Enhancing UNICORE Storage Management using Hadoop Distributed File System

Wasim Bari[2], Ahmed Shiraz Memon[1], Dr. Bernd Schuller[1]

1. Jülich Supercomputing Centre, Forschungszentrum Jülich
&
2. Institute of Scientific Computing, RWTH Aachen

Tuesday, 25th August , 2009

SCIENTIFIC COMPUTING

JÜLICH FORSCHUNGSZENTRUM

RWTH AACHEN UNIVERSITY

Contents

- Motivation
- Hadoop
- UNICORE
- UniHadoop
- Demonstration
- Summary

Motivation

- **Computation** power has increased exponentially
- More Computation with advanced technologies
  - Grid Computing, Parallel computing, High Performance Computing
    - Environment Modeling
    - High Energy and Nuclear Physics
    - And many more in “Feasible” time
- Applications are producing HUGE **data**
  - Big Bang, Large Hadron Collider at CERN – **15 Petabyte** annually

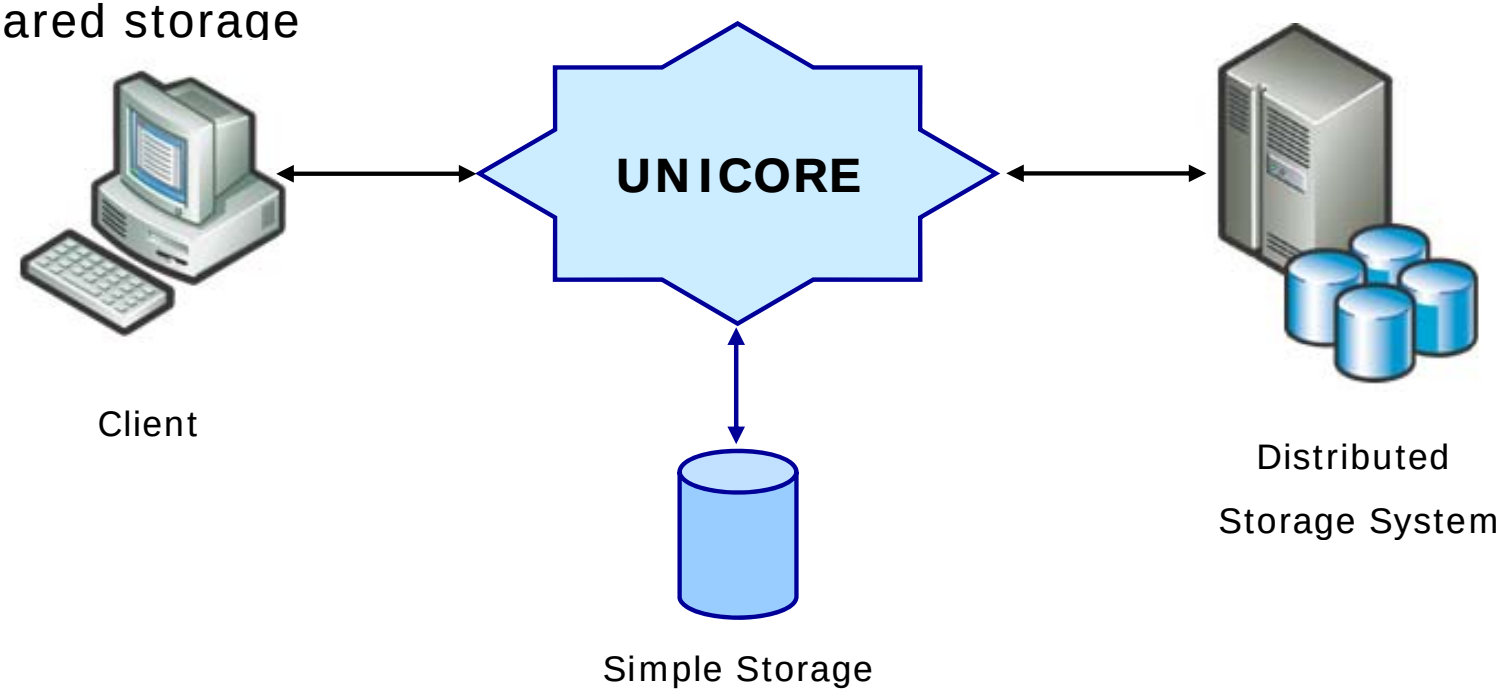
Motivation

- **Traditional Data Storage Systems** cannot store such amount of data
  - Limited Storage
  - Disaster Recovery
  - Number of Files
  - Parallel Access

- **Distributed Data Storage System**
  - Elastic expandable Storage
  - Disaster Recovery
  - Fast and Parallel Access
  - Durable
  - Replication
  - Number of Files

Motivation

- UNICORE supports simple Storage mechanisms with the Filesystem located at TargetSystem Site
- Becomes complicated while using complex workflows
- Goal: Allows UNICORE to support Distributed Storage System as big shared storage



Hadoop

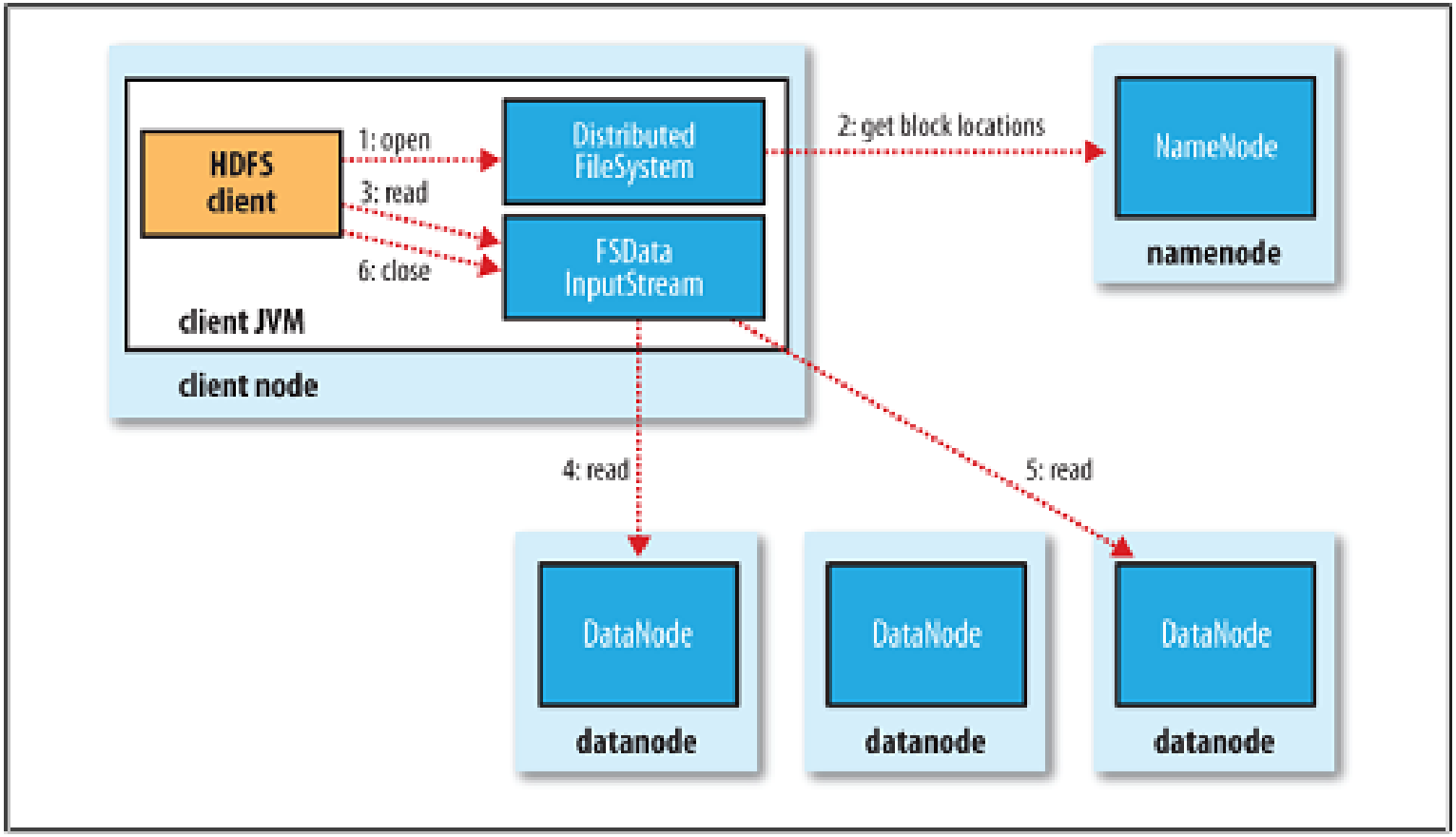
- In 2003 Google Proposed a highly scalable, fault tolerant, dynamic Distributed File System called Google File System (GFS).
- In 2004, Google introduced MapReduce programming model, A mechanism to achieve parallelism.
- Apache Hadoop is an Open Source implementation of GFS and MapReduce
- Main contributor is Y!
- Hadoop Distributed File System (HDFS)
  - Scalable
  - Economical
  - Efficient and
  - Reliable

Hadoop Distributed Filesystem (HDFS)

- Very Large Distributed File System
  - 10K nodes, 100 million files, 10 PB
- Master/Worker Architecture
  - Single Master (NameNode)/Many Workers (DataNode)
- Single Namespace for entire cluster
- Blocks: Files are broken up into smaller chunks
  - Typically 128 MB block size
  - Each block replicated on multiple DataNodes
- Client
  - Finds location of blocks
  - Accesses data directly from DataNode

– Security
  - Permission model similar to POSIX

HDFS: Reading a file



Courtesy: Hadoop: The Definitive Guide, *O'Reilly publishers*, First Edition, 2009

Hadoop

- Abstract Filesystem concept

- Supported File- and Storagessystems
  - Hadoop Distributed File System **(hdfs://)**
  - Cloudera (formerly Kosmos File System) **(kfs://)**
  - Amazon S3 File System
    - S3 Native File System **(s3n://)**
    - S3 Block File System **(s3://)**
  - Local disk **(file://)**
  - FTP File System **(ftp://)**

- Runtime selection of one of the supported Storagesystems

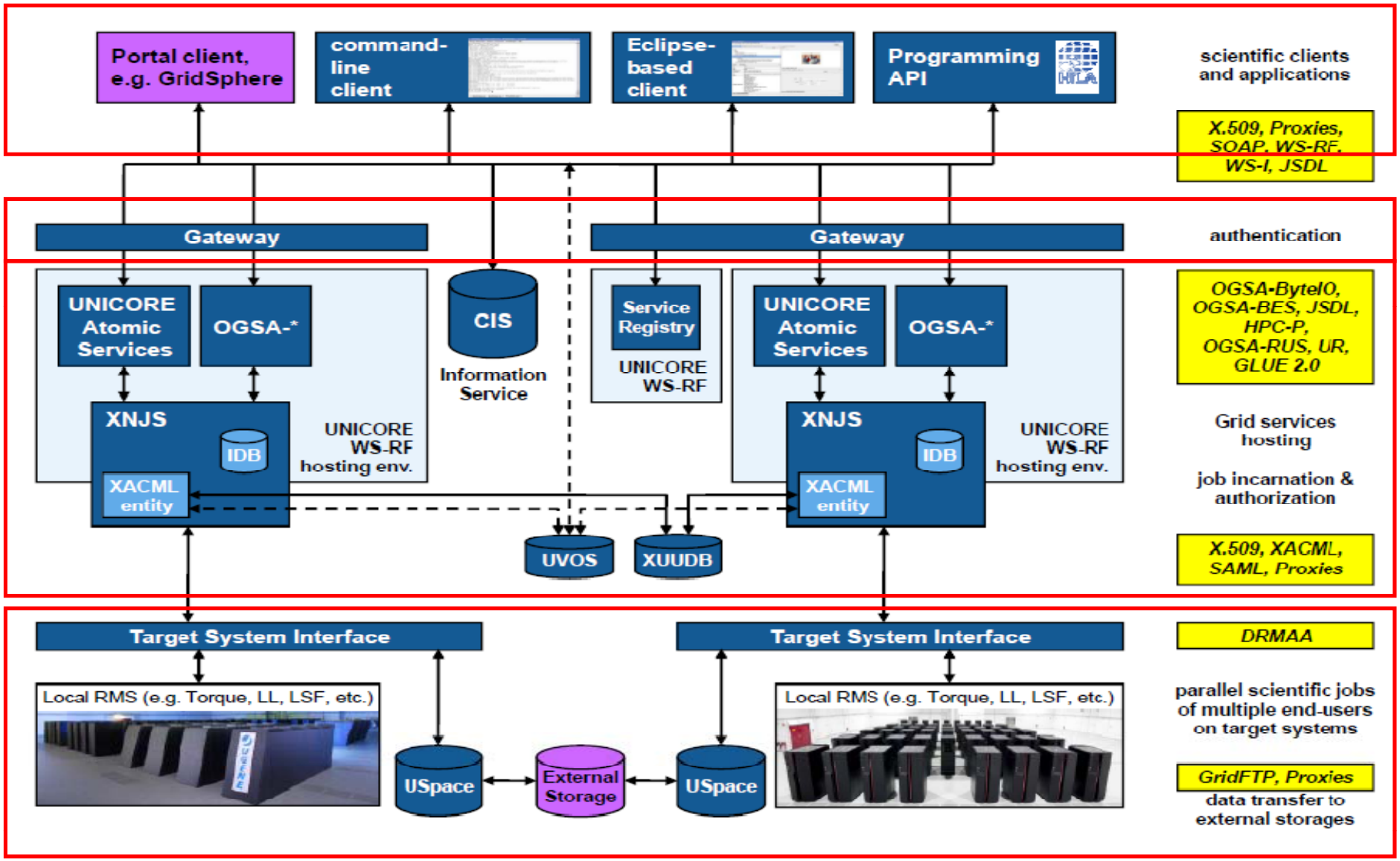
UNICORE

- A Ready-to-Run, Open Source Grid Middleware that enables seamless and secure access to the Computing resources

- UNICORE Atomic Services (UAS)
  - **Storage Management Service (SMS)**
  - File Transfer Service (FTS)
  - Job Management Service

- Implementation of standards OASIS, OGSA, WS-I etc…
  - BES, ByteIO, GLUE2, WS-RF etc…

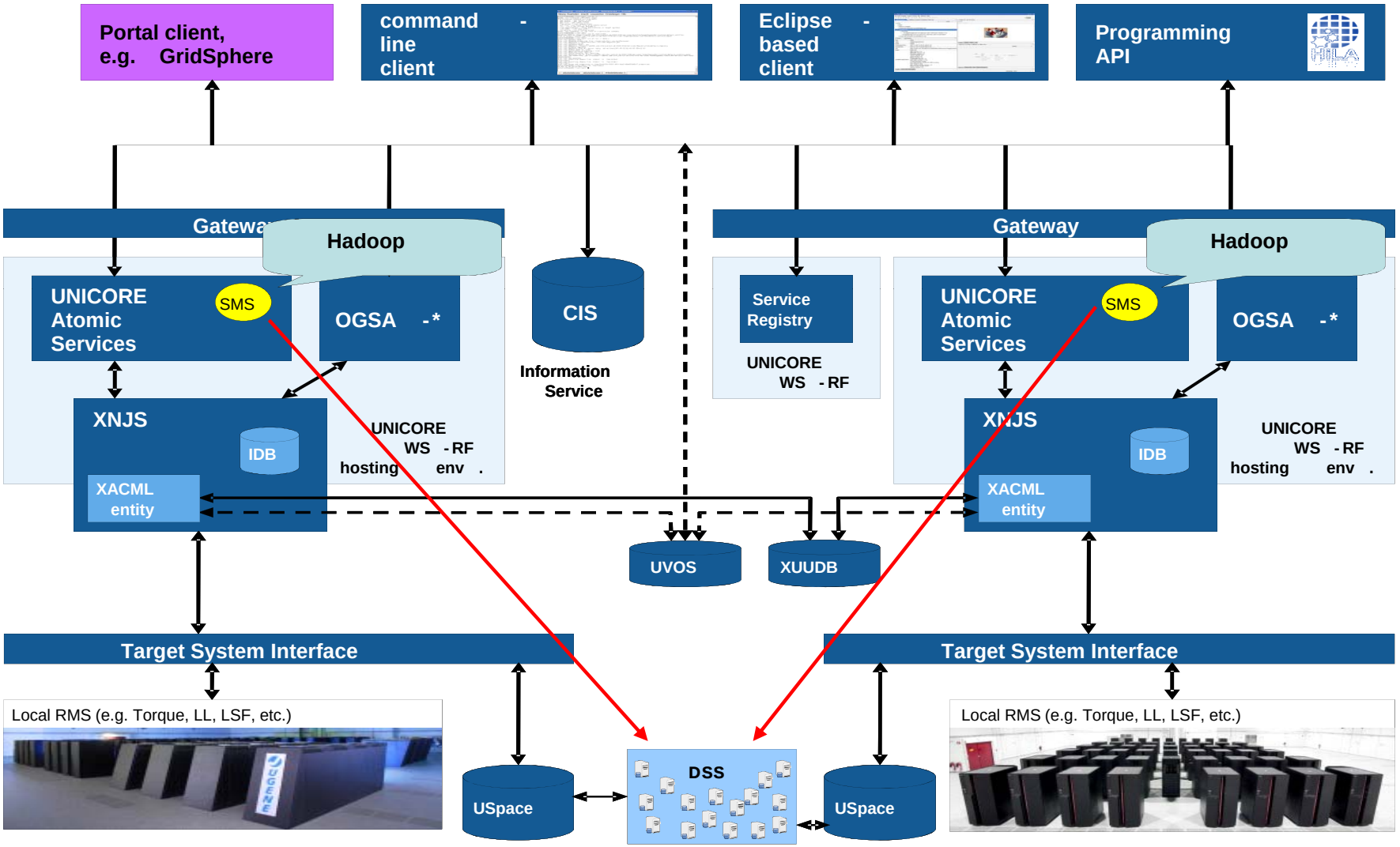
UNICORE Architecture



UniHadoop: Overview

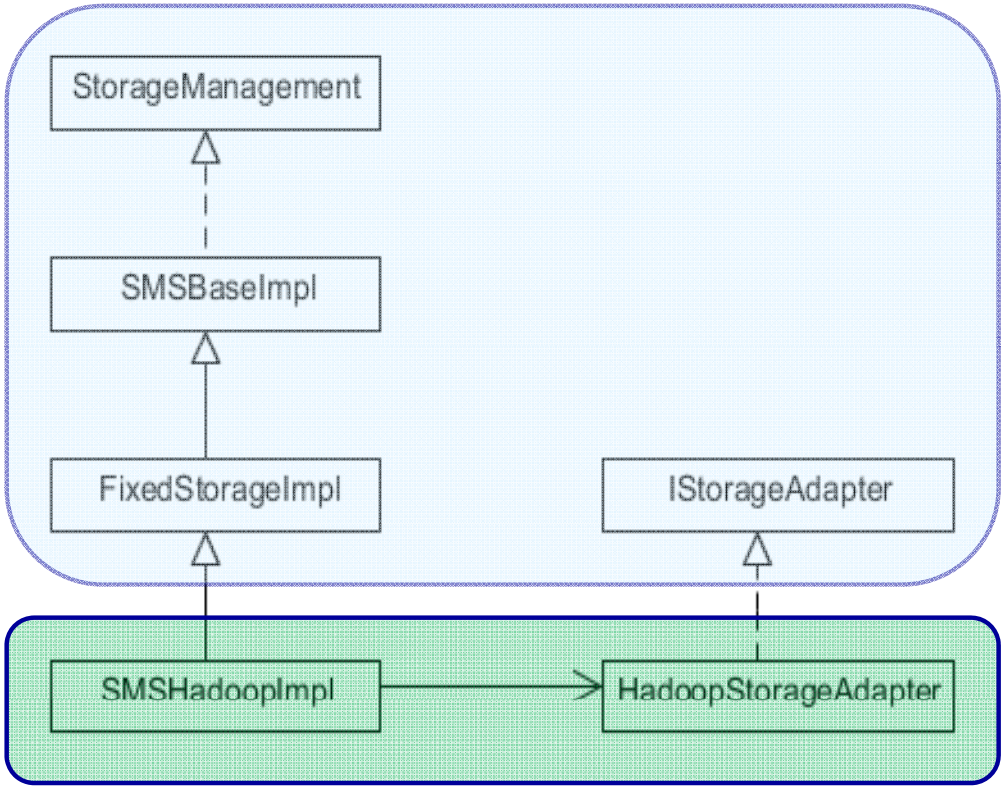
- UniHadoop is an integration of UNICORE with Apache Hadoop
- Inspired by Hadoop design
  - Using an Abstract notion of Filesystem
  - Supports all storage/filesystems
  - Specialisation of UNICORE Storage Management Service (SMS).
  - Enable UNICORE clients (UCC, URC etc…) to use Hadoop transparently

UniHadoop: Architecture



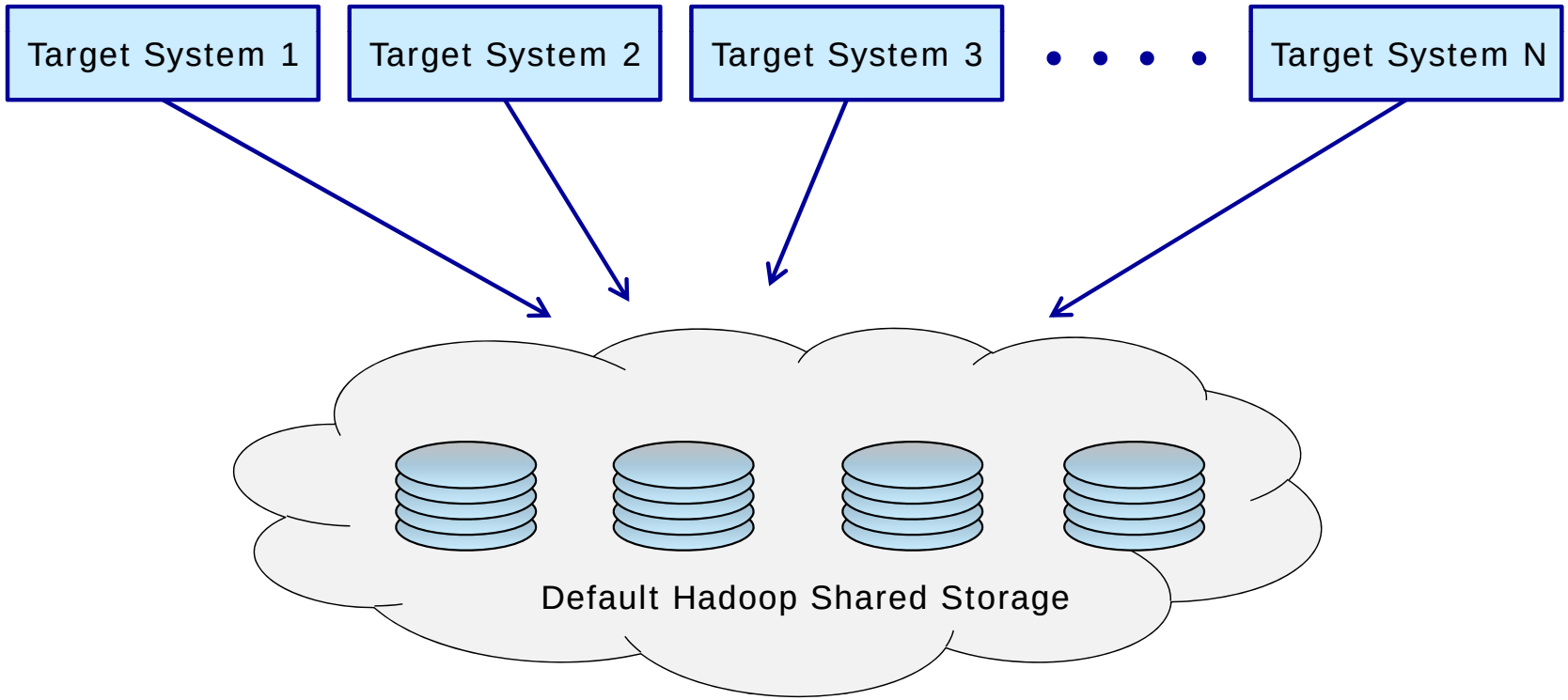
UniHadoop: Design

- StorageManagement: Base Interface
  - List directory
  - create directory
  - change permissions etc...
- SMSBaseImpl: Basic implementation
- FixedStorageImpl: A storage service files for fixed path, such as „/work“
- SMSHadoopImpl: SMS Adaptation for Hadoop
- IStorageAdapter: Interface for accessing low level hierarchical storage systems
- HadoopStorageAdapter: HDFS specific implementation



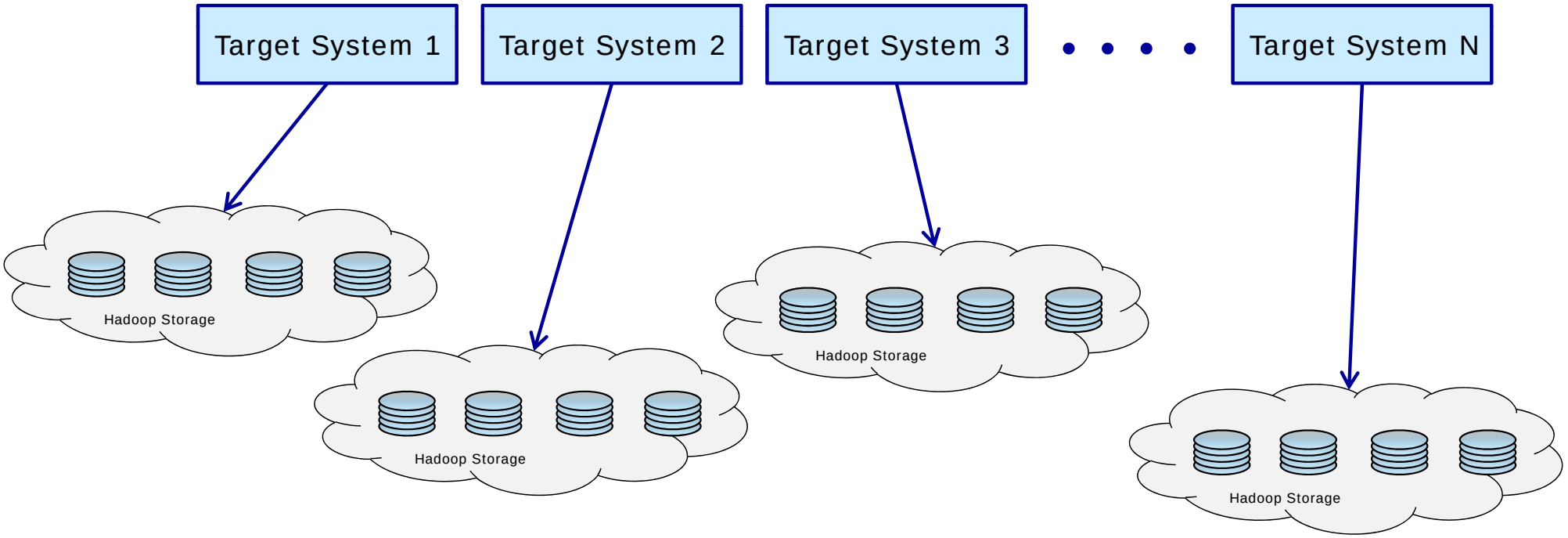
UniHadoop: Usage Scenario 1

- Default Shared Storage
- “Huge” workflows



UniHadoop: Usage Scenario 2

- A shared storage for every Target System



Demonstration

- Listing directories/files on HDFS
- Creating files
- Submitting jobs using data from HDFS

Feature Matrix

Features	Default SMS	Hadoop SMS
Unlimited Storage	X	✓
Durability	X	✓
Unlimited File Size	X	✓
Disaster Recovery	X	✓
Unlimited No. Of files	X	✓
Trash Recovery	X	✓

Summary

- Requirements
- Architecture and Design
- Usage scenarios
- Hadoop integration provide support for
  - MapReduce
  - Data analysis and intelligence using Pig, Hive and HBase
  - A number of available File- and Storagesystems
    - Amazon S3, Cloudera, KFS etc…

Q & A

Questions

Contact

a.memon@fz-juelich.de

UNICORE-Hadoop:

Bundled in release 6.2.1 server, www.unicore.eu

Source:

http://unicore.svn.sourceforge.net/viewvc/unicore/unicorex/trunk/uas-hadoop/