LFC API	Axis2 Client	Genesis II Client
Small (10 entries)	0.42s	1.49s	1.94s	4.43s
Medium (100 entries)	0.47s	11.73s	15.92s	20.27s
Large (1000 entries)	0.71s	2m 0.37s	2m 1.11s	2m 5.05s

SABLE Performance

File Size:	*lcg-cp* Cmdline	GFAL API	Axis2 Client	Genesis II Client
Small (1 KB)	2.54s	2.60s	2.55s	3.49s
Medium (1 MB)	2.83s	3.61s	2.93s	3.64s

CONCLUSIONS

Future Directions

- Add security context
- Implement MIME, DIME, MTOM
- Explore use as interface between LFC and other tools
- Update for RNS 1.1
- Implement StreamableByteIO

Conclusion

■ Implementation of RNS and ByteIO specification for existing grid system is possible

■ RNS and ByteIO provide an interface that enables the sharing of grid data

Thank You!

- CERN & DM Team
- UVA Genesis II Team
- OpenLab Program at CERN

1. "**Open Grid Forum**," *http://www.ogf.org*.

2. M. Pereira, O. Tatebe, L. Luan, and T. Anderson, "**Resource Namespace Service Specification**," http://www.ggf.org/documents/GFD.101.pdf*, 2006.

3. M. Morgan, "**ByteIO Specification**," http://www.ggf.org/documents/GFD.87.pdf, 2006.

4. M. M. Morgan and A. S. Grimshaw, "**Genesis II – Standards Based Grid Computing**," *Seventh IEEE International Symposium on Cluster Computing and the Grid,* 2007.

5. "**Apache Axis2/C**," *http://ws.apache.org/axis2c*

QUESTIONS?