International Lattice DataGrid (ILDG) / Latfor DataGrid (LDG): Datagrids for Lattice QCD

D. Pleiter

Outline

- ➔ Introduction
- ➔ ILDG Metadata
- ➔ ILDG Middleware
- ➔ Latfor Datagrid
- ➔ Status and Outlook

Introduction

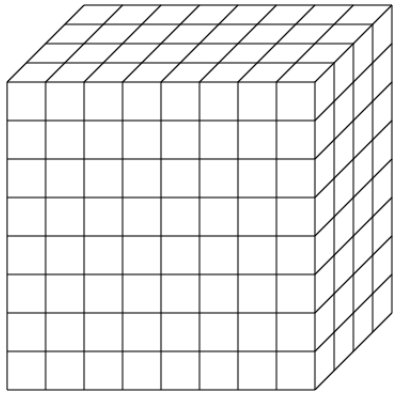
Lattice QCD

Path integral formulation:

$$\langle 0|T\{\phi(x_1)...\phi(x_n)\}|0\rangle = \frac{1}{\mathcal{Z}}\int(\mathcal{D}\phi)\;\phi(x_1)...\phi(x_n)e^{-\mathcal{S}_{\mathsf{E}}[\phi]}$$

where

$$\int(\mathcal{D}\phi) = \prod_x \int d\phi(x)$$



$$\begin{aligned} x_\mu &\rightarrow x_\mu = an_\mu \\ \partial_\mu\phi &\rightarrow \frac{1}{a}\left[\phi(x+a\hat{\mu})-\phi(x)\right] \\ \int d^4x &\rightarrow a^4\sum_x \end{aligned}$$

☞ **Different discretisations (i.e. actions)**

☞ **Evaluation of integral by Markov chain**

Lattice QCD (2)

Simulations should be done using

- ❒ light quarks ($m_{\rm ps} \to 0$)
- ❒ large boxes ($L \to \infty$)
- ❒ small lattice spacing ($a \to 0$)

Costs for generating gauge configurations with dynamical fermions:

$$\text{costs} \quad \propto \quad (r_0 m_{\rm ps})^{-z_{\rm ps}} \left(\frac{L}{a}\right)^{z_L} \left(\frac{r_0}{a}\right)^{z_a}$$

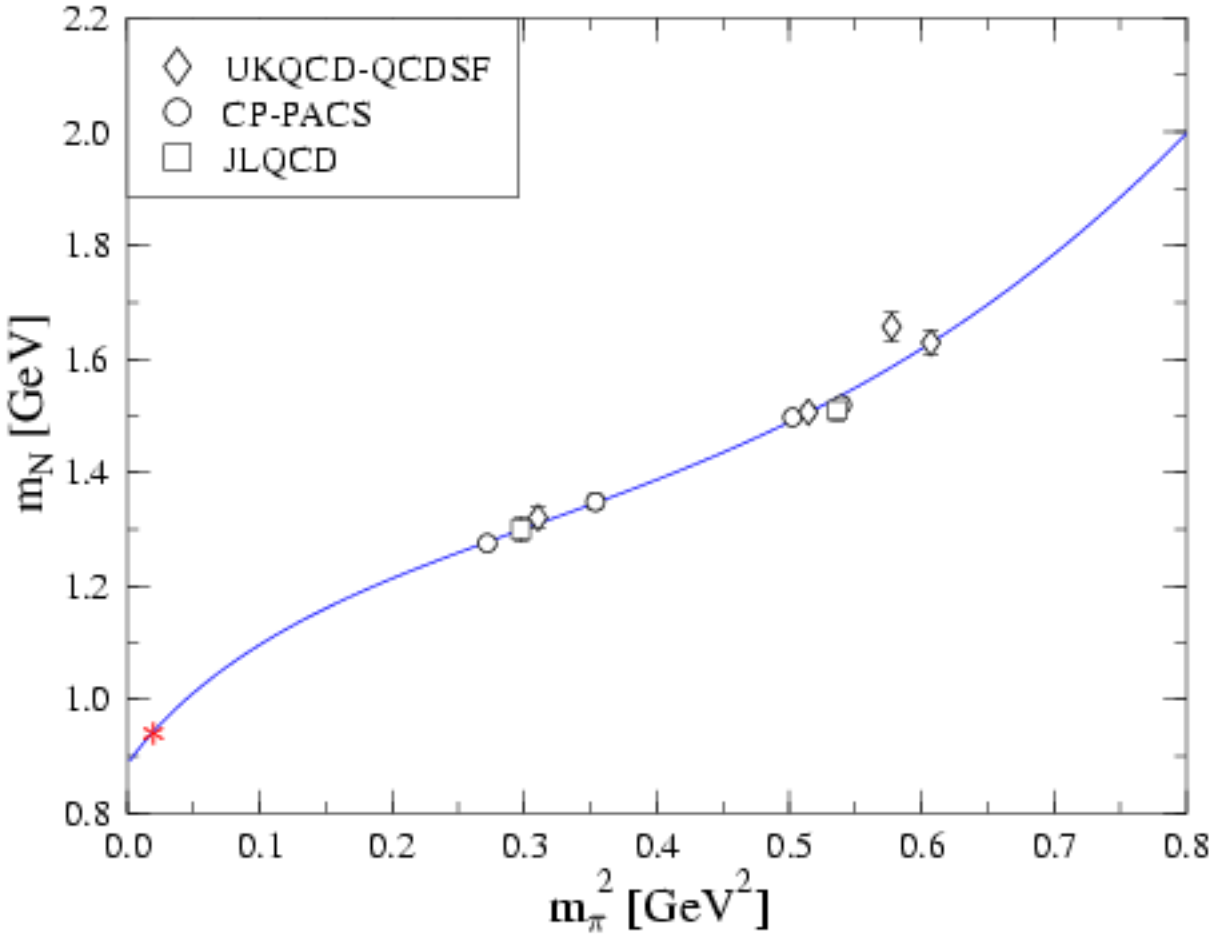
where $z_{\rm PS} \simeq 4 - 6$, $z_L \simeq 5$, $z_a \simeq 2$.

☞ **Increasing computational costs!**

State-of-the-art simulations require TFlops-Computers.

Example: Nucleon Mass

[QCDSF, 2004]



ILDG Initiative

The International Lattice DataGrid was proposed 2001.

Aim: **Longterm storage** and **global sharing** of gauge configurations within a datagrid

☞ **Make more efficient use of expensive data**

☞ Our problem is not handling large amounts of data

Participants: France, Germany, Italy, Japan, UK, USA

Working groups:
- ❒ Metadata working group
- ❒ Middleware working group

`http://www.lqcd.org/ildg`

Requirements

Sharing gauge configurations requires

- Semantic data access to worldwide distributed data
- Standardised description of the configuration (=metadata)
  ➔ XML documents which conform to a XML schema
- Standards on binary file format
- Definition of common middleware interfaces

Metadata

Metadata Items

1. **Physics**
   Description of action and simulation parameter

2. **Algorithm**
   Information about the used algorithm and algorithmic parameters

3. **Source code**
   actually used code, compile time parameters, compilation software

4. **Machine**
   Machine used to produce the configuration

5. **Data management**
   Information about data handling and checksums

Requirements

Description of the data should be:

1. **unique**
   Avoid situation that action is described in two different ways,
   e.g. Iwasaki gauge action

   $$S = \beta \ (c_0 \, \mathrm{plaquette} + c_1 \, \mathrm{rectangular})$$

   or

   $$S = \tilde{c}_0 \, \mathrm{plaquette} + \tilde{c}_1 \, \mathrm{rectangular}$$

2. **extensible**
   Schema should allow to describe future actions ☞ data procurement

3. **simple**
   At least as simple as possible ...

4. **general**
   Allow to describe data other than gauge configurations
   (e.g. propagators, correlators, ...)

Action

Needed for describing action:

- **Array of couplings** - Name and value
  - Information on N_{f}
- **Description of fields** - Normalisation
  - Boundary conditions
- **Further information** - E.g. choice of couplings
  - ➔ **Glossary**

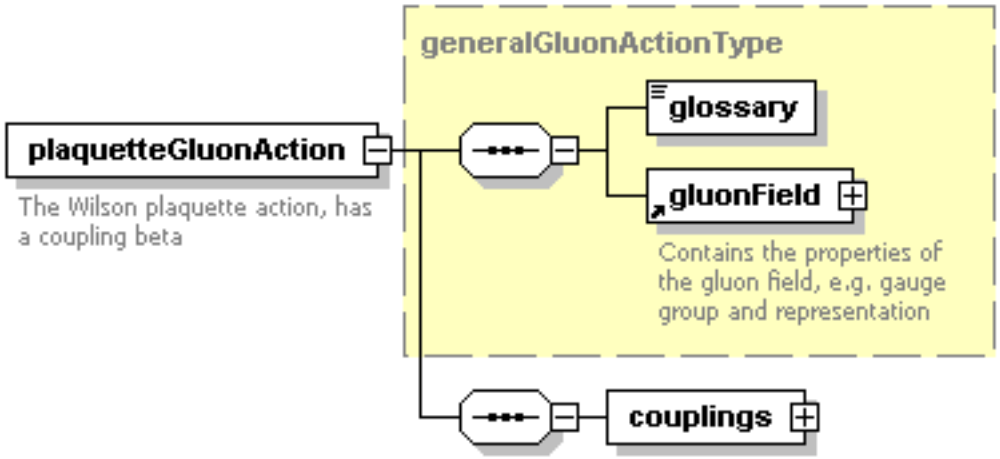
glossary $=$ URL of a (human readable) documentation provided by contributors

Example: Wilson-Plaquette Action

$$S_{\mathrm{G}} = \beta \sum_{\mathrm{plaquette}} \frac{1}{3} \mathrm{Re}\ \mathrm{Tr}\,(1 - U_{\mathrm{plaquette}})$$

- Gluon fields U
- Gauge coupling β

Example: Wilson-Plaquette Action (2)

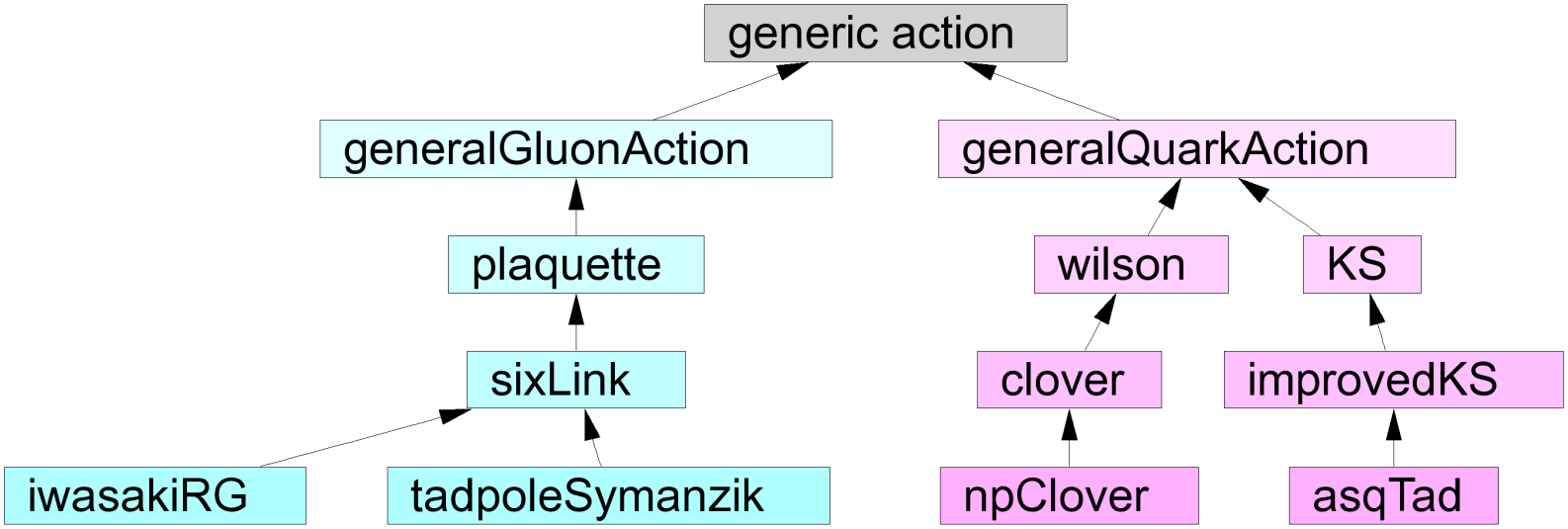



```
<plaquetteGluonAction>
    ...
    <couplings>
        <elem>
            <beta>5.2</beta>
        </elem>
    </couplings>
</plaquetteGluonAction>
```

Action (2)

Description of action parameters most critical to ensure metadata being unique and extensible.

Strategy: hierarchy of actions



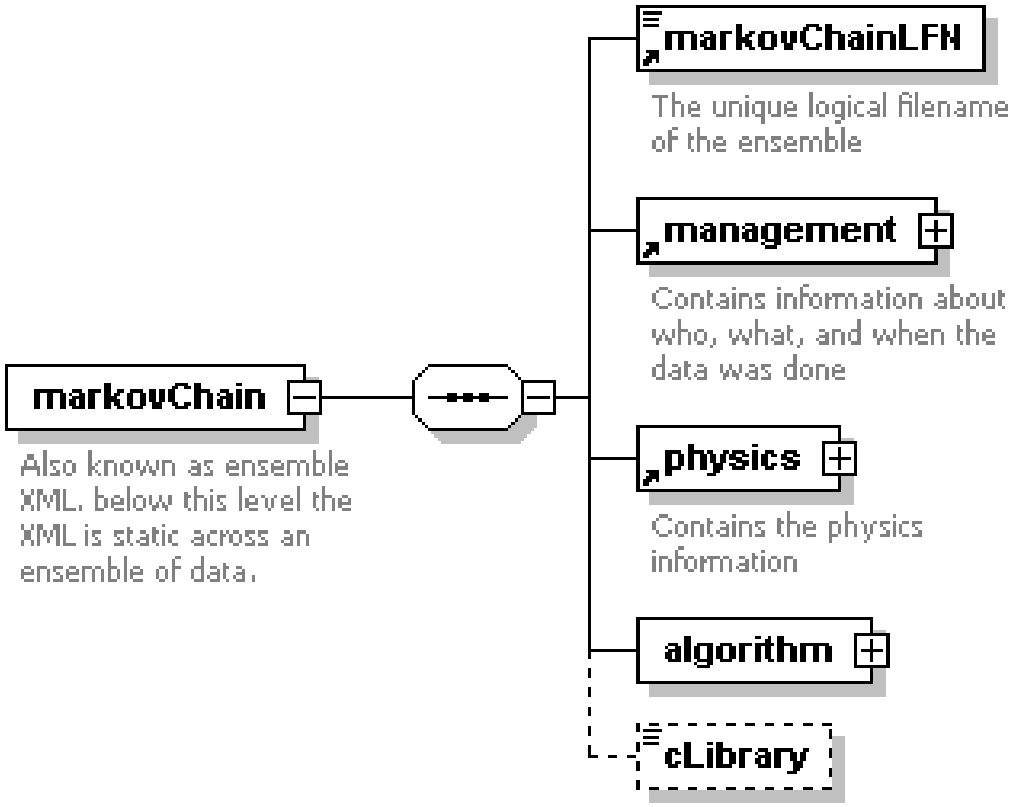
Ensembles and Configurations

Aim: Avoid replication of complicated data structure

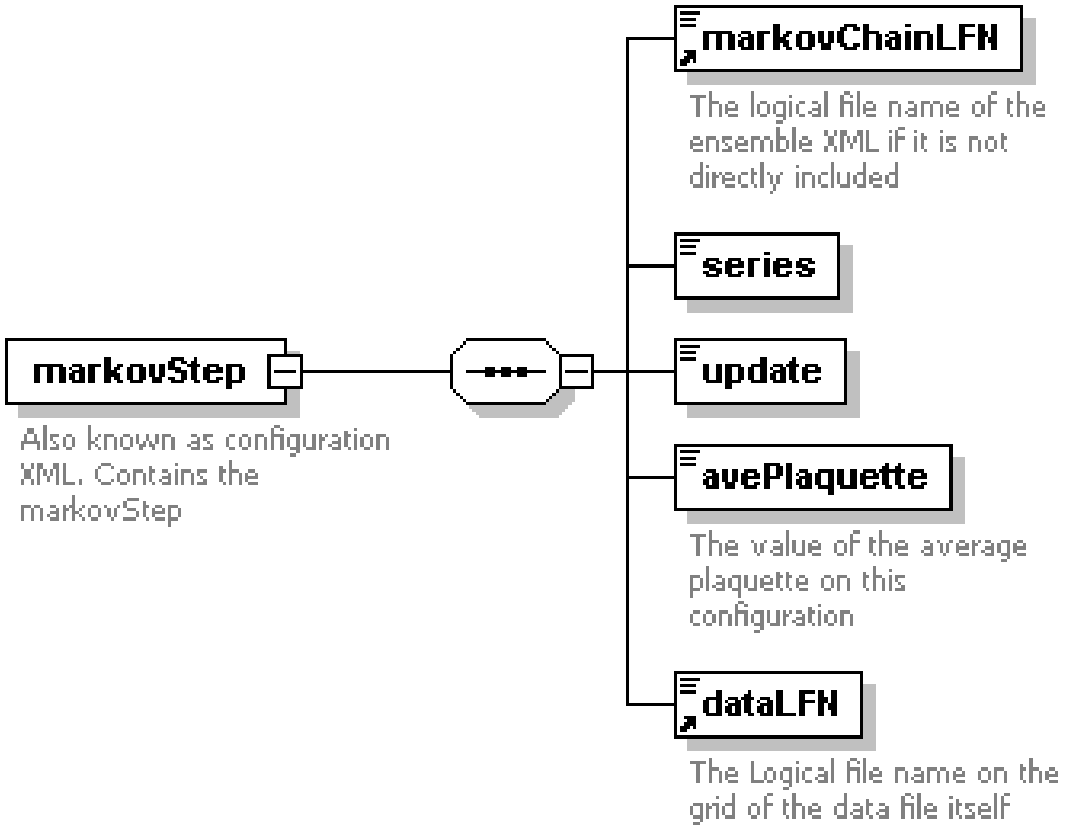
Markov chains are used to generate gauge field configurations

Markov chain	$\leftrightarrow$	Ensemble XML
Markov step	$\leftrightarrow$	Configuration XML

Ensemble XML

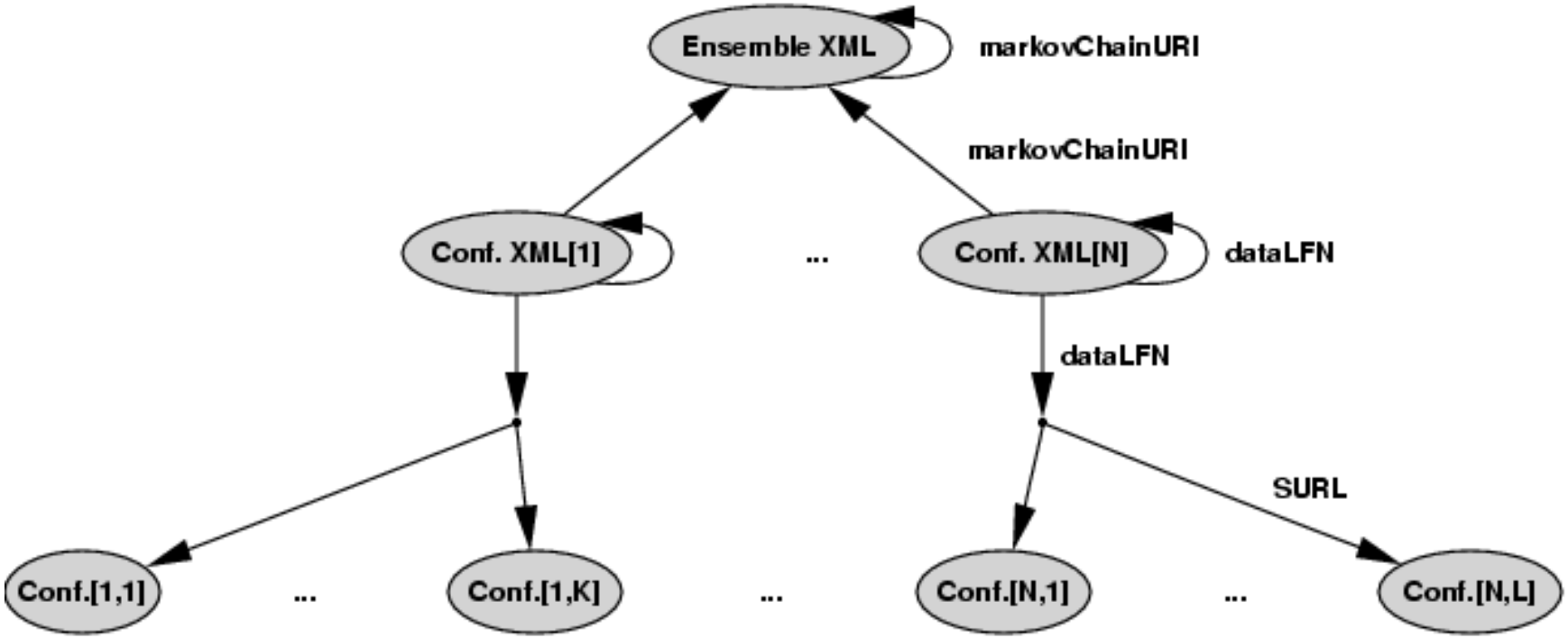


Configuration XML



Linking Metadata and Data

Objects	Links
Ensemble XML document	`markovChainURI`
Configuration XML document	`dataLFN`
Binary data file	



Binary Data File Format

ILDG binary files consists of (at least):

- XML document with parameters useful for reading (`ildg-format`)

```
<?xml version="1.0" encoding="UTF-8"?>
<ildgFormat>
  <version> 1.0 </version>
  <field> su3gauge </field>
  <precision> 32 </precision>
  <lx> 20 </lx> <ly> 20 </ly> <lz> 20 </lz> <lt> 64 </lt>
</ildgFormat>
```

- Binary data (`ildg-binary-data`)
- LFN (`ildg-data-LFN`)

☞ **Use LIME for packaging**

LIME = Lattice QCD Interchange Message Encapsulation [SciDAC]

LIME Records and Messages

- LIME allows to encapsulate ≥ 1 **messages**, each consisting of ≥ 1 **records**.
- Only 3 messages/records mandatory for ILDG:

message	record	LIME record type
#1	...	...
...	...	...
#n	...	...
	#i	ildg-format
	...	...
	#j	ildg-binary-data
	...	...
...	...	...
#m	#1	ildg-data-LFN
...	...	...

- Collaborations free to add other messages/records

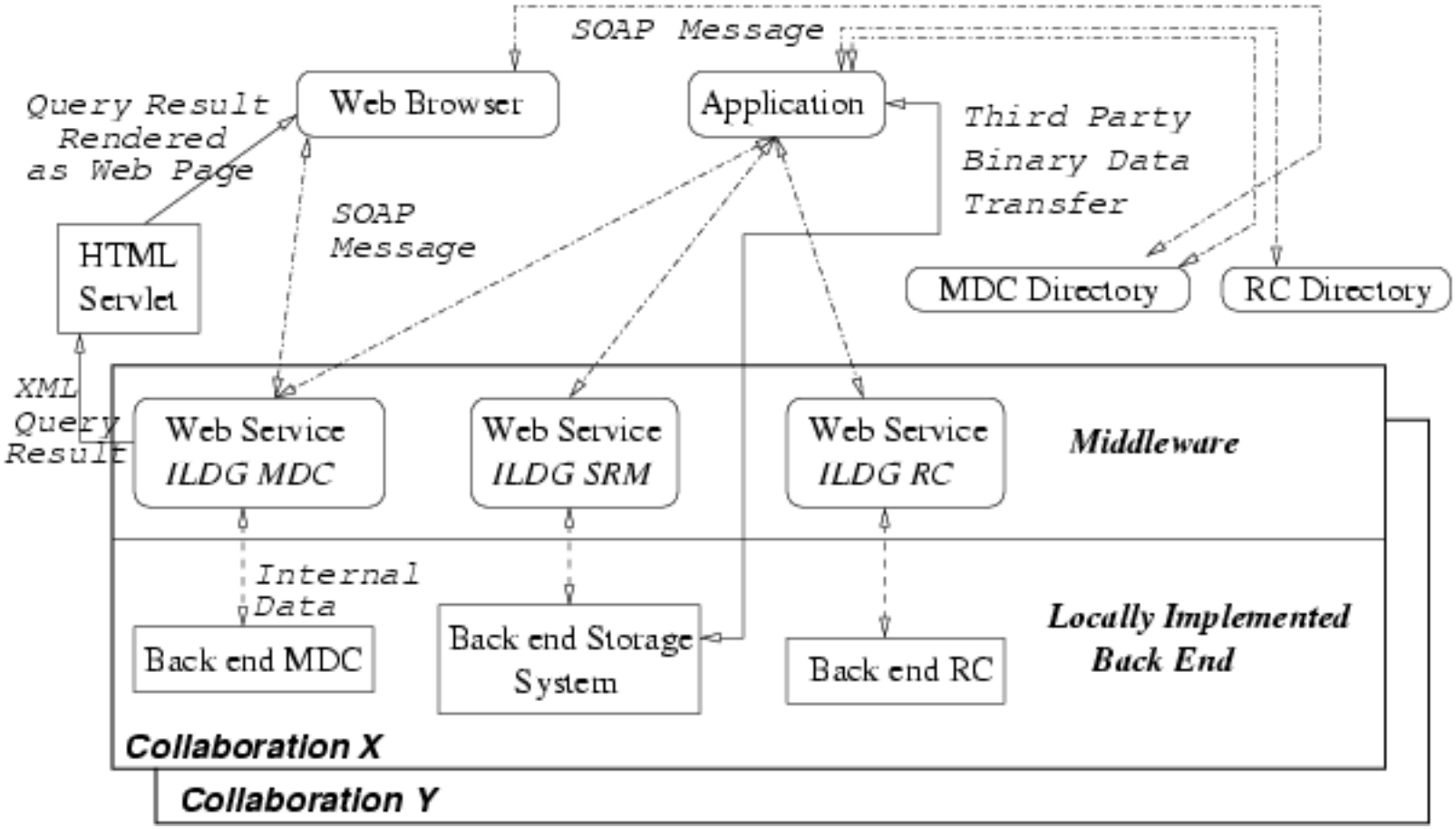
Middleware

Overview

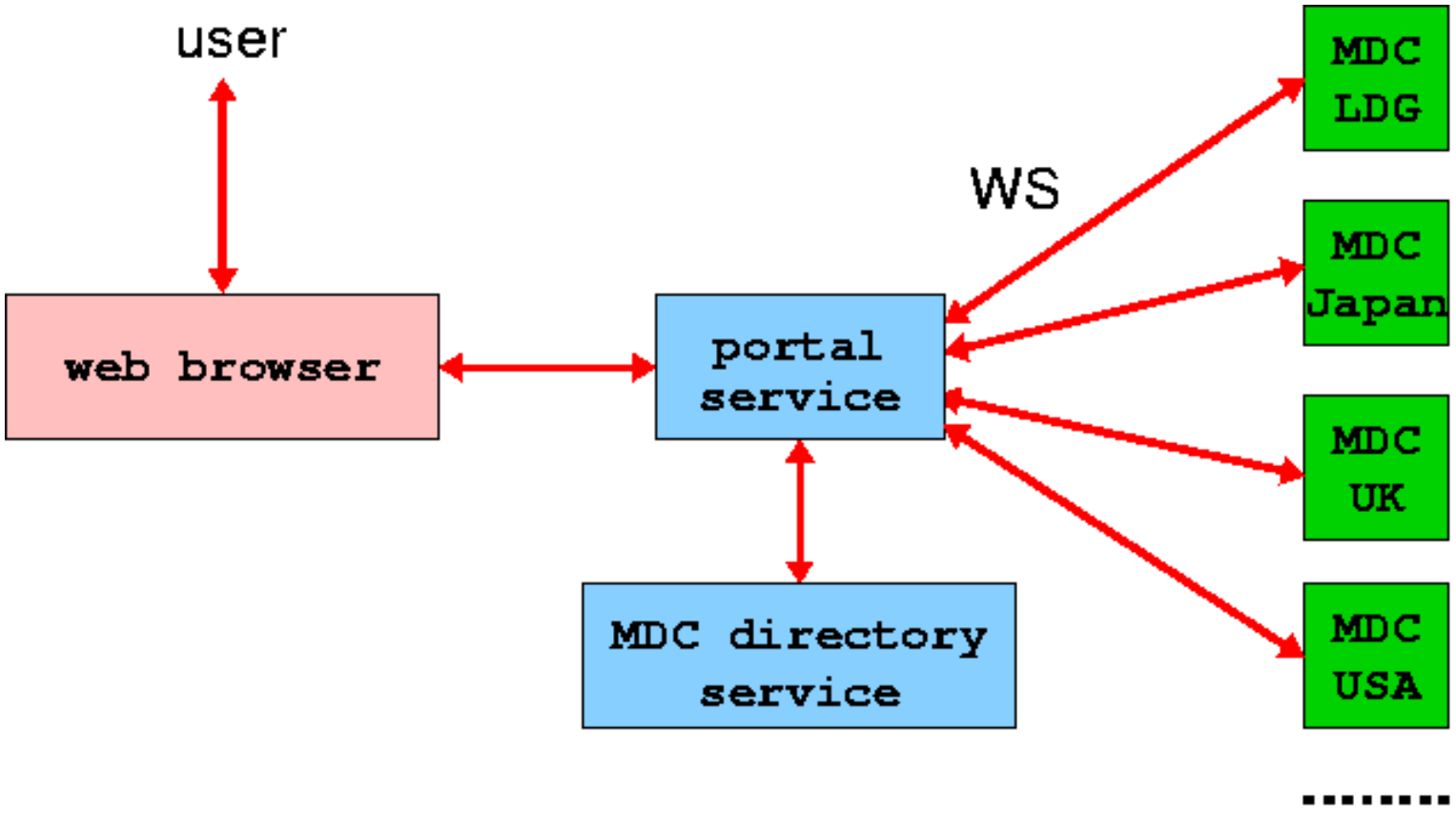
Approach by ILDG-Middleware Working Group:

- Grid-of-grids
- Minimal set of interface definitions:
  - MDC-WS
  - RC-WS
- Few directory services

Middleware Architecture



Example: MDC Portal



MDC/RC-WS Interface

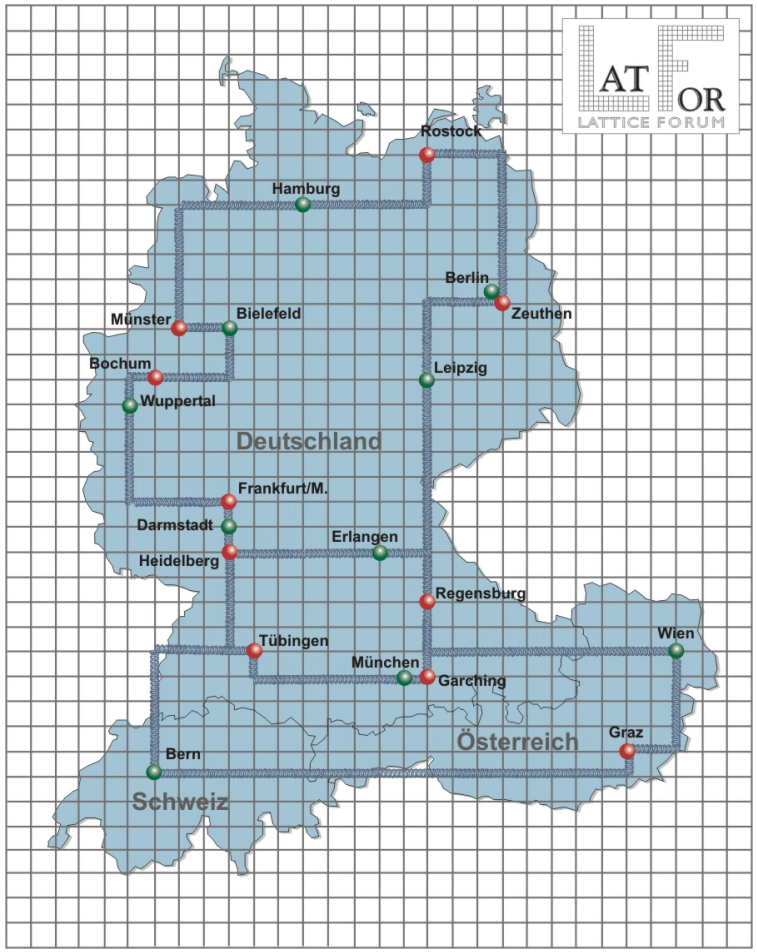
- MDC-WS
  - `doMetadataQuery()`
  - `doEnsembleQuery()`
  - `doConfigurationQuery()`
  - `getSupportedQueryTypes()`
- RC-WS
  - `getURL(LFN)`
  - `addURL(LFN,SURL)`

LatFor DataGrid

Credits

DESY Hamburg:	Michael Ernst, Andreas Gellrich
NIC/DESY Zeuthen:	Karl Jansen, David Melkumyan, Dirk Pleiter, Peter Wegner
NIC/ZAM Jülich:	Otto Büchner, Wolfgang Gürich, Boris Orth, Thomas Lippert,
ZIB Berlin:	Hinnerk Stüben, Stefan Wollny

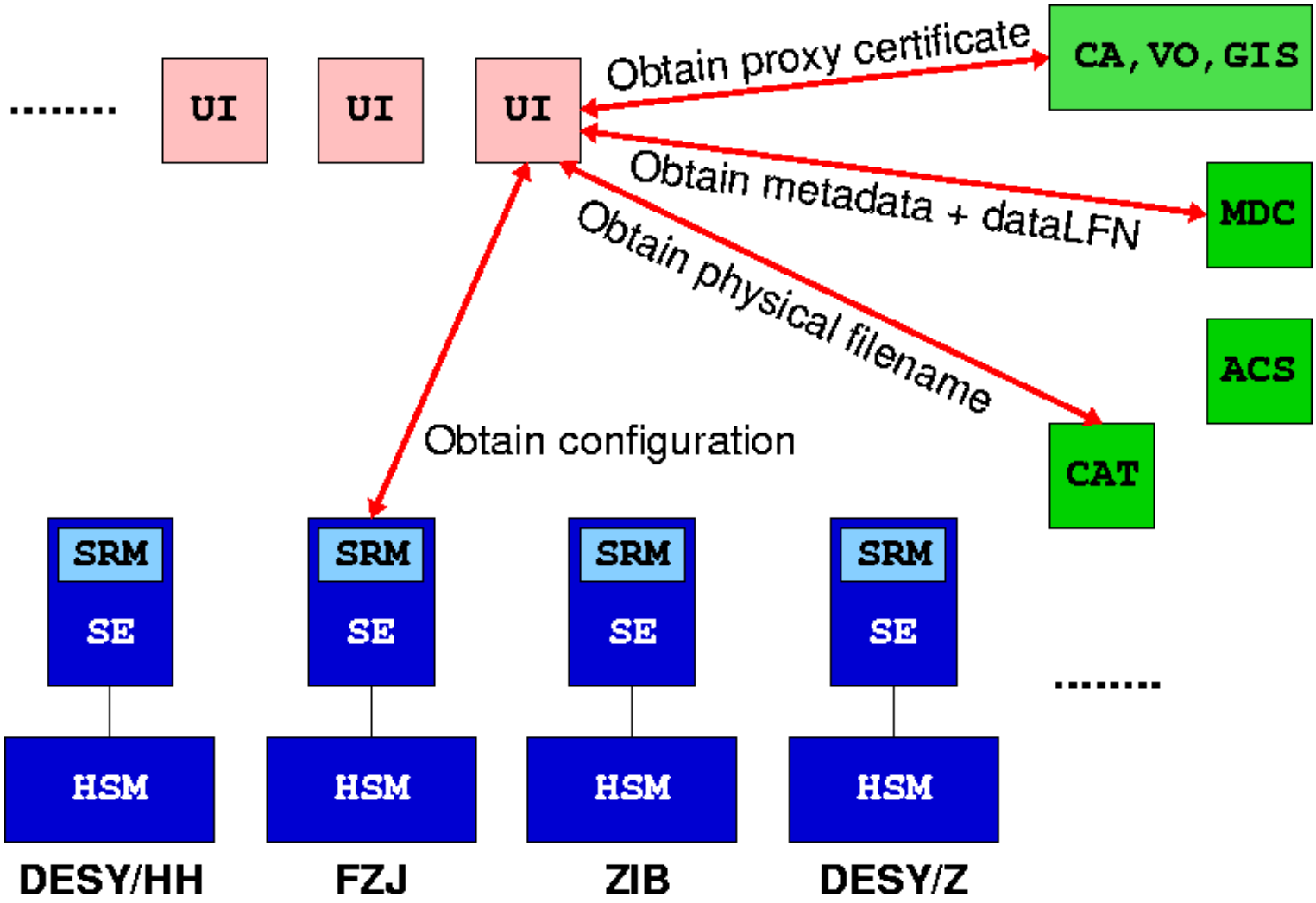
Lattice Forum



➔ Founded in 2001

➔ >50 member physicists

LDG Components



Middleware

[M. Ernst]

- **LCG-2** software is used, e.g.

  - File catalogue
  - Data management client tools
  - Problem: Installation on not(RH|SL) platforms

- Using **dCache** for SE with SRM interfaces

  - Runs on various platforms and supports different HSM systems

- **VO 'ildg'** and **Grid Information System** maintained at DESY/HH

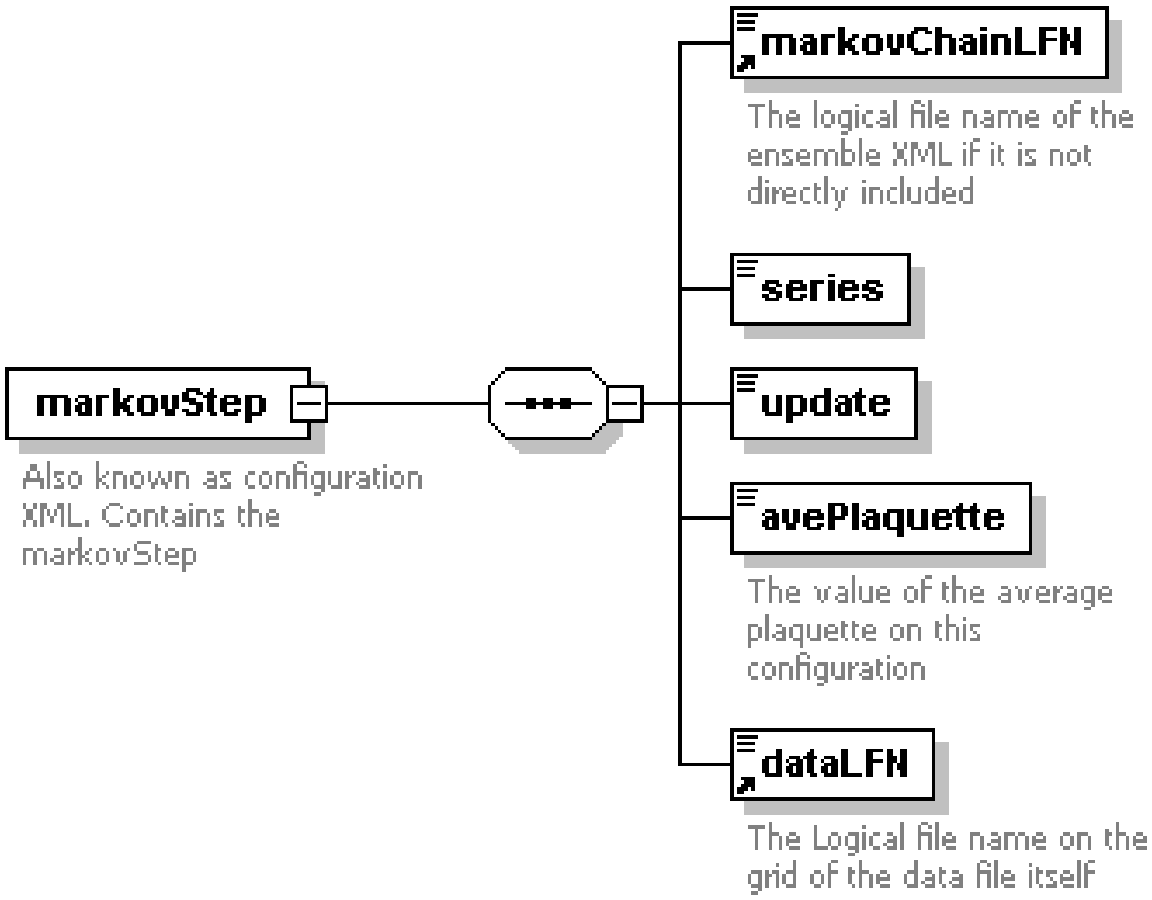
Metadata Catalogue (MDC)

Requirements:
- MDC must be extensible
- Want to use relational database
- Usable beyond ILDG

Solution: [D. Melkumyan]

Advantages:
- Solution as extensible as schemata
- Variety of relational databases supported (we use `mySQL`)
- Usable beyond ILDG (we plan extension to astroparticle physics)
- Easy to install on different platforms

Example from Configuration XML Document



Original XML

```
<markovStep>
  <markovChainURI>
    www.lqcd.org/ildg/qcdsf/b5p40kp13610-24x48
  </markovChainURI>

  <series>561</series>
  <update>1500</update>

  <avePlaquette>0.5612510601</avePlaquette>

  <dataLFN>
    qcdsf_b5p40kp13610-24x48_bqcd.561.1.1.01500.tar
  </dataLFN>
</markovStep>
```

1st Mutation: Java Object

```
public class MarkovStepTypeImpl ...
{
  protected java.lang.String _MarkovChainURI;
  protected boolean has_AvePlaquette;
  protected double _AvePlaquette;
  protected java.lang.String _DataLFN;
  protected java.lang.String _Series;
  protected java.lang.String _UpdateMarkovStep;
  private java.lang.String idInternal;

  ...
}
```

2nd Mutation: Relational Table

MarkovStepType					
Field	Type	Null	Key	Default	Extra
idInternal	varchar(32)		PRI		
ildg_markovChainURI	varchar(255)	YES		NULL	
ildg_avePlaquette	double	YES		NULL	
ildg_update	varchar(255)	YES		NULL	
ildg_dataLFN	varchar(255)	YES		NULL	
ildg_series	varchar(255)	YES		NULL	

☞ currently 59 such tables (not all actions included yet)

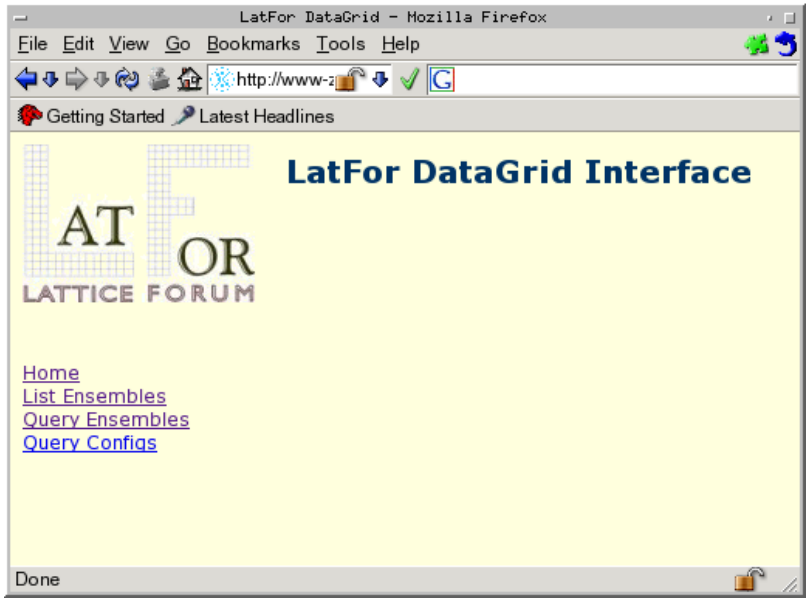
MDC Implementation Issues

- Not all features of XML schema supported
  - ☞ Use modified (but equivalent) ILDG schemata
- Evaluation of general XPath queries slow
  - ☞ Optimize most relevant use case: query to simple elements

Searching for Configurations within LDG

Query Medata Catalogue:

1. Search for ensemble ➔ **markovChainURI**
2. Download **ensemble XML document**
3. Search corresponding configurations ➔ **dataLFN**
4. Download **configuration XML document**



`http://www-zeuthen.desy.de/latfor/ldg/demo`

Access to MDC

- Access to MDC via web-services
- Modifying operations will require authorisation
  ☞ Use GSI for authentication

Operation	Return
`doEnsembleQuery`	List of `markovChainURI`
`doMetadataQuery`	List of `dataLFN`
`doEnsembleGet`	Ensemble XML document
`doMetadataGet`	Configuration XML document
`doEnsembleUpdate`	
`doMetadataUpdate`	
...	

Access Control Service (ACS)

- LDG-MDC will for each ensemble store permissions for
  - modifying metadata
  - inserting/updating/deleting data files (configurations)
  - downloading data files
- Access control information will be forwarded to file catalogue

Status and Outlook

Status International DataGrid (ILDG)

- ✔ XML schemata defined
- ✔ Binary file format defined
- ✔ Middleware basics defined
- ✘ Details middleware interfaces need to be worked-out
- ✘ Open questions concerning interoperability

Status Latfor DataGrid (LDG)

- ✔ Initial prototype **Metadata Catalogue** (MDC) running
- ✔ **WS** defined
- ✔ **User tools** defined, implementation started
- ✔ **Virtual Organisation** (VO) installed

  Associated resources (e.g. SE) published via GIS
- ✔ **File Catalogue** running
- ✔ 4 **Storage Elements** (SE) installed (all with HSM)
- ✘ **Access Control Service** (ACS) to be implemented
- ✔ O(8) ensembles, O(10,000) configurations marked-up, inserted into MDC, uploaded to SE, registered in catalogue

Next Steps and Future Plans

- Push LDG to production level
- Make LDG available to all lattice groups in Germany
- Extent LDG to groups in France and Italy
- Setup compute grid

Questions?