Towards UNICORE 8
development directions

Bernd Schuller
b.schuller@fz-juelich.de

UNICORE Summit 2015, Karlsruhe, June 7, 2015
Outline

• New features
  – Python TSI
  – Apache Hadoop

• Plans
  – CDMI

• Current issues towards UNICORE 8
UNICORE The TSI

other UNICORE servers  command-line client  Eclipse-based client

UNICORE/X  connect  callback  TSI

Cluster access node(s)

Local RMS (e.g. Torque, LL, LSF, etc.)
Python TSI - motivation

• Current Perl TSI
  − basic concepts are (still) very good!
  − well tested in production

• … but
  − Perl code is hard to maintain
  − Next to no unit tests
  − SSL support in Perl is horrible
  − The Perl code's license is kind of unclear
Python TSI - goals

- Clean re-implementation
  - unit tests
- Keep existing configuration as far as possible
- Improve
  - SSL support
  - Structure and readability
  - Extensibility: adaptation to local should be in one place
Python TSI: status

• Working
  – both Python 2.7 and 3.x support
  – Nobatch, Slurm and Torque versions
  – Packaging (one package per BSS)

• TODOs
  – SSL certificate pinning (only allow particular XNJSs to connect)
  – SGE version
  – New feature: computing time budget
  – Review documentation
Apache Hadoop

- Large-scale, distributed „Big Data“ framework
- Both data and compute functionality
  - HDFS (Hadoop Distributed File System)
  - YARN (Yet Another Resource Negotiator)
- Basis for many applications and additional frameworks (e.g. MapReduce, Apache Spark, ...)

Hadoop - HDFS

• Distributed storage
• Master / slave architecture
  - NameNode for metadata
  - DataNodes stores the data
• Replication, fail-over, etc
Hadoop - HDFS

HDFS Architecture

Metadata (Name, replicas, …): /home/foo/data, 3, …

Client

Read

Datanodes

Block ops

Namenode

Replication

Datanodes

Blocks

Write

Rack 1

Rack 2

http://hadoop.apache.org/docs/current/
Hadoop - YARN

• Resource management
  – ResourceManager
    • arbitrates resources among the applications in the system
  – NodeManager
    • one per node – monitoring resource usage

• Job management
  – scheduling, starting, monitoring, ...
Hadoop - YARN

http://hadoop.apache.org/docs/current/
UNICORE / Hadoop integration
(when UNICORE can access the Hadoop services)
UNICORE / Hadoop integration alternative
(when Hadoop services are not directly accessible via the network)

UNICORE Client

UNICORE/X Server
  XNJS  →  SSH libs

run Hadoop commands on login node

YARN HDFS NameNode
  Worker  Worker  Worker
  Worker  Worker  Worker
  Worker  Worker  Worker
Hadoop – UNICORE integration: goals

- HDFS as storage backend
  - already done previously, only updates required

- YARN as „batch system“

- „UNICORE-like“ application support
  - IDB, Generic gridbean, portal support, ...
Demo – FSD Testgrid

- HDFS as storage backend
- YARN for running jobs
- Apps configured in IDB
Hadoop – UNICORE integration: status

• HDFS / YARN work, using latest API v2.7.0
• Unit tests with embedded Hadoop, also tested on FSD cloud testbed
• Features:
  – HDFS can be used as Uspace and normal storage, including storage factory
  – Yarn applications can be defined in IDB
• Target release 7.5.0
Hadoop – UNICORE integration: TODOs

- File permissions
  - Better way? Multiuser support?

- Documentation
  - Both admin and end-user

- More example apps
  - Implement real life use cases
CDMI

- Cloud Data Management Interface
- Implement as SMS back-end
- Username/password authentication
- Collaboration with
  - dCache (CDMI server)
  - TU Dresden (use case)
- Not started yet, some old prototype code exists somewhere :-)


Towards UNICORE 8

- Remove dependency on XmlBeans?
  - Not maintained any more, upcoming issues might not be solveable
  - Replace by JAXB (part of JDK)
  - Very high effort!
    - SAML code / security library
    - XNJS, JSDL, Brokering
    - WS interfaces (core, workflow, clients, ...)

Towards UNICORE 8

• Further increase usage of REST API
  – e.g. SOAP/XML to setup a security session
  – use REST API during security session lifetime

• Simple notification mechanism
Thanks

• Tim Kreuzer (Hadoop integration)