

MMF: A Flexible Framework for Metadata Management in ***UNICORE***

Waqas Noor¹, Bernd Schuller²

¹ RWTH Aachen Technical University

² Jülich Supercomputer Centre (JSC)

UNICORE Summit 2010, May 18th, 2010, Jülich, Germany

Outline

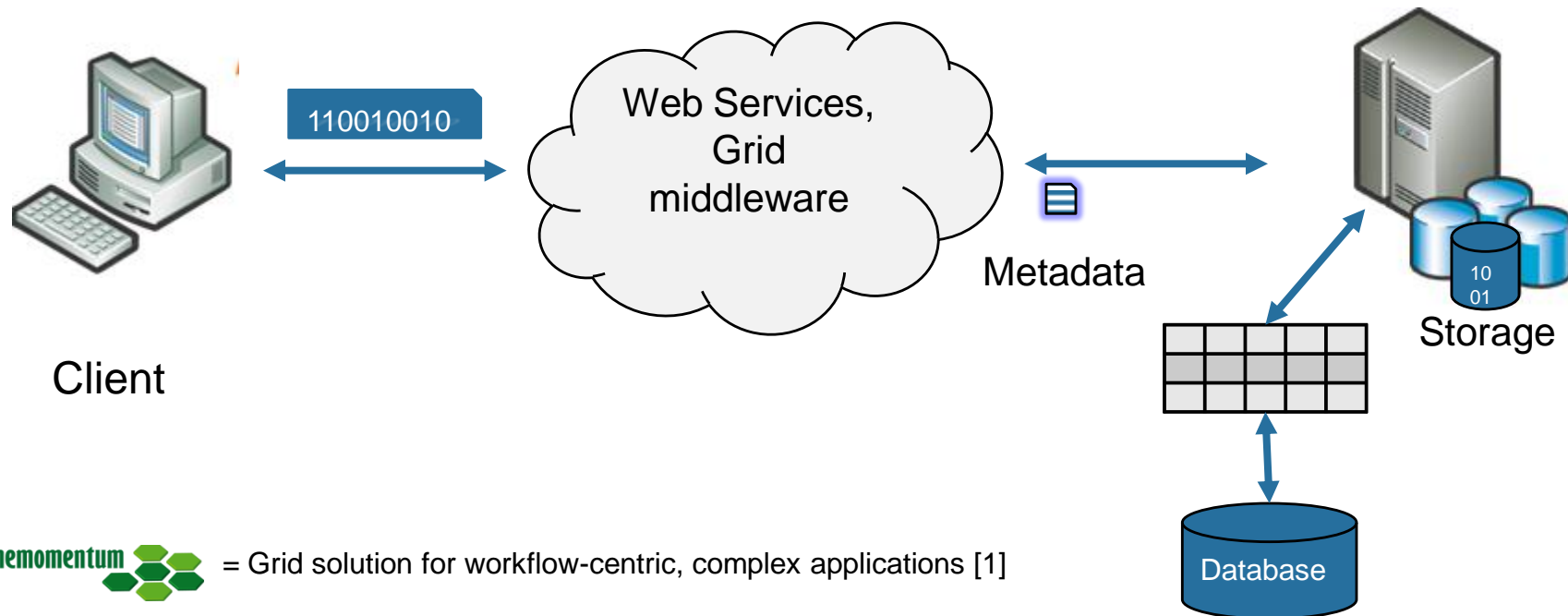
- Motivation
- Objectives
- Metadata Management
 - Architecture
 - Implementation
 - Performance & Evaluation
- Usage Scenarios
- Future Work

Motivation

- Metadata
 - Classification of data
 - Makes data highly searchable
- Metadata & Grid
 - No full text search
 - No flexible support for user defined metadata
 - Data and metadata stored separately
 - Data migration problem

Motivation

- Centralized repositories for metadata



= Grid solution for workflow-centric, complex applications [1]



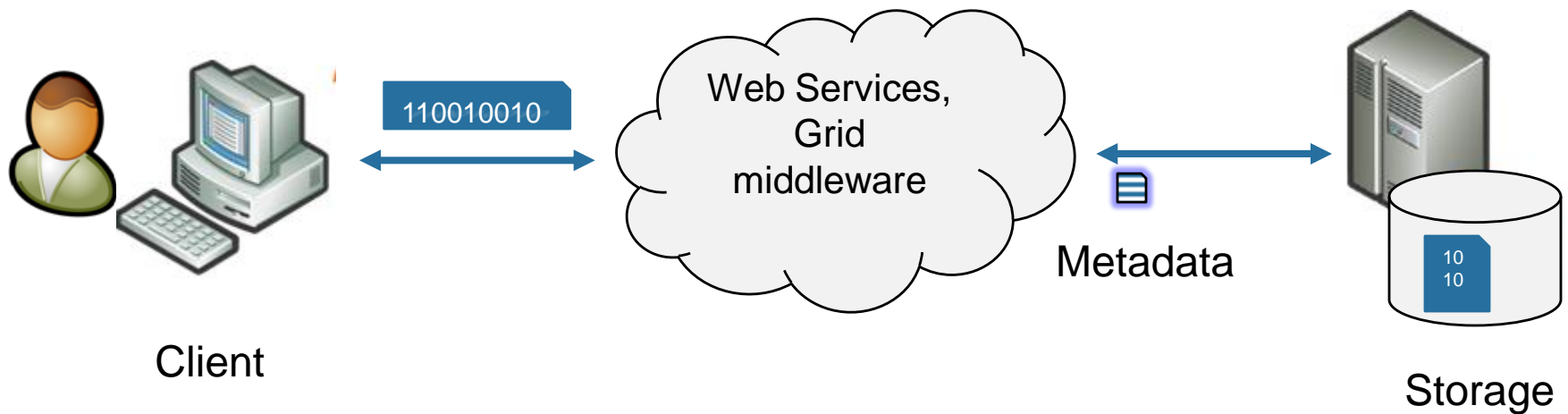
= Metadata catalogues for gLite grid middleware [2]

Approach

- No centralized metadata management
- Schema free metadata model
- User defined metadata as tags
- Full text search
- Extensible & scalable

Metadata Management Framework (MMF)

Metadata Management Framework (MMF)



Architecture

UNICORE

(*Uniform* Interface to *Computing Resources*)

- Open source, ready to run grid middleware
- Provides seamless access to grid resources
- OGSA based architecture
- Workflow can be submitted to grid
- Involved in various grid R&D projects [3]

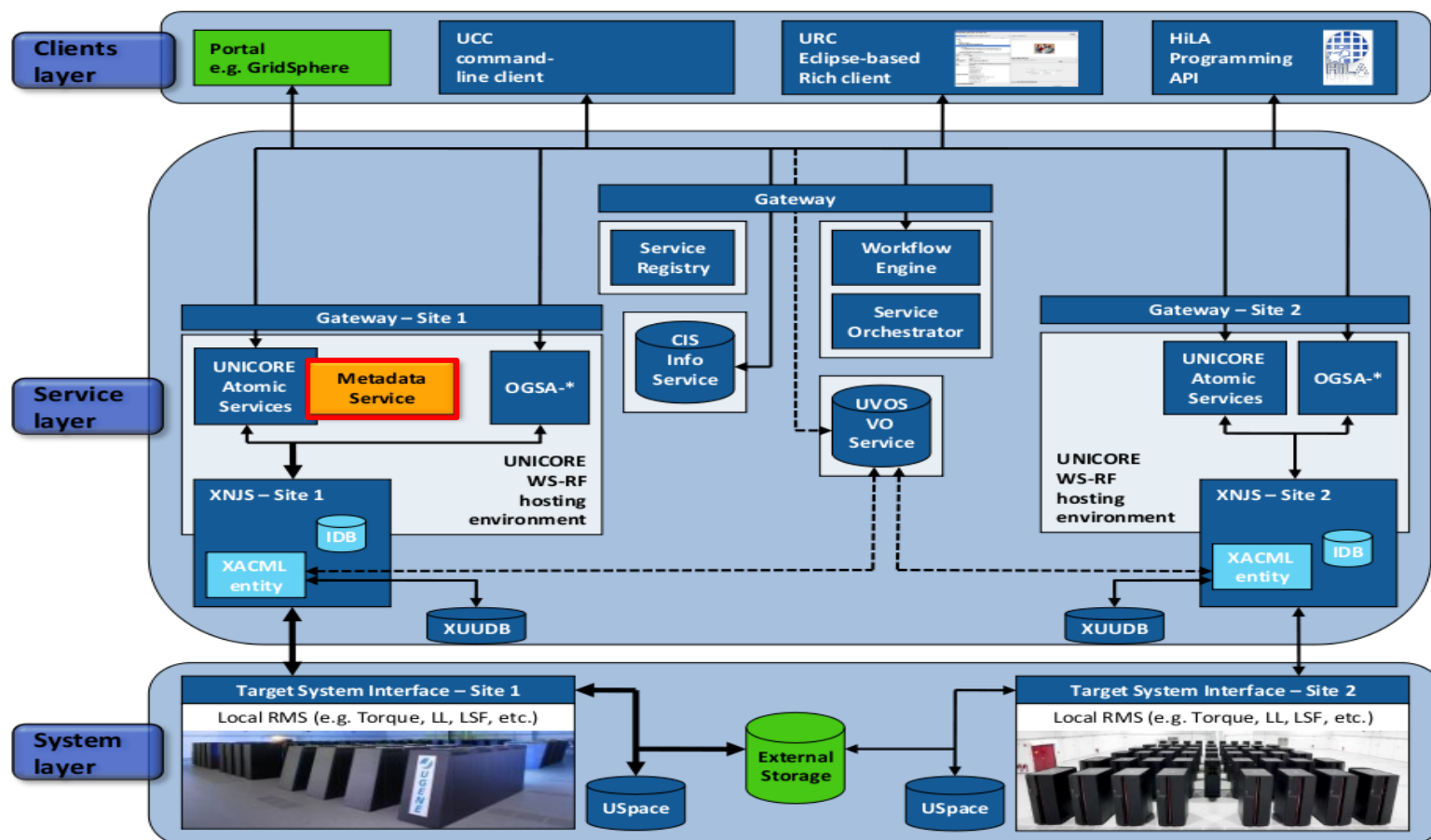
- FIT4Green
- WisNetGrid
- SLA4D-Grid
- DEISA2
- And many more ...



SLA4D²GRID



UNICORE Architecture



Metadata Management Architecture

Metadata Service

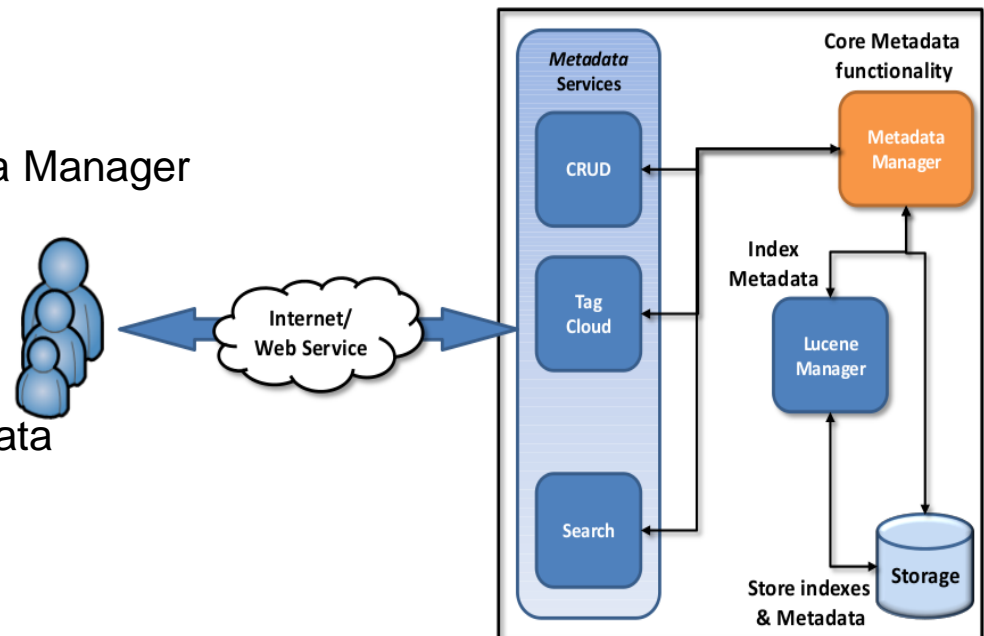
- OGSA based Web Service
- Functionality is backed by Metadata Manager

Metadata Manager

- Provides core functionality of metadata Management
- Communicate with Lucene Manager

Lucene Manager

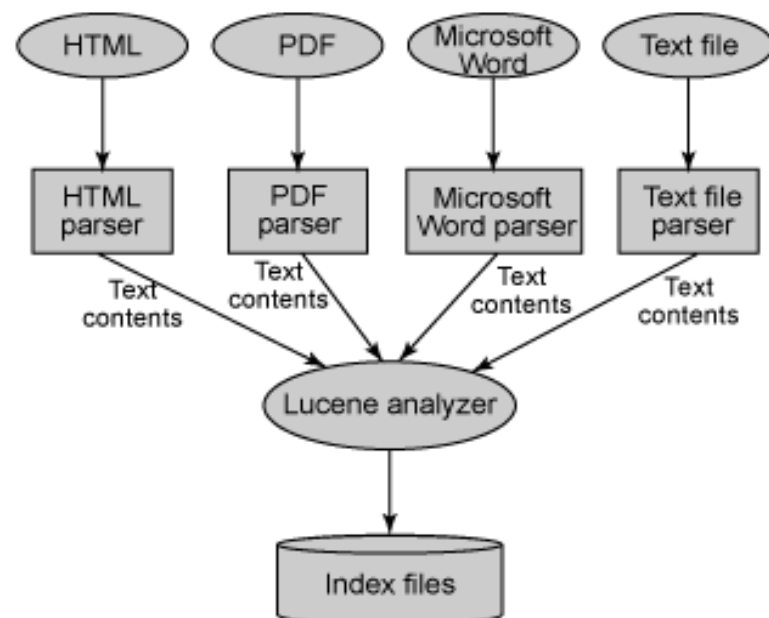
- Responsible for indexing and retrieving the data
- Provides search interfaces



Apache Lucene 2.9.0 is used for indexing data [4].

Apache Lucene (2.9.0)

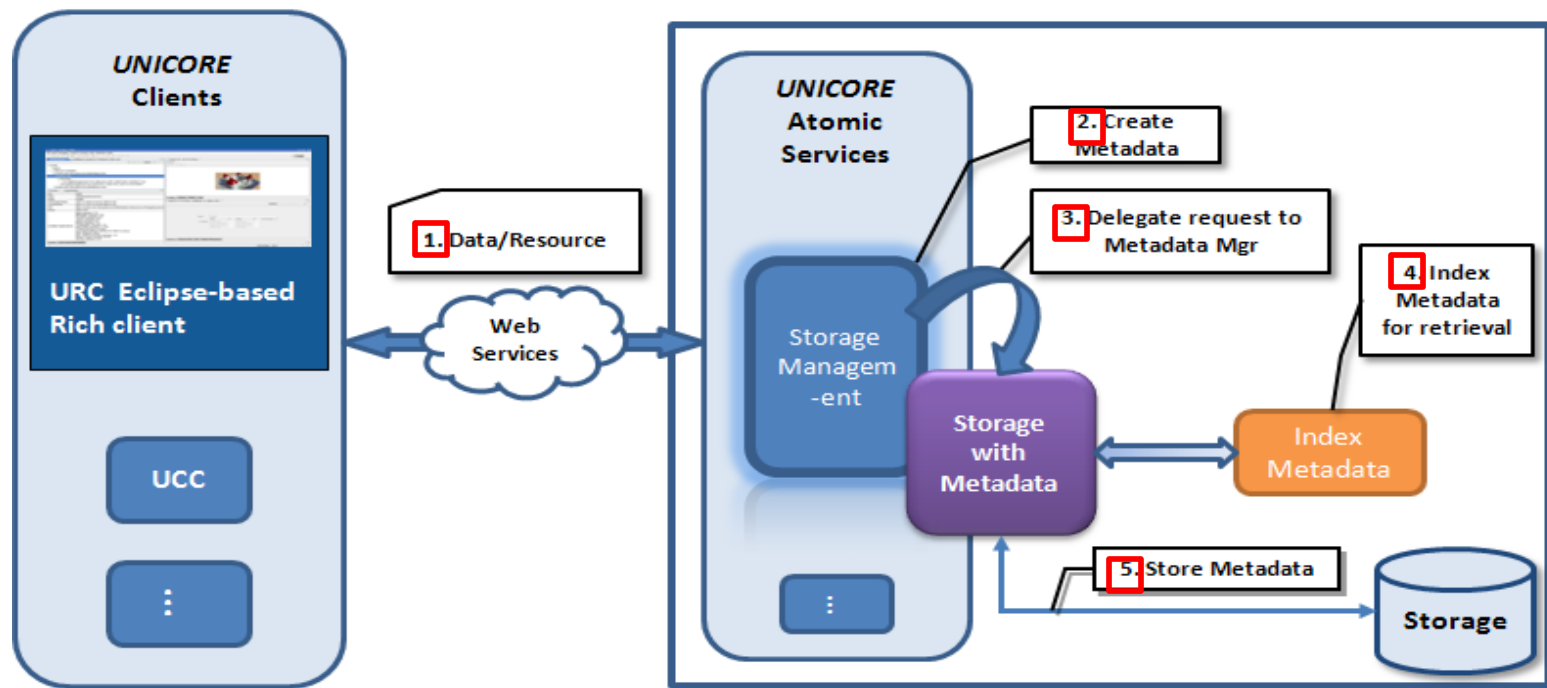
- Open Source, cross-platform full text search engine library
- Indexing, searching and retrieving
- Supports many types of queries
- Scale to millions of documents
- Easy to integrate and use
- Inverted Indexing
 - <docId, term, termFreq, position>



The Lucene logo, featuring the word 'Lucene' in a stylized, green, cursive font with a green outline and a green shadow effect.

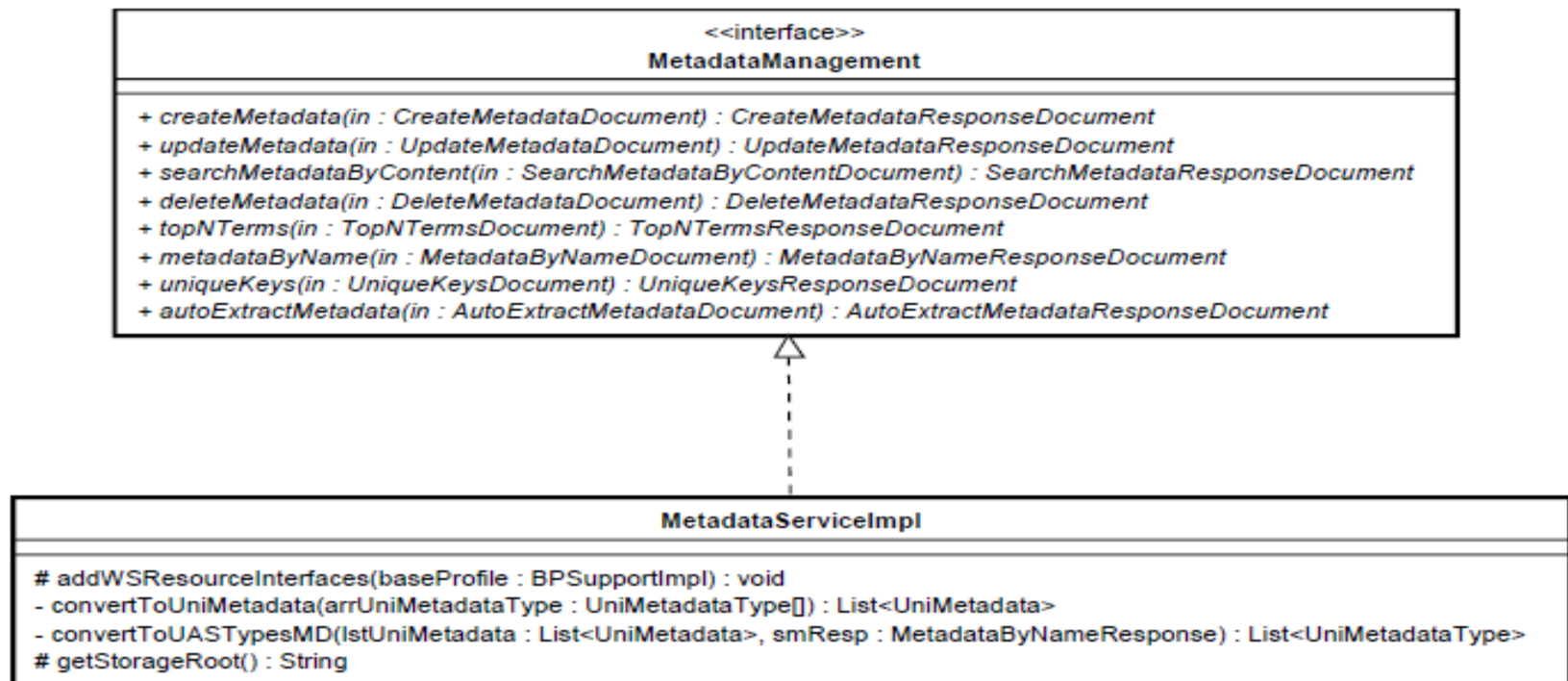
Metadata Management Architecture

Metadata with reference to UNICORE storage service



UCC: UNICORE commandline client

Metadata Management Interface



Implementation

Metadata Service Implementation

- As a *Web Service (WSRF)*
- Apache Lucene for indexing
- Only *English* documents are indexed
- JSON (JavaScript Object Notation) as data structure
- Metadata in UNICORE's Storage Management Service (SMS)
- Code added in UNICORE Sourceforge code repository as "**contribution**" code

Metadata Service Implementation

Sample JSON representation of metadata.

```
{  
  "resourceName": "/../TimeSeriesData",  
  "type": "txt",  
  "createdDate": "Date/Time",  
  "owner": "<username>",  
  "experimentPlace": "FZJ",  
  "key...": "value...",  
  "description": "The multiple dipole construction method using MUSIC",  
  "tags": "Time series, parallel approach, estimation, noise, auditory  
  experiment"  
}
```


Performance & Evaluation

Evaluation – Search Comparison

Search options as comparison with UNICORE' SMS.

Search by	Storage Management Service (SMS)	<i>Extended</i> Storage Management Service
Basic File Metadata (filename, creation date etc)	✓	✓
Boolean Query	Partially (AND, OR)	✓
Wild Characters search on text	Partially (filename only)	✓
Range Query	Partially (date only)	✓
Fuzzy Query	✗	✓
Full Text Search	✗	✓

Evaluation – Testing Environment

The following configuration is used for evaluation and performance.

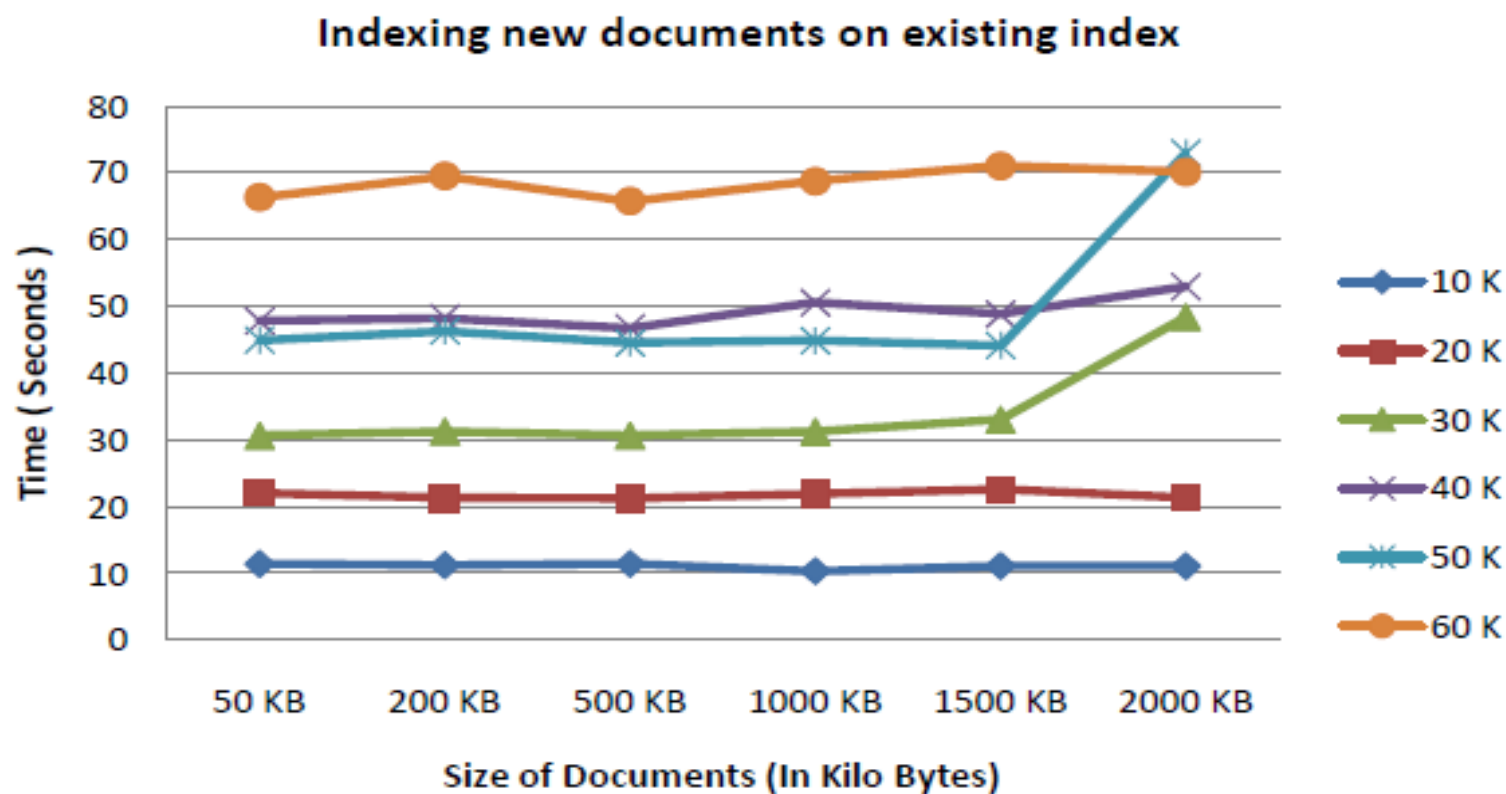
Hardware environment	
CPU:	Intel(R) Core(TM)2 Quad CPU Q9400 @ 2.66GHz
RAM:	4GB
OS:	OpenSUSE 11.2
Software environment	
Java Version:	1.6
Lucene Version:	2.9.0
Analyzer:	StandardAnalyzer
General settings	
Index Storage Place	Local system hard drive:
Documents Language:	English
Dedicated machine for index:	No

Evaluation – Indexing Time

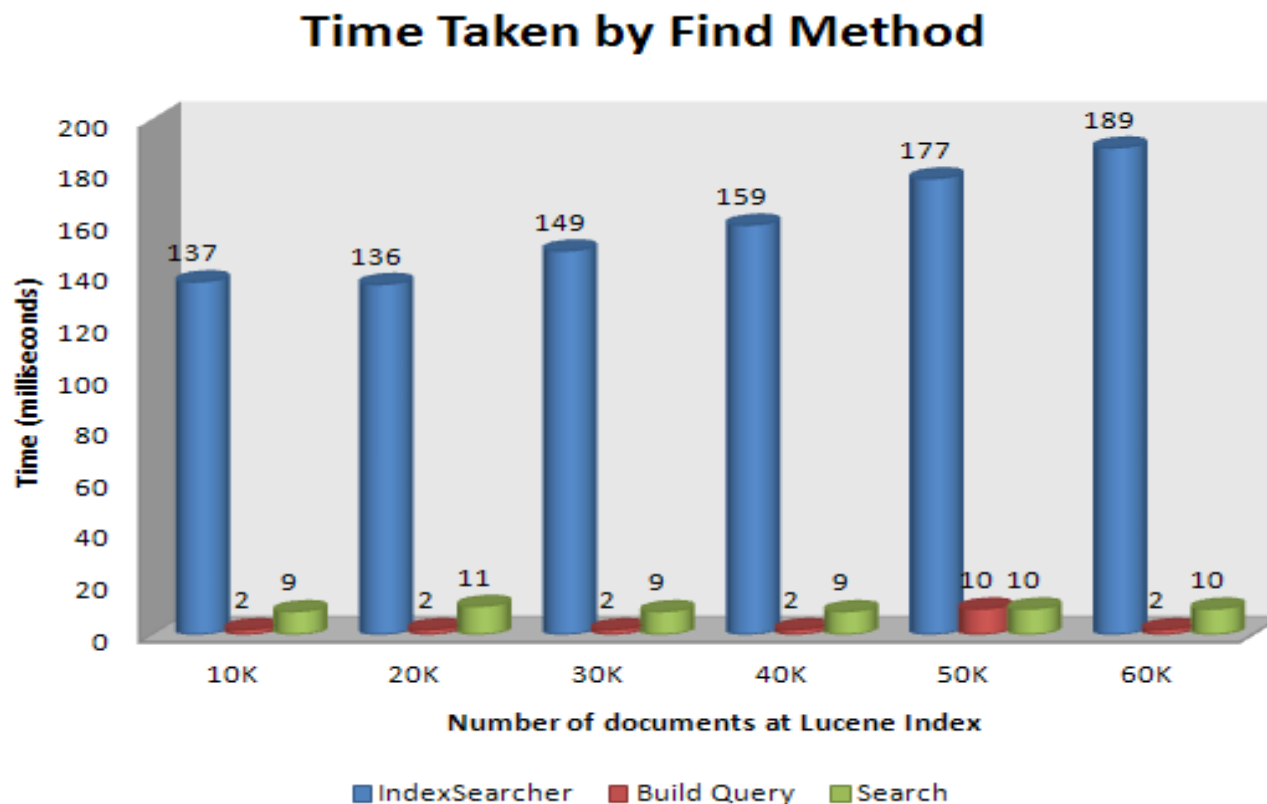
- # of documents varied, size of documents same (~48 KB).

Documents (K)	Time to index (Seconds)
0.1	0.37
1	1.7
10	3.3
50	10
100	17
500	132

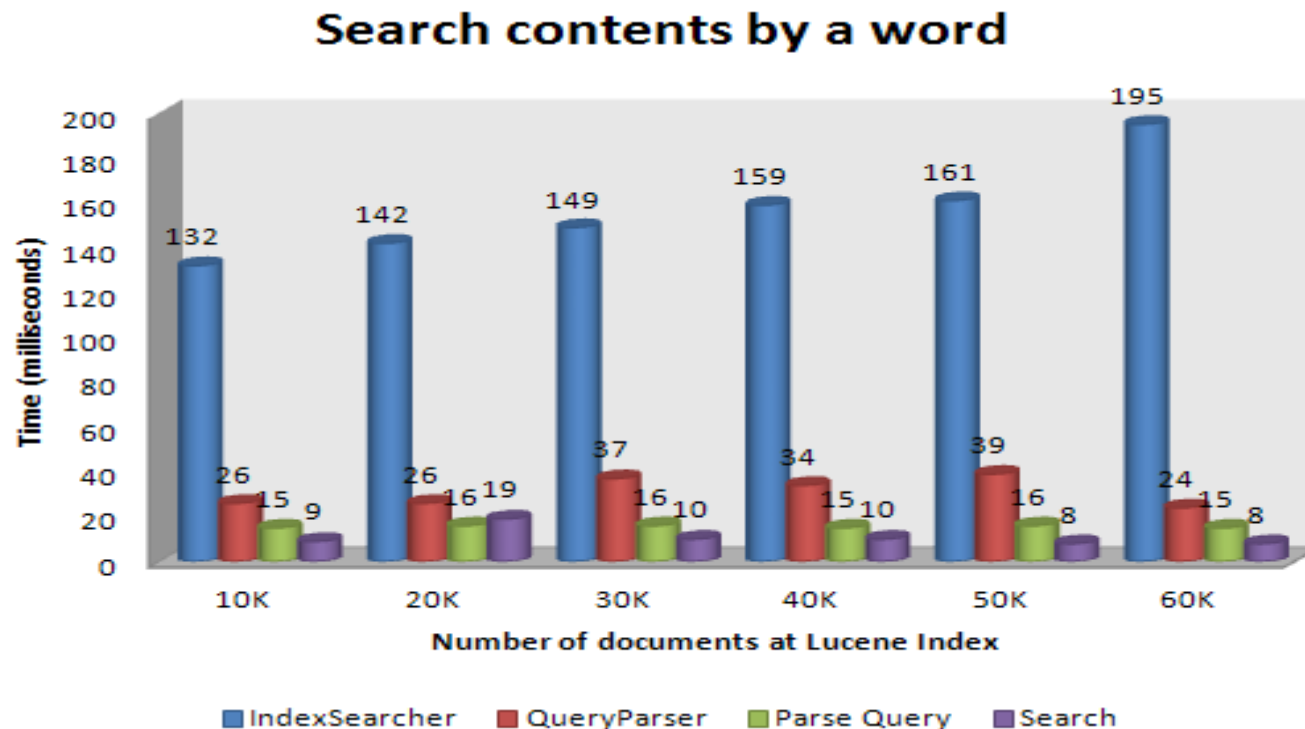
Evaluation – Re-open Index



Evaluation – Search by key/value pair



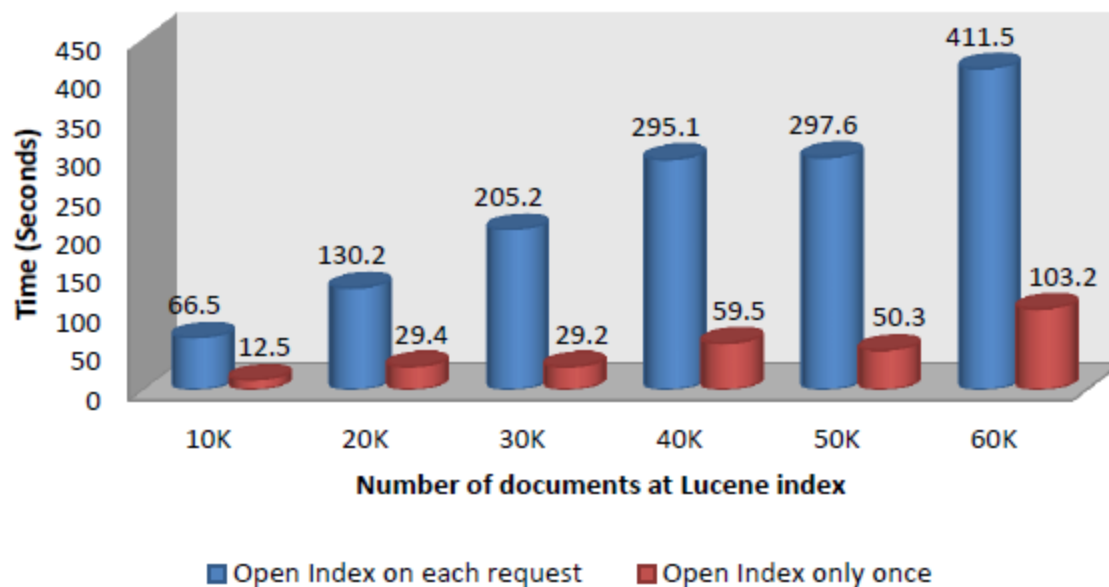
Evaluation – Search by content



```
SELECT documents FROM index where index contains 'word'
```

Evaluation – Performance enhancement

Open index on request/once



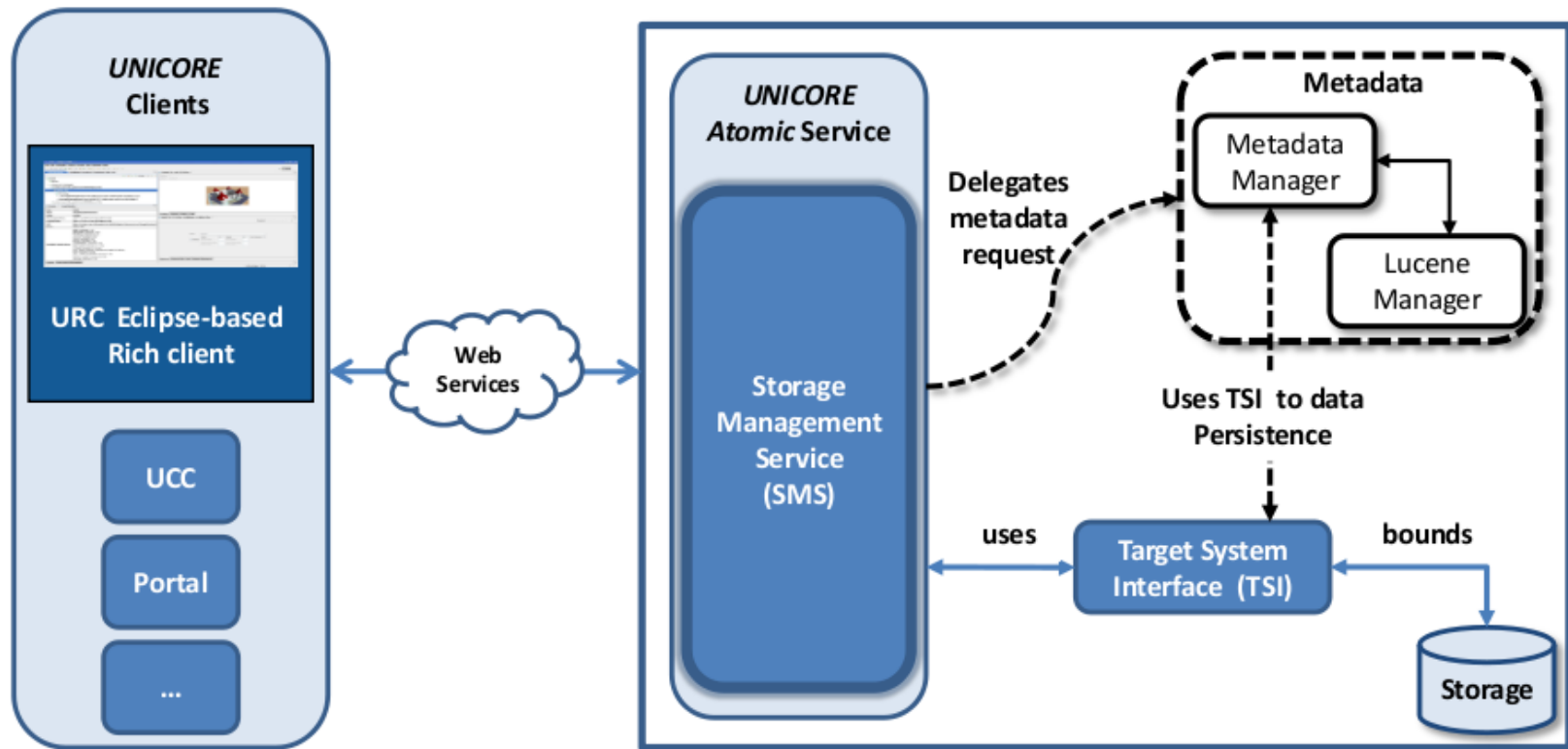
- 50 KB
- 200 KB
- 500 KB
- 1000 KB
- 1500 KB
- 2000 KB

Approx. 4-5 times indexing time is reduced.

Usage Scenarios

Usage Scenario 1 – Within UNICORE

Metadata Management integrated in UNICORE's Storage Service.



UCC: UNICORE commandline client

Usage Scenario 1 – Create Metadata

UNICORE :: Meta-data

Create Metadata Update Metadata Delete Metadata Tags [Cloud] Search [Simple] Search [Advance]

Associate metadata with the resource (path/resource Id).

Attach with: /localdisk/simulation/weather_forecast_report

Key: weatherLocation

Value: Aachen

Tags: Aachen weather unicare

Metadata: weatherLocation:Aachen

Add >>

Finish

Usage Scenario 1 – Update Metadata

UNICORE :: Meta-data

Create Metadata Update Metadata Delete Metadata Tags [Cloud] Search [Simple] Search [Advance]

Update metadata (already associated with the resource).

Associated with:

Key:

Value:

Tags:

Metadata:

Usage Scenario 1 – Delete Metadata

X UNICORE :: Meta-data

Create Metadata

Update Metadata

Delete Metadata

Tags [Cloud]

Search [Simple]

Search [Advance]

Delete partially/complete metadata associated with the resource.

Associated with:

Fetch

Key	Value

Delete Selected

Delete All

Usage Scenario 1– Search (simple)

The screenshot shows a web application window titled "UNICORE :: Meta-data". The interface includes a toolbar with five buttons: "Create Metadata" (notepad icon), "Update Metadata" (pencil icon), "Delete Metadata" (red minus icon), "Tags [Cloud]" (globe icon), and "Search [Simple]" (magnifying glass icon, highlighted in blue). To the right of the toolbar is a "Search [Advance]" button. Below the toolbar, the section "Search by keyword" contains a text input field labeled "keyword(s):" and a "Search" button. Below the input field is a table with three columns: "Resource Name", "Exist", and "Details". The table is currently empty.

Resource Name	Exist	Details
---------------	-------	---------

Usage Scenario 1– Search (advance)

UNICORE :: Meta-data

Create Metadata
 Update Metadata
 Delete Metadata
 Tags [Cloud]
 Search [Simple]
 Search [Advance]

Search metadata with options.

All these words:

Any of these words:

Exactly these words:

Similar to:

Unwanted words:

Add Date/Numeric

Field	Operator	Value	Value
<div>▼</div>	<div>▼</div>		

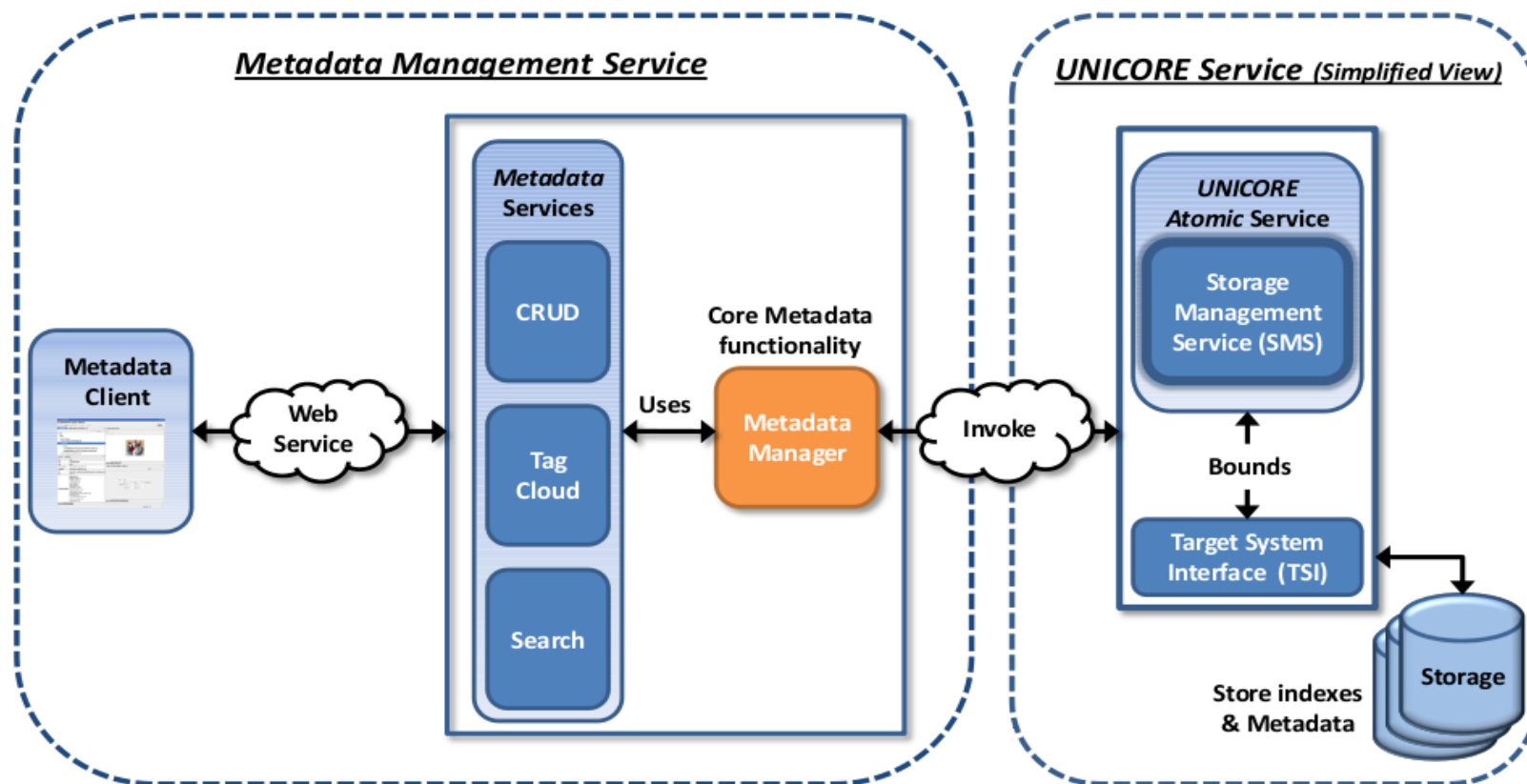
Reset Fields

Search

Results:

Document Name	Description

Usage Scenario 2 – Catalog Service



Future Work 1

- More types of languages support for indexing.
- Asynchronous process for indexing data.
- Integration of Metadata Graphical Client (MGC) with UNICORE Rich Client (**URC**).
- Enabling metadata support for other UNICORE services.

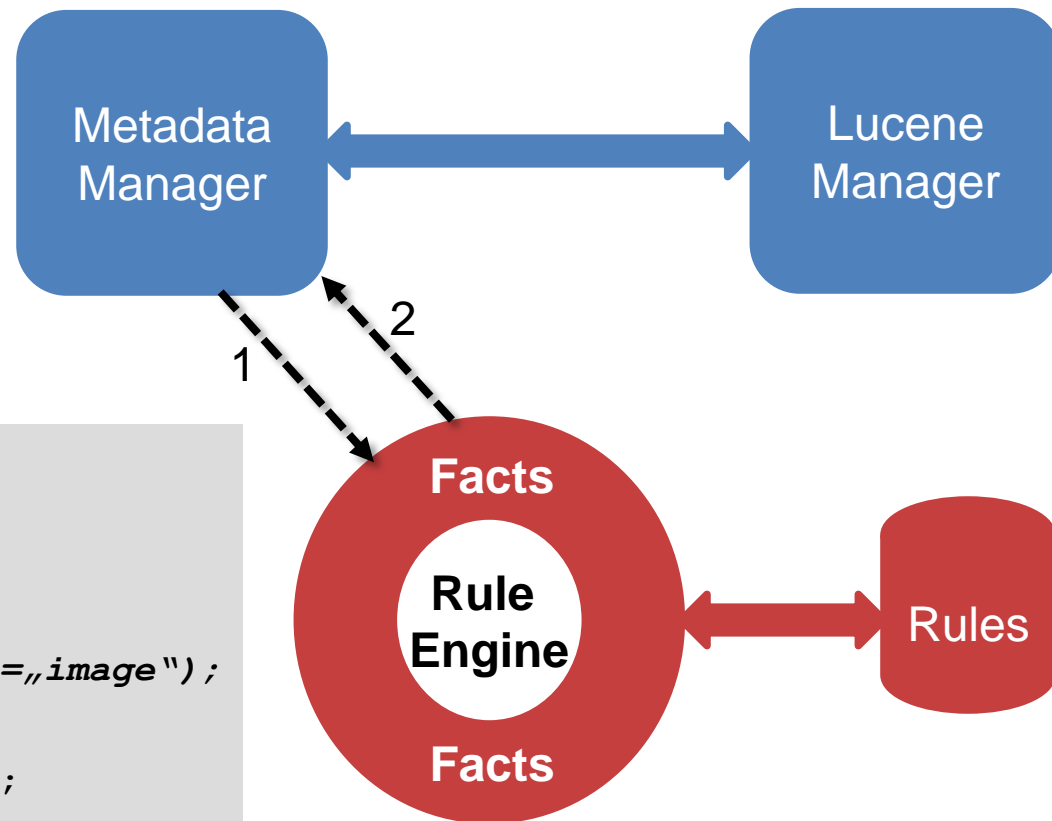
Future Work 2

Rule based selection of data to be indexed.

- 1. Forward request (what kind of data to be indexed?)
- 2. Based on *rules*, rule engine respond to Metadata Manager.

```
// Rules

Rule „file_type_image“
  when
    file : new File(type=„image“);
  then
    return „Not Allowed“;
end
```



References

- [1] <http://uvos.chemomomentum.org/index.html>
- [2] <http://amga.web.cern.ch/amga/>
- [3] <http://www.fz-juelich.de/jsc/grid/#GridRDProjects>
- [4] <http://lucene.apache.org/>

Questions ?

