Mitglied der Helmholtz-Gemeinschaft

UNICORE

JÜLICH
FORSCHUNGSZENTRUM

Federated Access to High-Performance Computing and Big Data Resources

Bernd Schuller
Federated Systems and Data division,
Jülich Supercomputing Centre

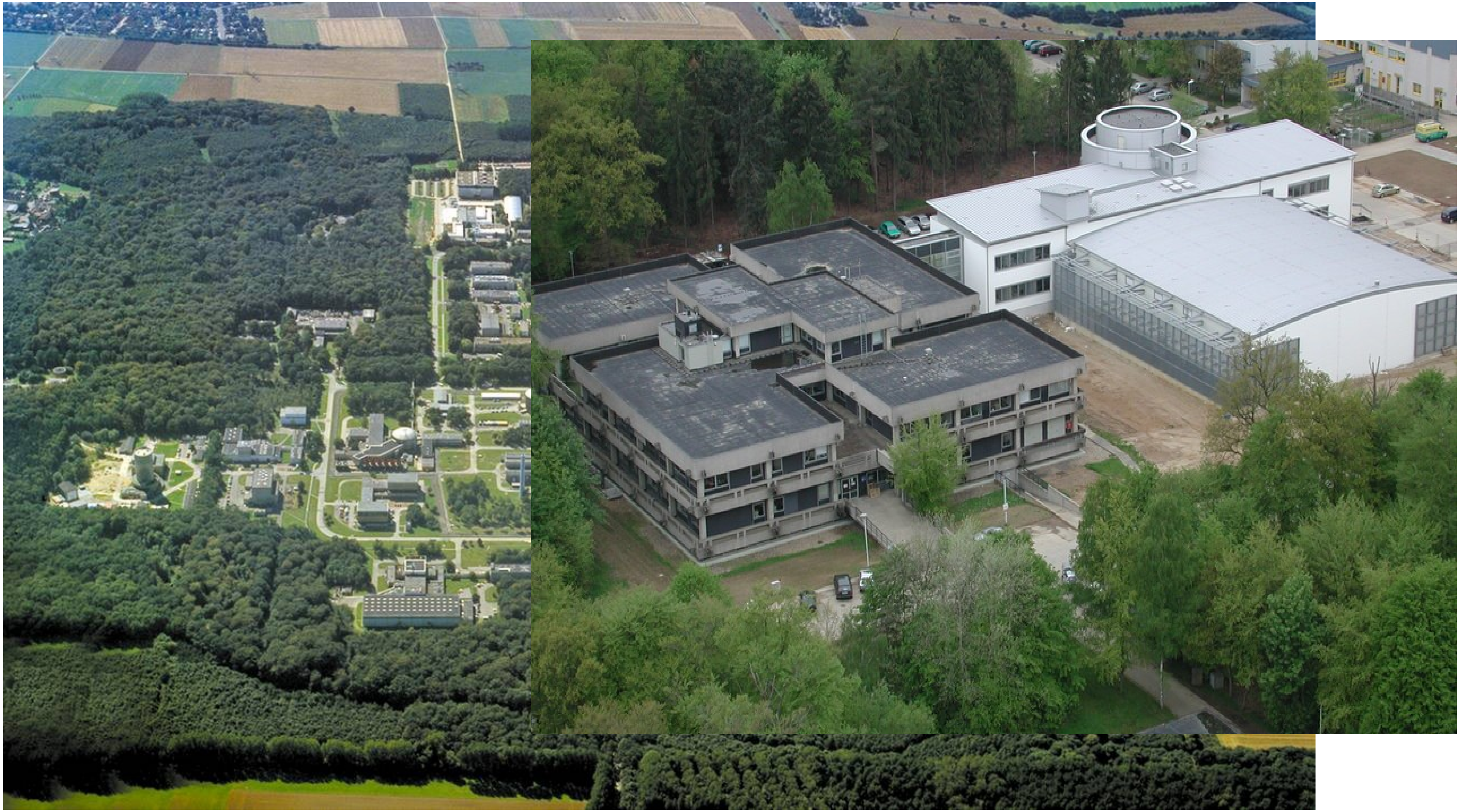
AFIN 2014, Lisbon November 17, 2014

companies
industry
way
traffic
IMAGERY
HAVING
REASONABLE
FACTORS
documents
BETTER
presenting
STORAGE
REQUIREMENTS
advantage
RECOMMENDATIONS
capacity
HDFS
cost
PROBLEM
DEVELOPMENT
throwing
programming
ENABLING
retrieving
signals
Hadoop
gain
EXPERIMENTATION
STREAMING
involves
DIVERSE
USERS
practitioners
PROBABLY
analysis
established
Culture
based
insights
COMMON
matters
FEEDBACK
TIME
SCIENTISTS
MAPREDUCE
BIG
VOLUME
marketplaces
Building
Google
HIDDEN
CONVENTIONALLY
DECISION
DATABASE
MAKES
NETWORKS
SCIENTIFIC
suited
exploiting
ANALYZING
static
CHARACTERIZES
INTERACTIVE
ORGANIZATIONAL
DEPLOYMENTS
GENERATING
EVOLVING
AMOUNTS
ABILITY
LOOK
ENTERPRISING
practice
STAGE
turned
models
DATA
PROCESSING
INPUT
bring
SCIENCE
EXPLORED
FORM
instance
ADVERTISING
YAHOO
SCHEMAS
STARTUPS
ISSUE
COMBINES
importance
agility
SOURCE
DIFFERENT
financial
cloud
categories
SETTING
reach
RESOURCES
DATASET
creating
CURIOSITY
typical
PLATFORM
AVAILABLE
teams
CLEANING
expertise
DIRECTLY
extract
ANALYTICAL
USE
revealed
IMMEDIATE
BENEFITS
DISCIPLINE
IMPLEMENTS
STORE
large
INCONSISTENCY
FAST
SUCESSFULLY
house
FRIENDS
AWAY
COMPETITIVE
massively
WEB
PRODUCTS
TAKING
INCREASINGLY
PREDETERMINED
TRANSACTIONS
things
variable
SERVICES
CUSTOMERS
ORGANIZATION
VARIETY
NEED
UNCOMFORTABLE
CONSUMERS
WAREHOUSES
WANT
INFORMATION
CLEVERNESS
INVESTIGATIVE
ENTREPRENEURIAL
STRUCTURED
COMES
INCREASES
ASPECTS
systems
infrastructures
SOFTWARE
PATTERN
UTILIZES
UNDERSTAND
FOLLOWING
COMPILING
EFFECTIVE
SENSORS

Outline

- Jülich Supercomputing Centre
- Example use cases
- Solutions for Federated Access
  - UNICORE : services suite
  - Unity : user authentication and identity management
  - UFTP : high-performance data transfer
  - Clients
  - RESTful APIs

Forschungszentrum Jülich and Jülich Supercomputing Centre (JSC)



JUQUEEN

- IBM Blue Gene/Q
- 28 racks, 458,752 cores

- PowerPC A2 1.6 GHz,
- 16 cores per node

- 5.8 Petaflop/s peak

- 460 TByte main memory

- 5D network

JUST: Juelich Storage Cluster

- IBM-GPFS (General Parallel File System)
- 19.2 PB online storage (15.1 PB net)
- 14,296 disks, MTBF 3 disks per week

- 9.2 PB GPFS Storage System
- Native RAID
- 4,640 NL-SAS + 120 SSD

- Fileserver for
  - HPC systems: JUQUEEN, JUROPA
  - Clusters: JUDGE, JUVIS (visualisation)
  - DEEP (Dynamical Exascale Entry Platform)
  - Big Data collaborations

© JSC

© JSC

Tape Libraries

- Automated cartridge systems
- 45 PB (upgrade to 80 PetaByte)

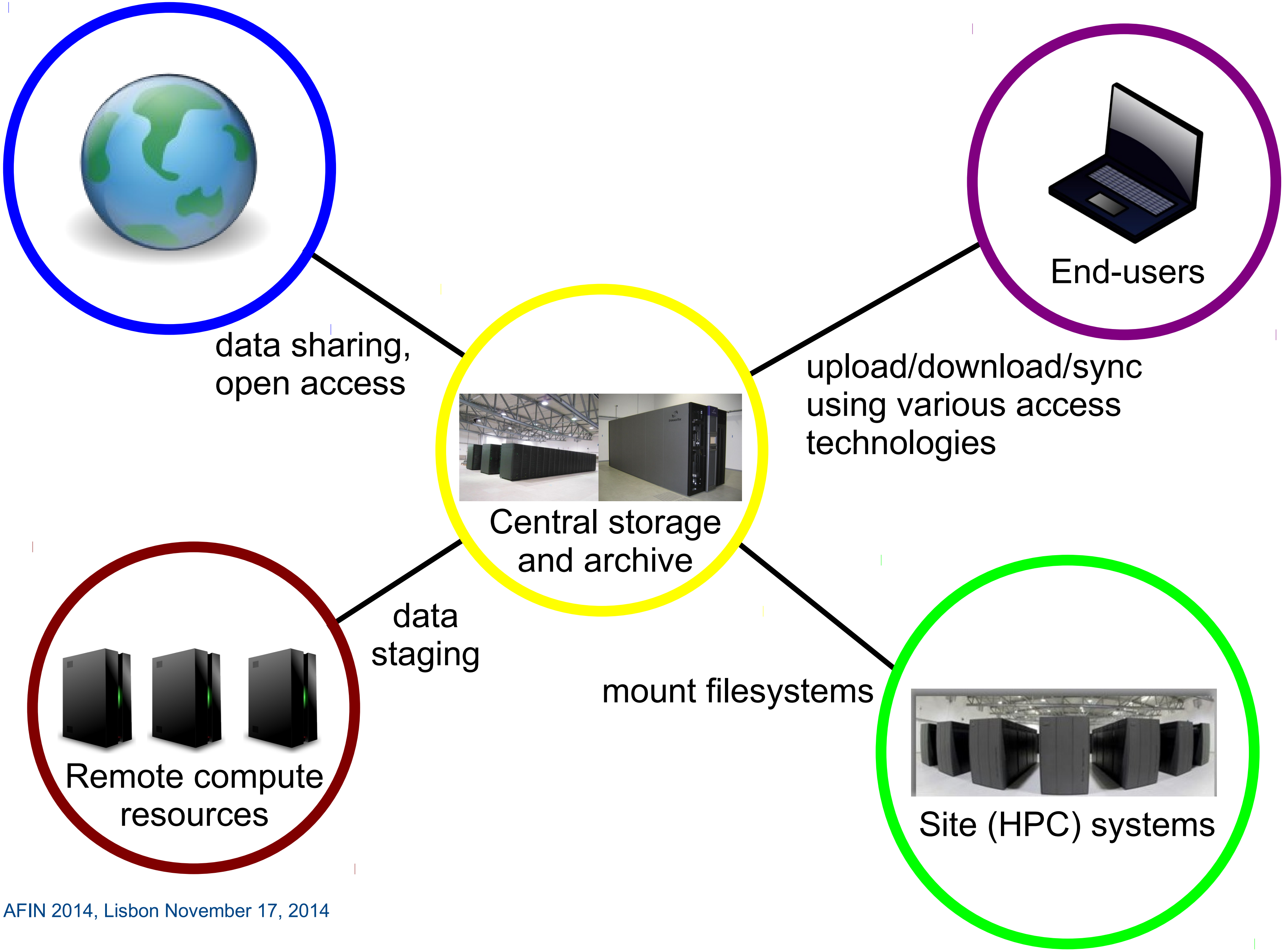
- Used for
  - Backup
  - Long term archive
  - Migration of active (online) data to less expensive storage media

- 2 libraries
  - 16,600 tapes
  - 48 tape drives

© JSC

© JSC

Data centric view



End-users

data sharing,
open access

upload/download/sync
using various access
technologies

Central storage
and archive

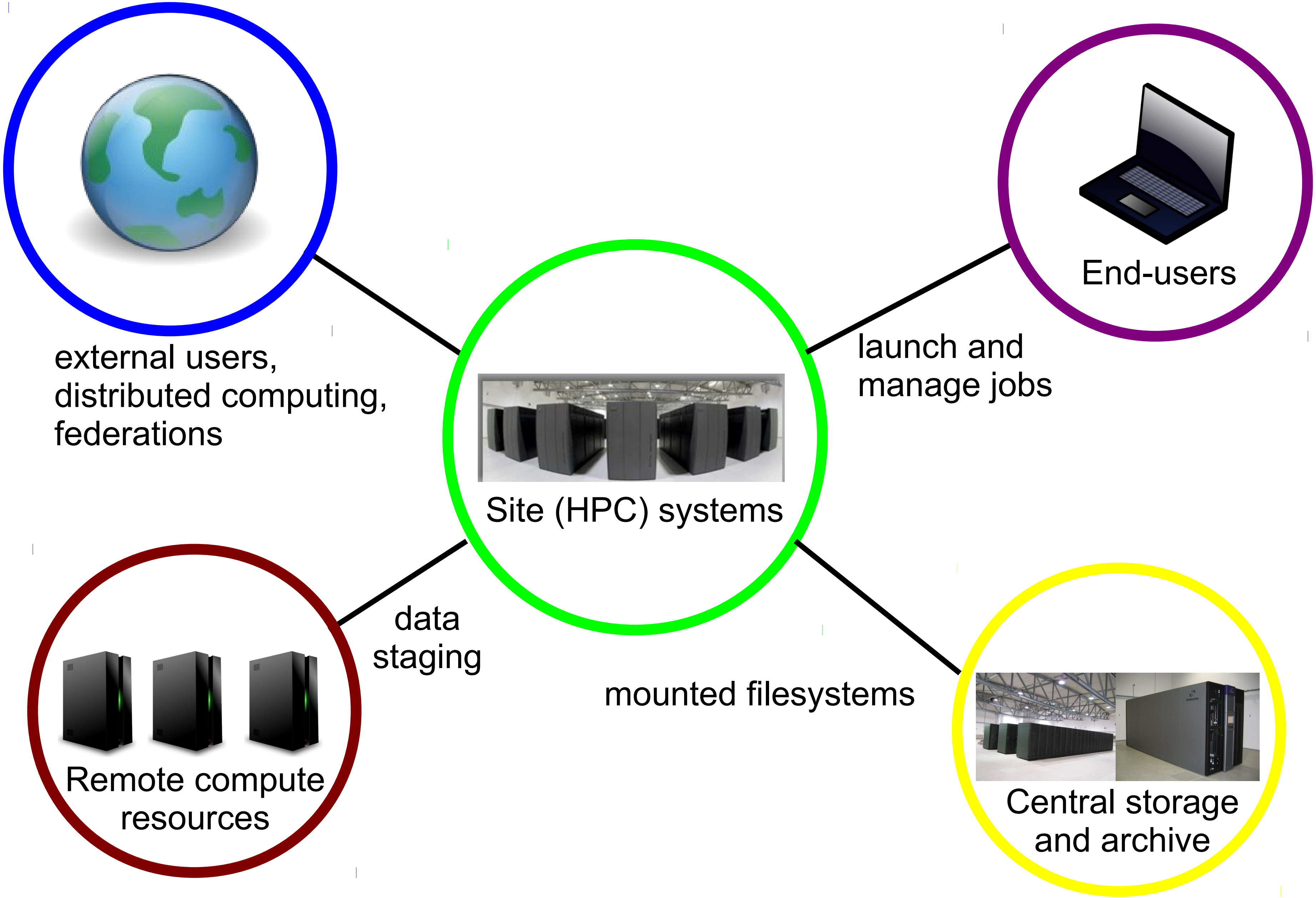
data
staging

mount filesystems

Remote compute
resources

Site (HPC) systems

Compute centric view



Site (HPC) systems

external users, distributed computing, federations

End-users

launch and manage jobs

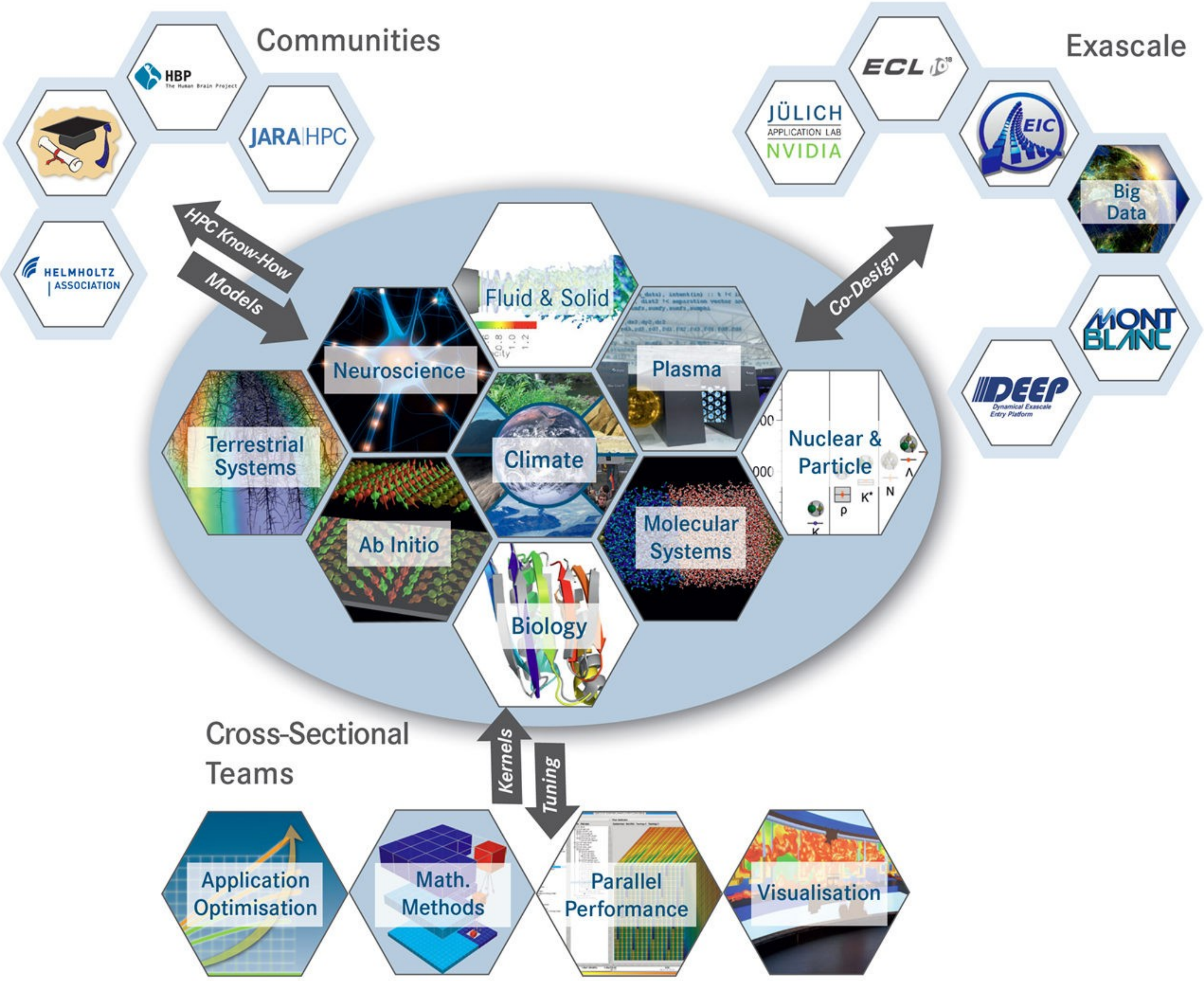
Remote compute resources

data staging

mounted filesystems

Central storage and archive

Application centric: Simulation Labs



Federated Systems and Data

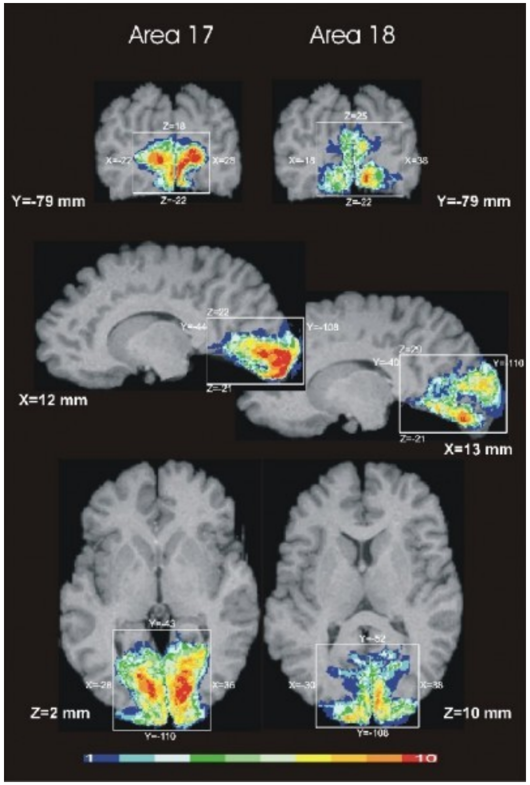
Focus on applications and their requirements in federated environments:

- *Data Management* investigates the data life cycle of applications and strategies, methods, tools and services required for all processing steps.
- *Data Analytics* addresses techniques and methods for analysing Big Data sets.
- *Application Support* deals directly with applications and their integration into distributed environments.
- *Federations* provide a basis for distributed environments by developing the necessary tools and services, e.g. for identity management or data processing models.
- *Standardisation* lays the foundations for the interoperability of federated computing and data infrastructures.

Two use cases from neuroscience

High-throughput brain scans –
a Jülich / Univ. Düsseldorf collaboration

- Goal is to create a 3D brain atlas
- Data aquisition
    - Brain section scans (ex vivo)
      (~2000 slices, 500GB per slice → **1 PB**)
    - MRT scans (in vivo)
- Processing: image registration, calibration, segmentation, etc
- Image processing using HPC
- Raw data often re-processed (new algorithms, new software versions)
- **Plus**: workflows, metadata, sharing with external partners

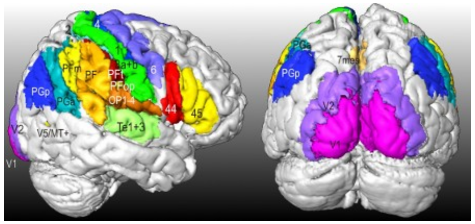


SAY BIG DATA
ONE MORE TIME
memegenerator.net

Human Brain Project

- FET Flagship
- ~10 years, ~1 Billion € (50% EC funding)
- Coordinated by EPFL (Lausanne)
- Huge, multidisciplinary Consortium
  - Neuroscience, medicine, physics, IT, philosophy, ...
  - ~200 partners by Y5
- www.humanbrainproject.eu

© EPFL/HE

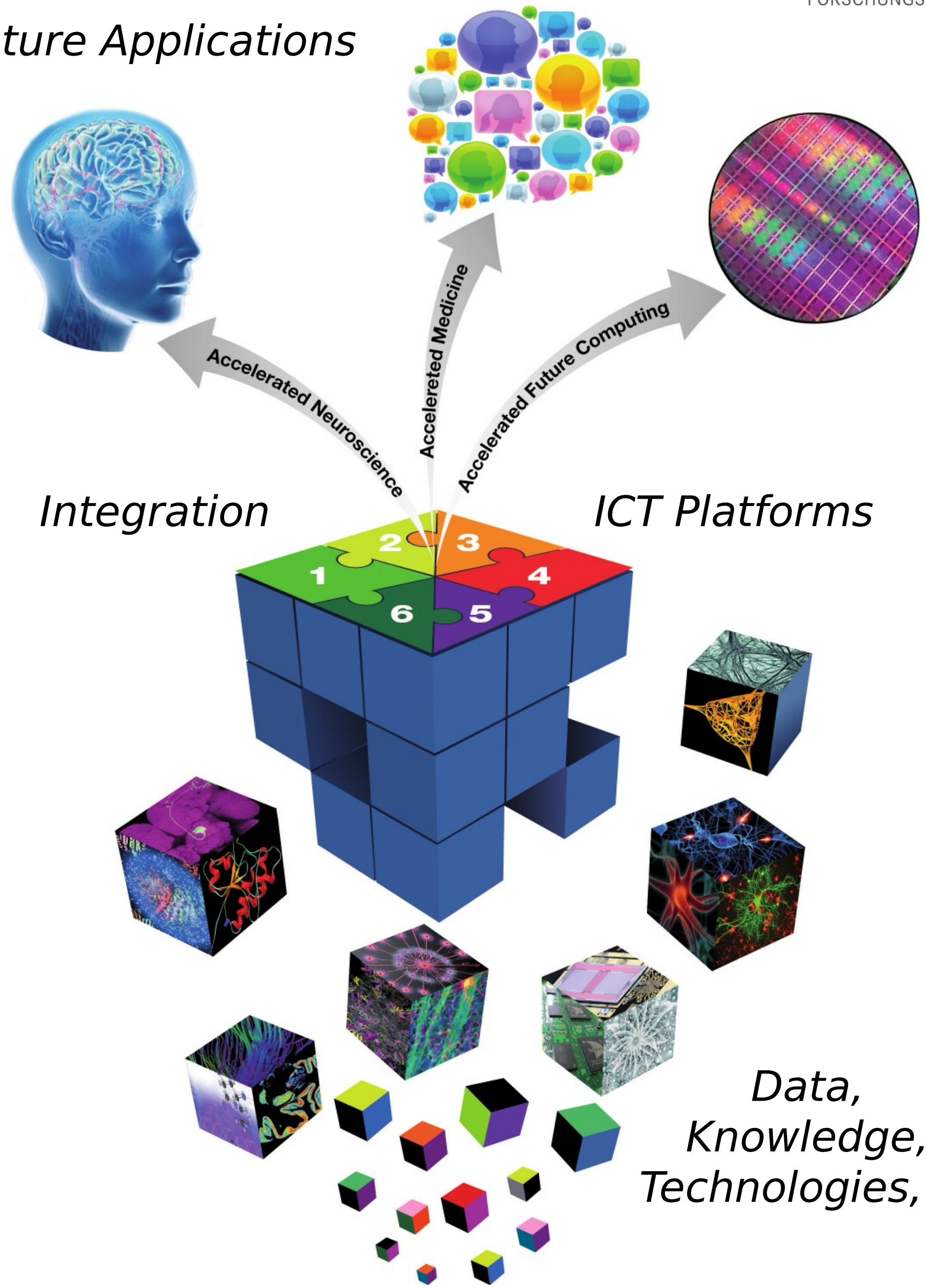


HBP Goal

To build an integrated ICT infrastructure enabling a

Global collaborative effort towards understanding the human brain, and ultimately

To emulate its computational capabilities



Future Applications

Integration

ICT Platforms

*Data,
Knowledge,
Technologies, ...*

HBP High performance computing platform

Technology evaluation and deployment of HPC systems

Main production system at Jülich (Exascale capability around 2021/22) plus facilities at CSCS, BSC, CINECA

Applications requirements analysis, subcontracting for R&D and prototypes

Mathematical methods, programming models and tools

Parallel and distributed programming models, work flows, middleware for resource management, performance analysis & prediction, numerical algorithms for neuroscience

Interactive visualization, analysis and control

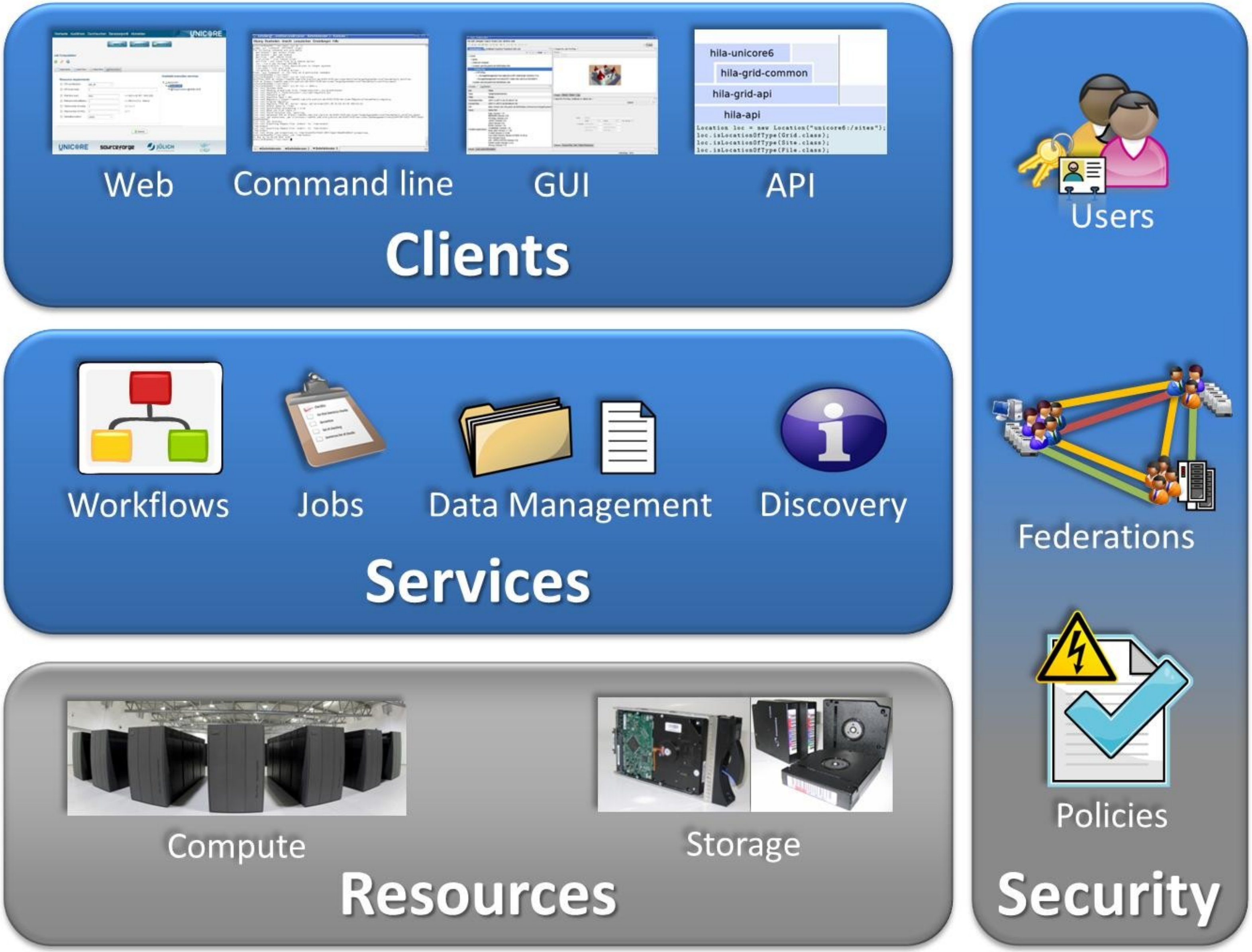
In-situ visualization and interactive steering and analysis of simulations

Exascale data management

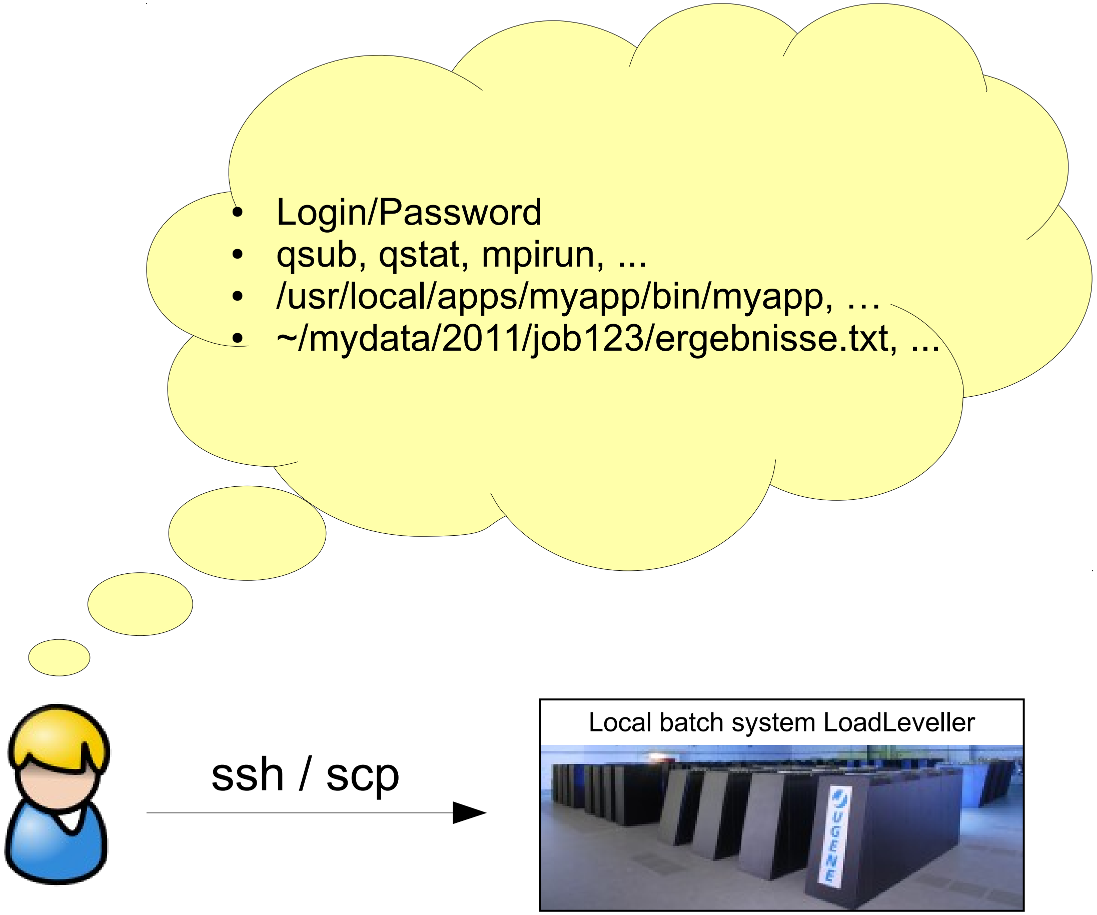
Scalable querying of datasets, data analytics, data provenance and preservation

Brain-inspired supercomputing

Solutions



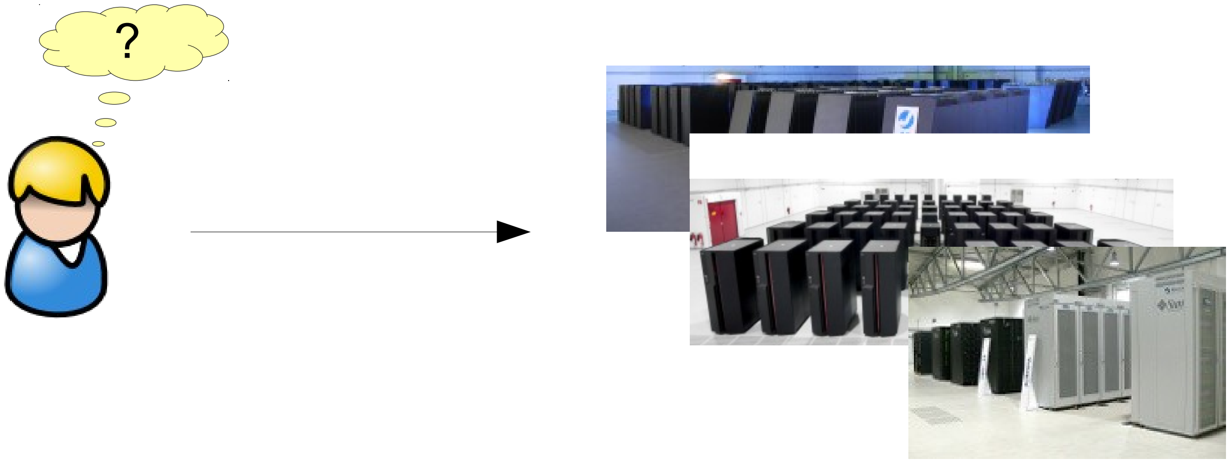
```
Location loc = new Location("unicore6:/sites");
loc.isLocationOfType(Grid.class);
loc.isLocationOfType(Site.class);
loc.isLocationOfType(File.class);
```



- Login/Password
- qsub, qstat, mpirun, ...
- /usr/local/apps/myapp/bin/myapp, …
- ~/mydata/2011/job123/ergebnisse.txt, ...

ssh / scp

Local batch system LoadLeveller



How can I ...

- … use multiple, heterogeneous systems seamlessly,
- … manage my job input data and results?
- … across systems? Workflows?
- This was the original motivation for developing UNICORE (1997)

A federation software suite

- Secure and seamless access to compute and data resources
- Excellent application and workflow support
- Complies with typical HPC centre policies
- Wide variety of clients: GUI, commandline, APIs, ...
- Java/Perl based, supports UNIX, MacOS, Windows and many resource management systems (Torque, Slurm, SGE, …)
- Easy to install, configure, administrate and monitor
- Small, active developer team, responsive to user wishes :-)

- **Open source, BSD licensed, visit http://www.unicore.eu**

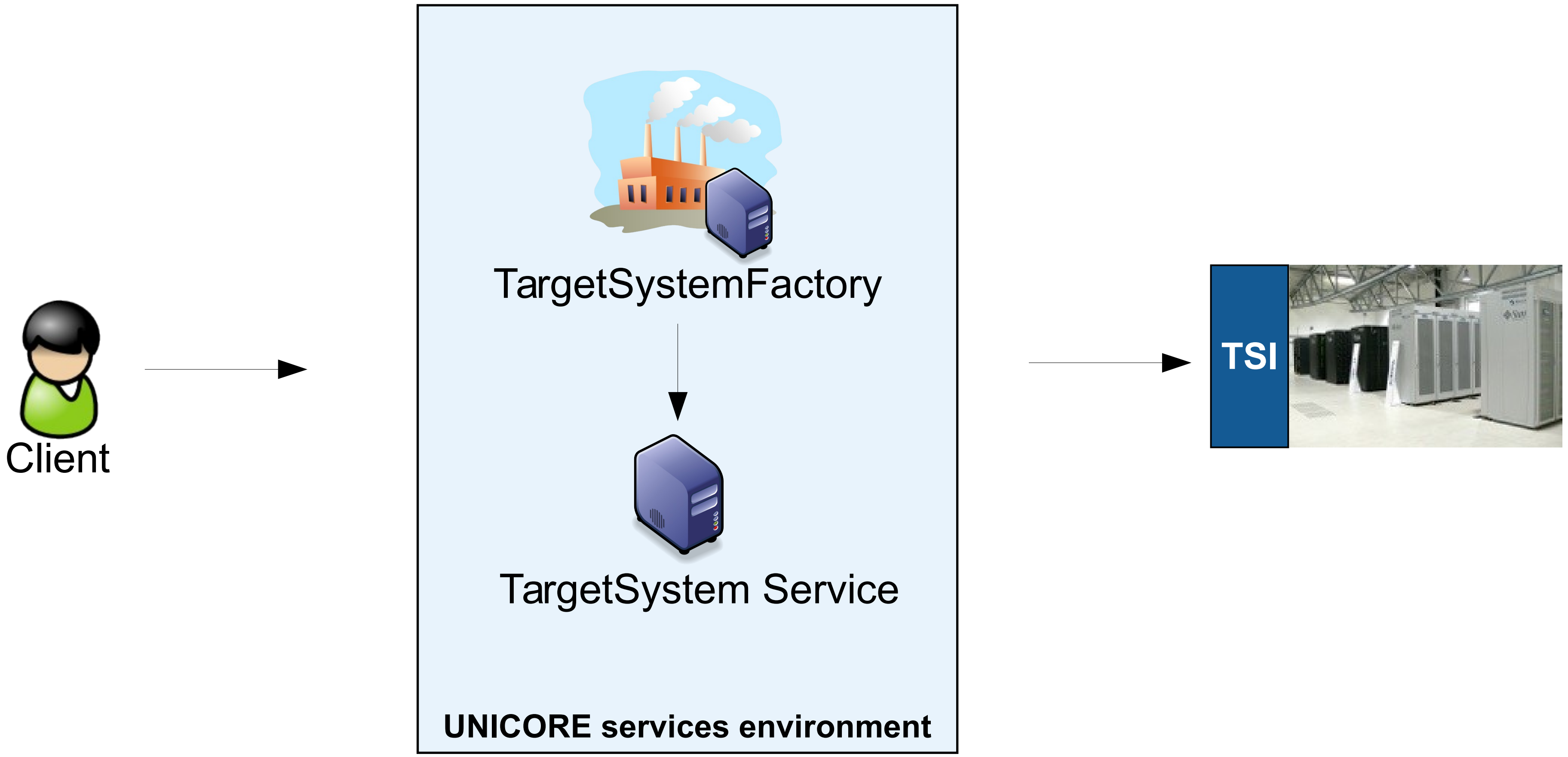
A (subjective) UNICORE timeline

- **1996** (mythical past) : first UNICORE project (Germany only)
- 2002 : UNICORE 4/5 → Eurogrid project, UNICORE goes Open Source, I started to work on the OpenMolGRID project
- 2005-2007
  - **UniGRIDS** project : UNICORE WS(RF) interfaces defined
  - **UNICORE 6.0** release
- Deployment in PRACE, XSEDE and other HPC infrastructures (national Grids, e.g. PL-Grid)
- **2013 : UNICORE 7.0 release**
- … and we're still going (thanks to projects and institutional funding)

UNICORE: Main services

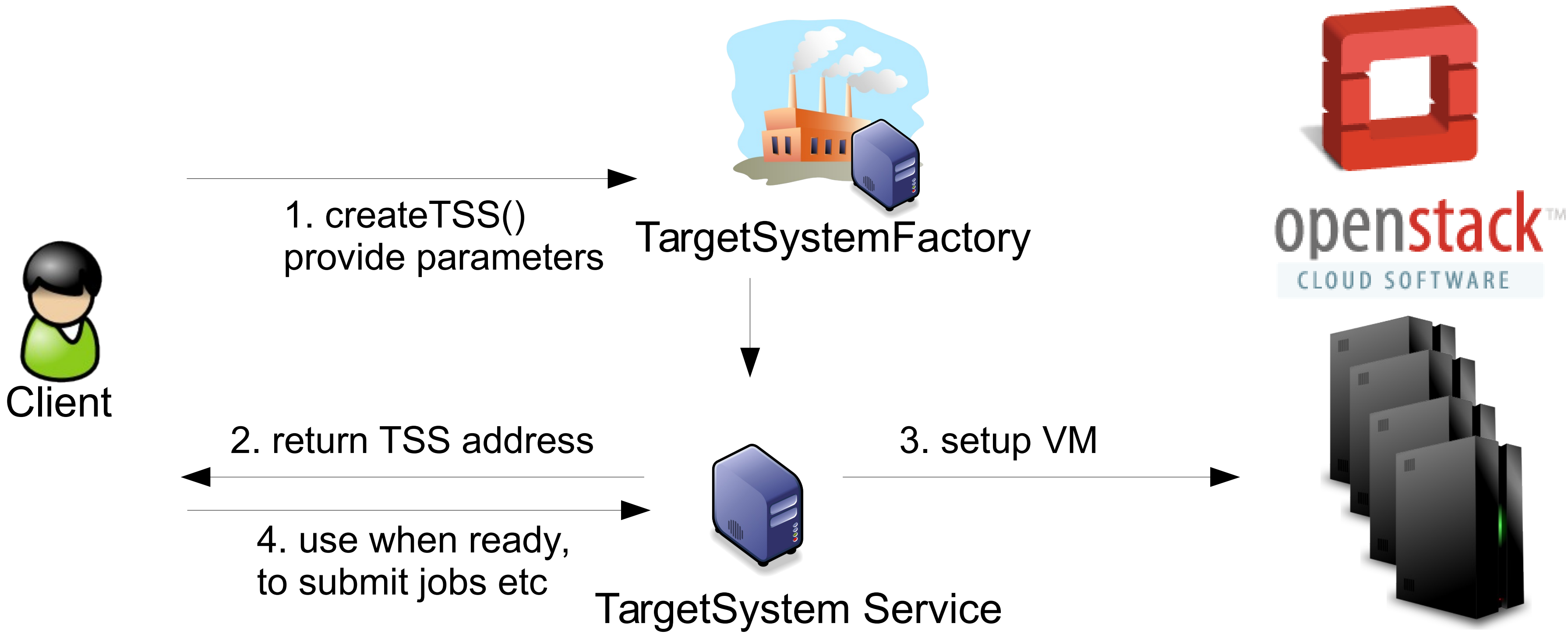
- Compute
  - TargetSystemFactory
  - TargetSystem
  - JobManagement
  - Reservations
- Storage and data
  - StorageFactory
  - StorageManagement
  - FileTransfer
  - Metadata
- Workflow
  - Workflow enactment
  - Task execution
  - Resource Broker
- Registry

Default setup



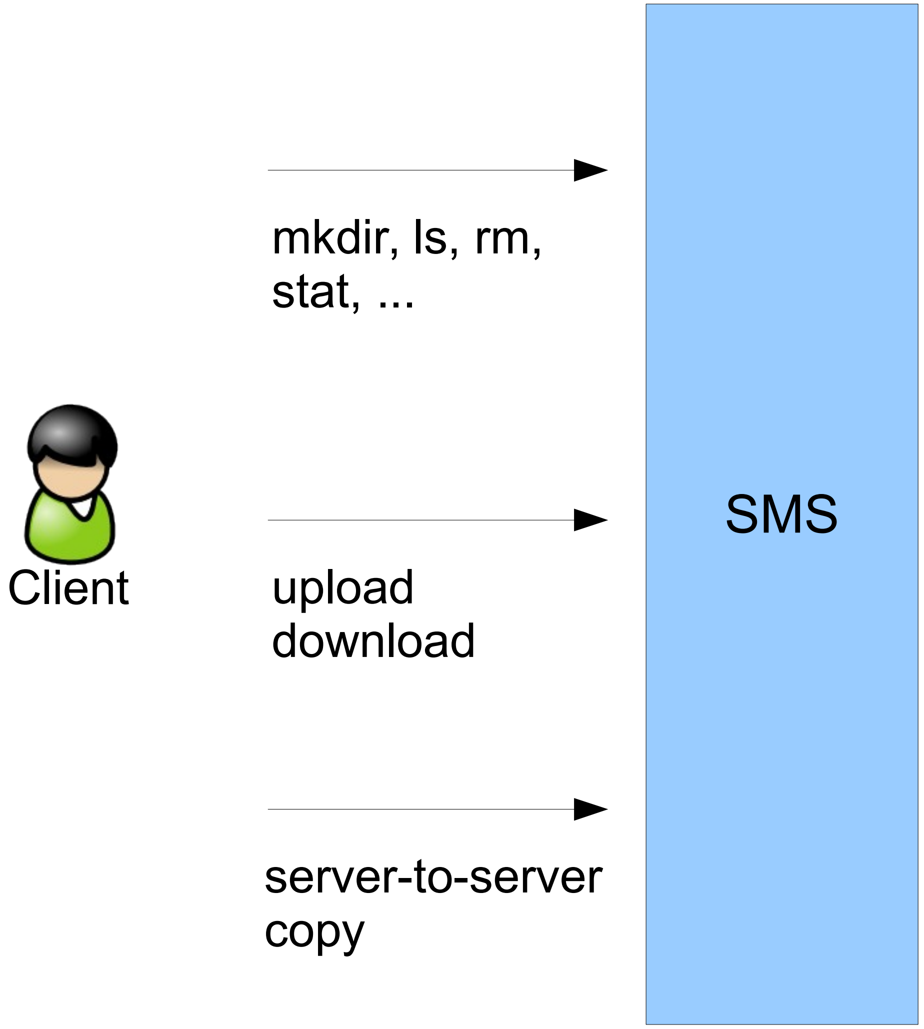
- Access to resource manager and file system via TargetSystemInterface (TSI) daemon installed on the cluster login node(s)

Factory services: virtualisation support



- Can add new types of TargetSystems, e.g. to set up a virtual image during its initialisation phase
- Provide access to the newly started virtual machine

Storage Management Service

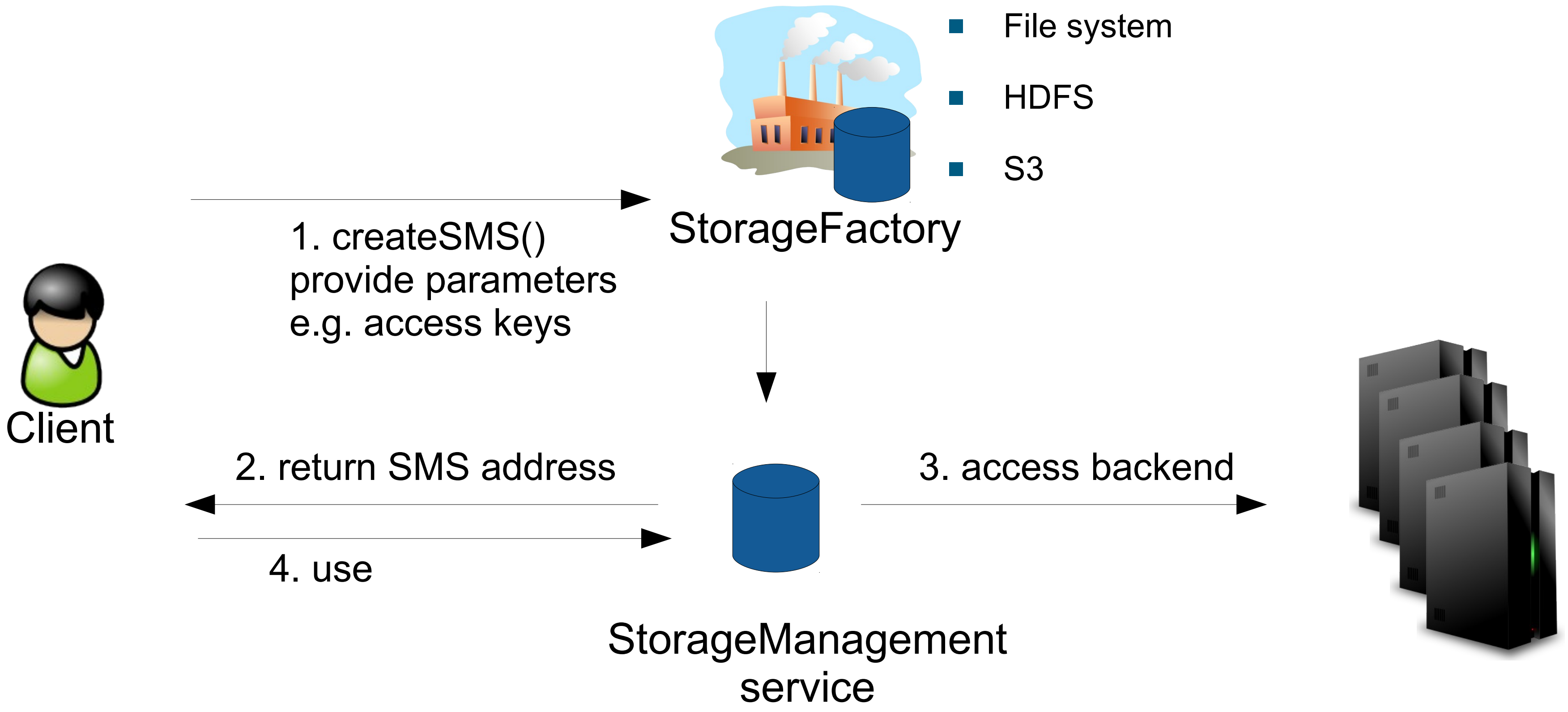


- File system
- Apache HDFS
- S3 (under test)
- iRODS (prototype)

Storage Management Service

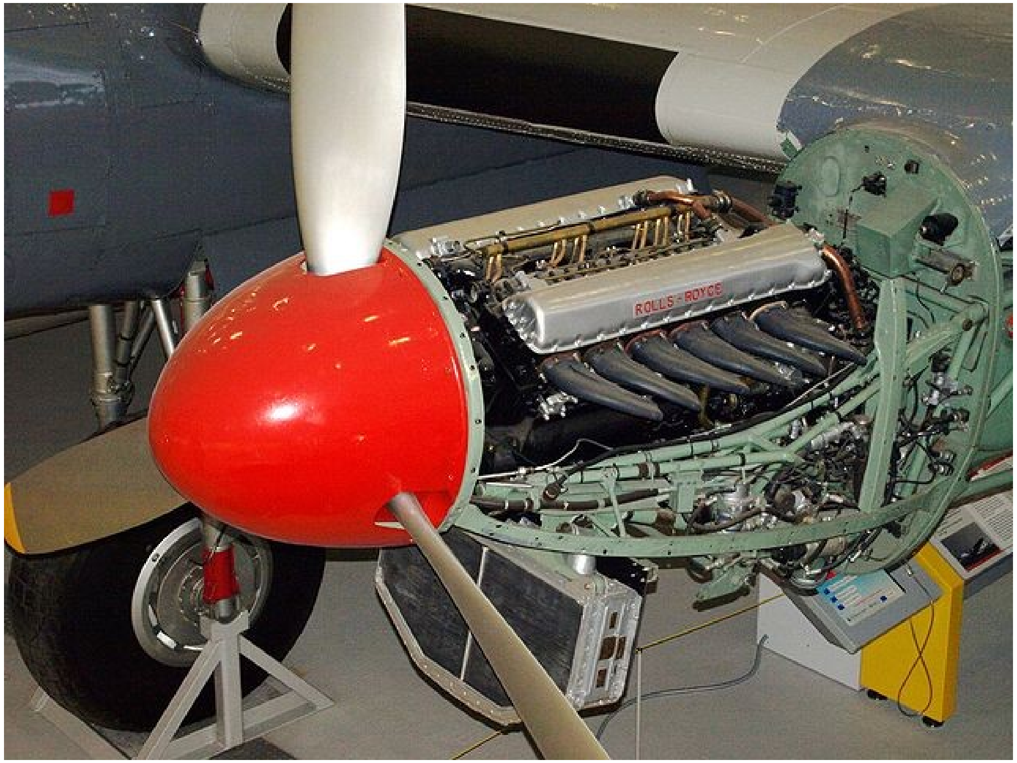
- Initiate file transfers
  - Multi-protocol support
- Metadata management
  - Schema-free, key-value
  - Indexed via Lucene, searchable
- Rule-based data processing
  - New files automatically trigger actions
  - e.g. metadata extraction, compression, etc

Factory services: virtualisation support



- Different types of storage backends can be supported
- User can select and provide required parameters

UNICORE : under the hood



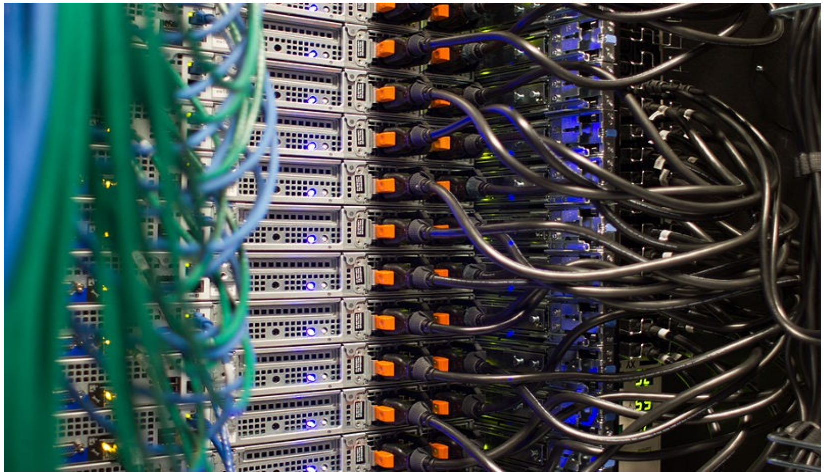
UNICORE Services Environment

- Implemented in Java

- Based on Apache CXF (http://cxf.apache.org/)
  - Very mature and up-to-date services stack. Current version is 2.7.x, 3.x coming soon
  - SOAP web services
  - REST via JAX-RS

- Numerous other open source libraries

Federated access: security is the key

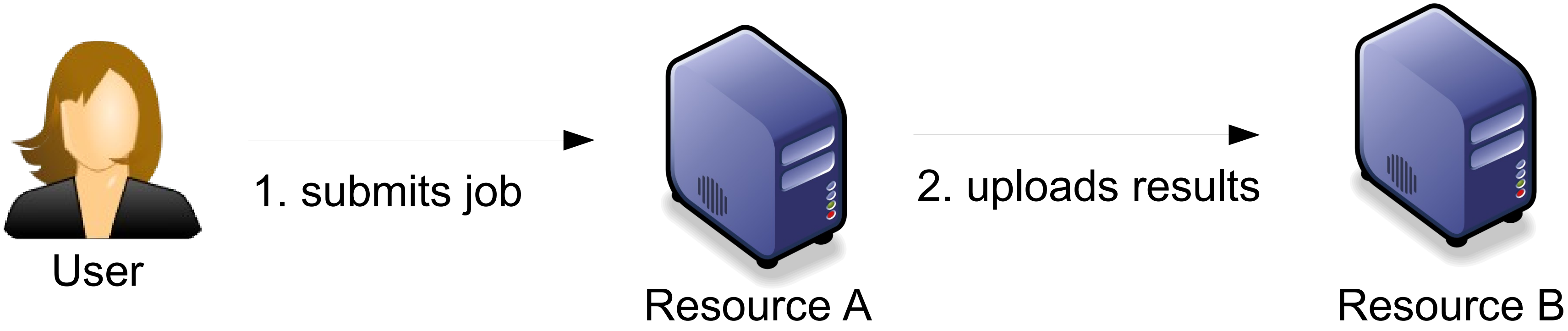


UNICORE – Basic security flow

- **User invokes a service**, i.e. makes a web service call to a UNICORE service
- **Authentication**: who is the user?
  - Results in the user's X.500 DN („CN=..., O=..., C=...“)
- **Assign attributes** to the DN
  - Standard attributes: role, Unix ID, groups, etc.
  - Custom attributes: (e.g. S3 access and secret keys)
- **Authorisation**
  - Add context: e.g. who owns the service?
  - Check local policies (XACML)
- **Allow or deny** the request

Delegation

- Allow Service to work on behalf of the user
- UNICORE solution based on SAML
  - Use chain of signed assertions
  - Trust always delegated to particular server
  - Can be validated and audited



End-user authentication in UNICORE

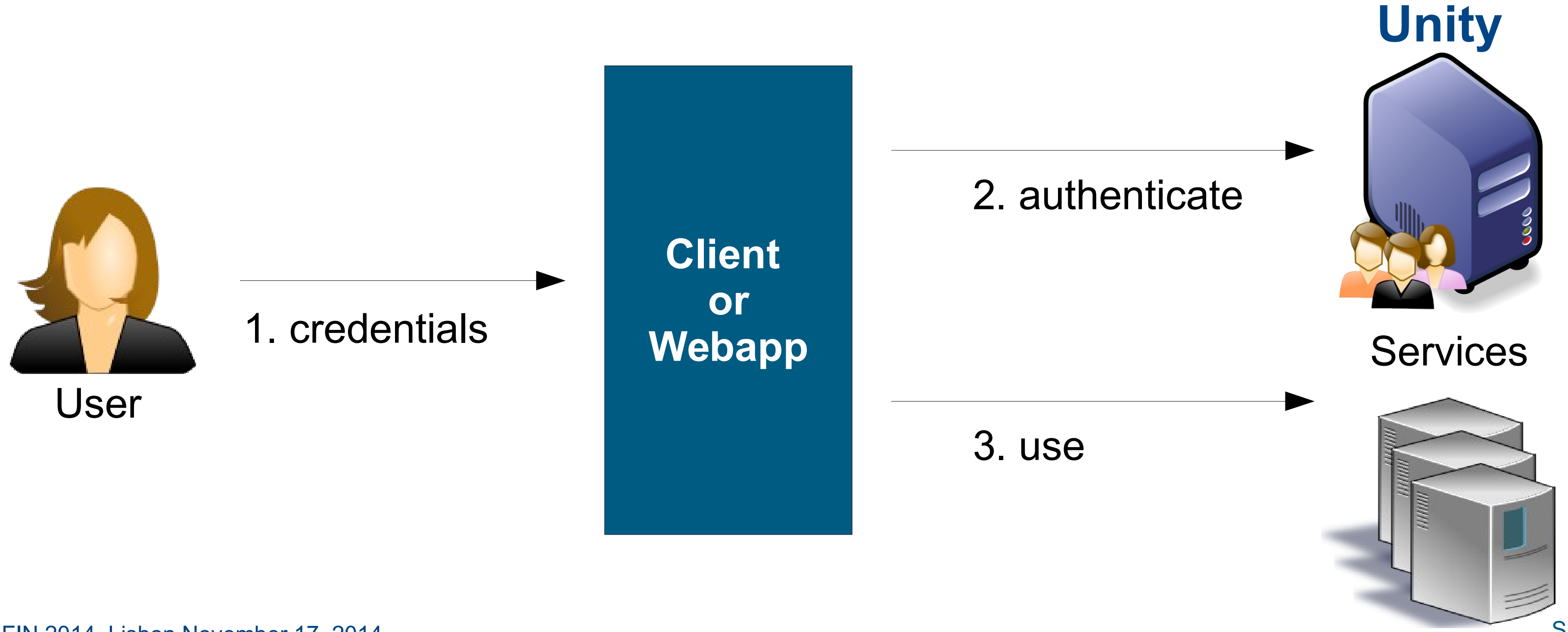
- Pre-UNICORE 7: X.509 client certificates REQUIRED for end-users
- Users tend to hate them
  - All sorts of usage issues
- Lack of understanding leads to lack of security (sending keys via email etc)
- Users understand passwords
  - and it is relatively easy to teach basic security measures

Certificate-less end-user authentication

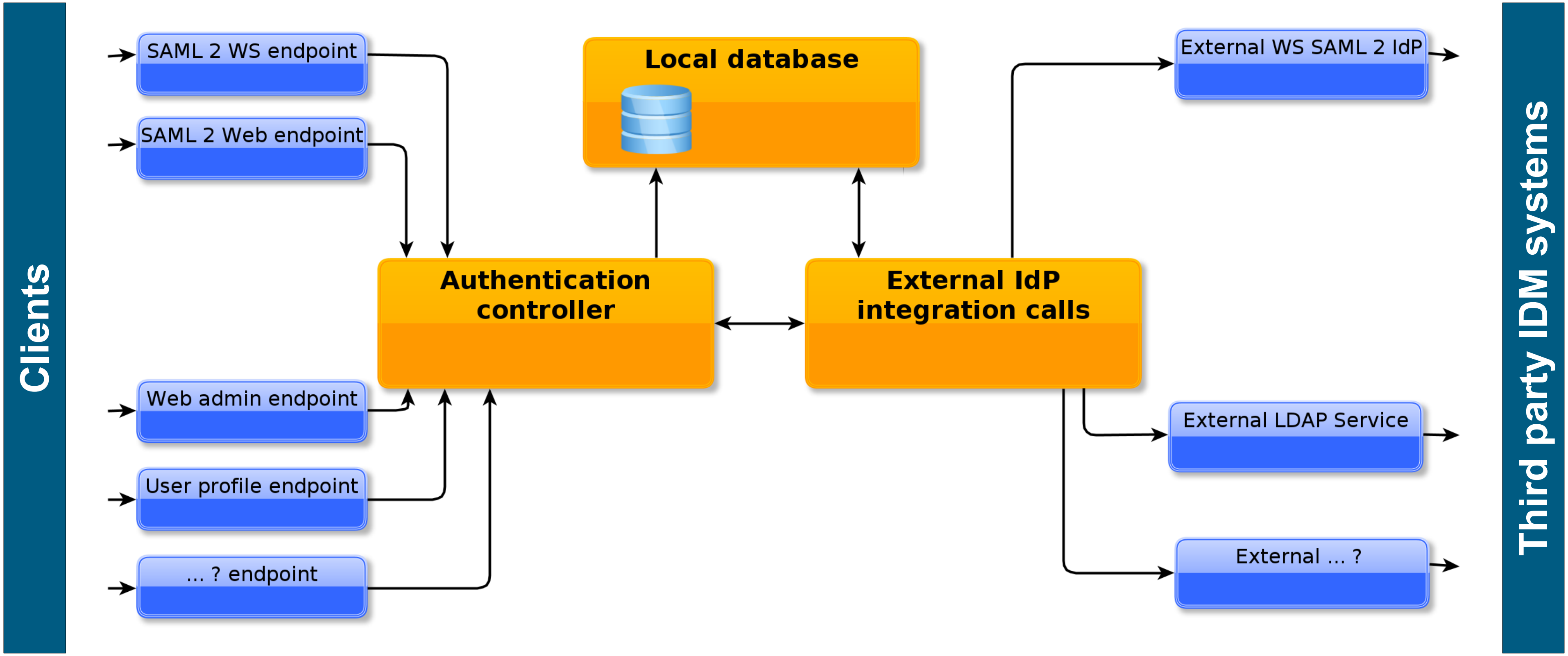
- Goal: **no end-user certificates** (not even short-lived)
- Approach
  - Use *signed SAML assertions*
  - Issued and signed by the trusted server (Identity Provider, IdP)
  - MANY options, e.g. support for existing SAML IdPs , federations like DFN AAI, etc
  - Flexible solution is required
- Implications
  - Client – server TLS is not client-authenticated any more
  - End-user cannot sign anything (no more „non-repudiation“)

Introducing Unity

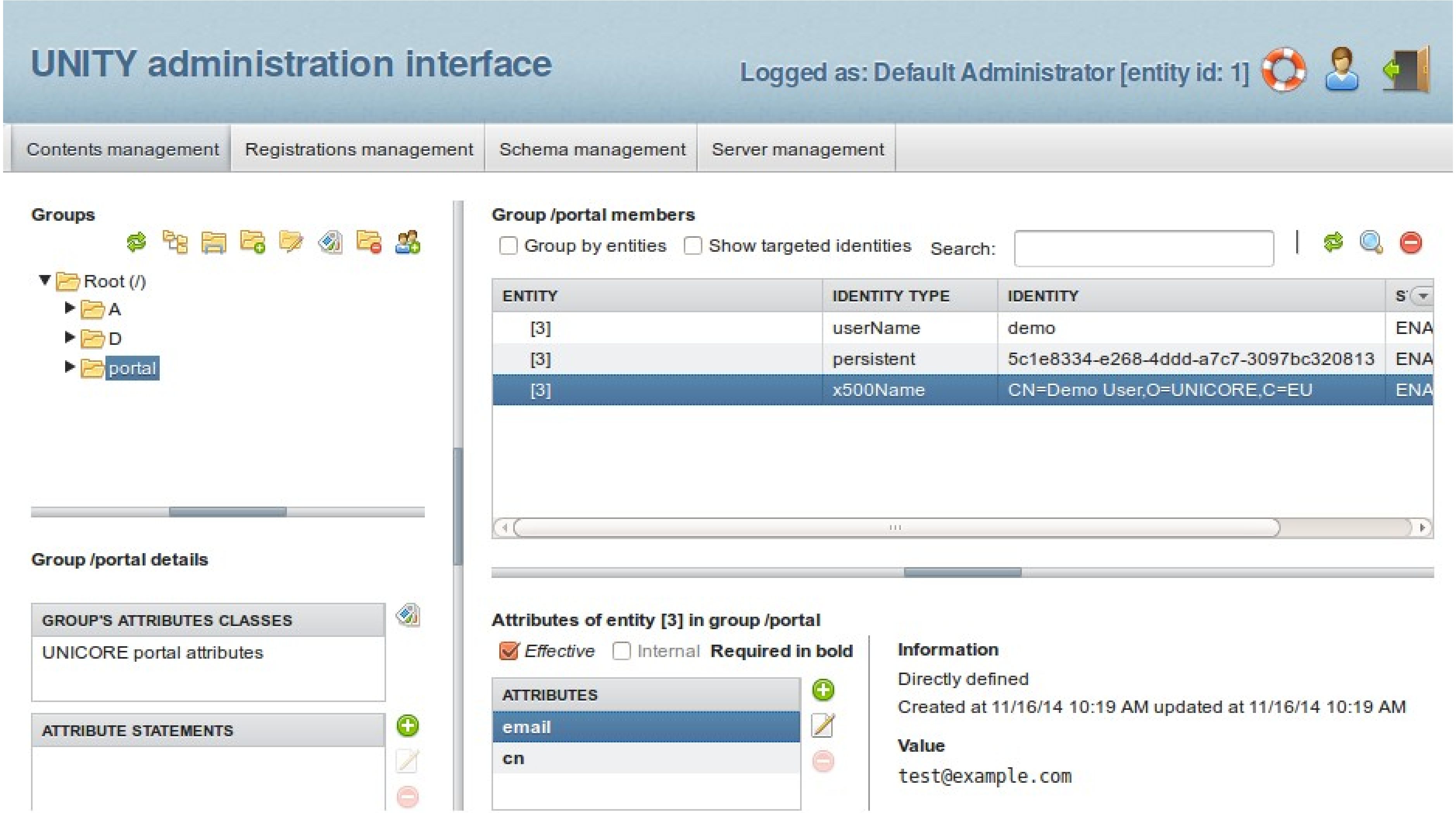
- Complete **Authentication and Identity Management** solution
- Manage users and user attributes, group membership
- Developed by **ICM / Univ. of Warsaw** (PL)
- Separate product: www.unity-idm.eu
- Increasing take-up: e.g. HBP



Unity architecture



Unity admin: managing content



UNITY administration interface

Logged as: Default Administrator [entity id: 1]

Groups

- Root (/)
  - A
  - D
  - portal

Group /portal details

GROUP'S ATTRIBUTES CLASSES
UNICORE portal attributes

ATTRIBUTE STATEMENTS

Group /portal members

□ Group by entities □ Show targeted identities Search:

ENTITY	IDENTITY TYPE	IDENTITY	S
[3]	userName	demo	ENA
[3]	persistent	5c1e8334-e268-4ddd-a7c7-3097bc320813	ENA
[3]	x500Name	CN=Demo User,O=UNICORE,C=EU	ENA

Attributes of entity [3] in group /portal

☑ *Effective* □ Internal **Required in bold**

ATTRIBUTES
email
cn

Information

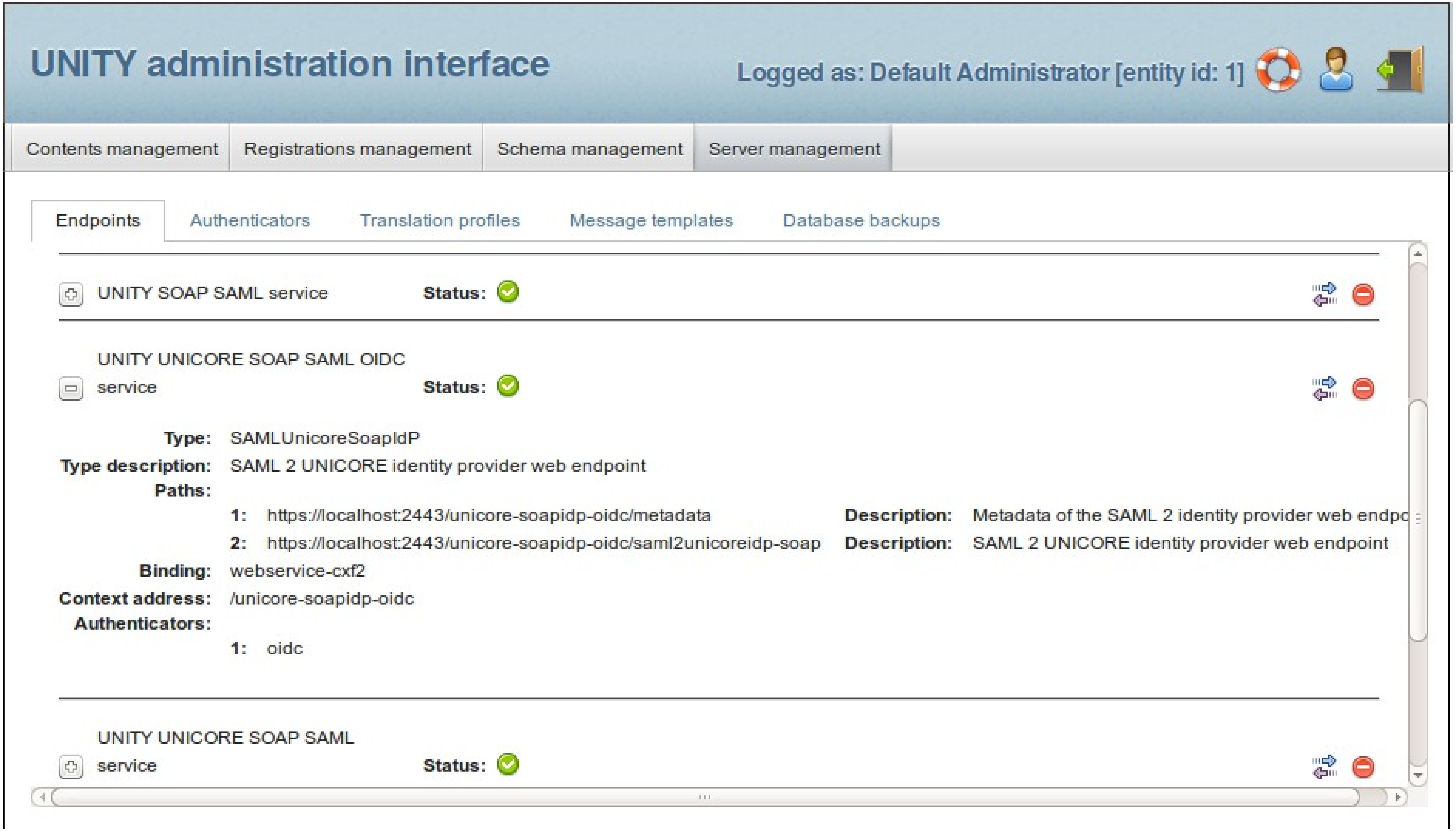
Directly defined

Created at 11/16/14 10:19 AM updated at 11/16/14 10:19 AM

Value

`test@example.com`

Unity admin: managing endpoints



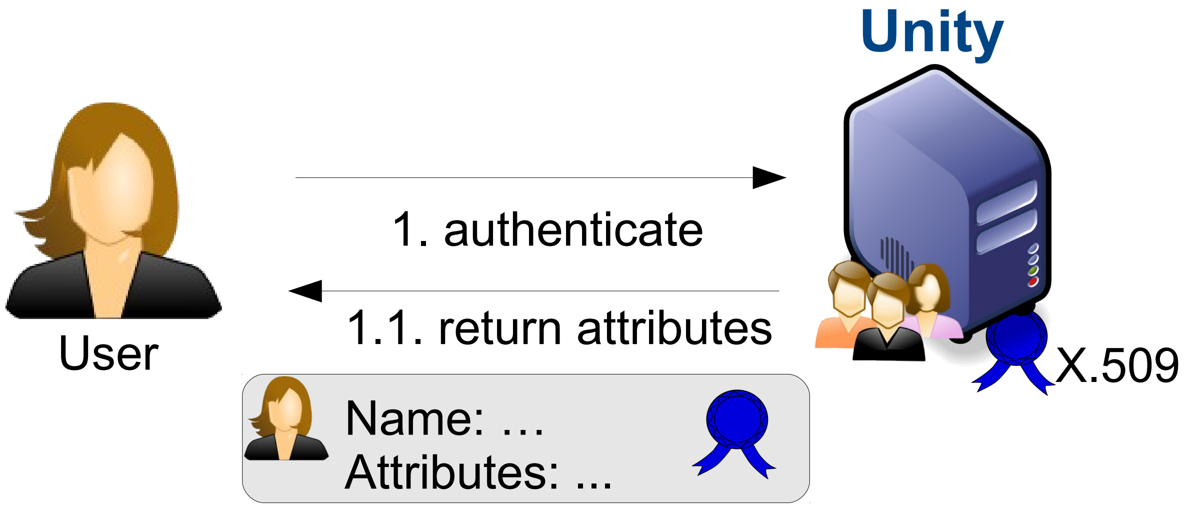
UNITY SOAP SAML service Status: ✓

UNITY UNICORE SOAP SAML OIDC service Status: ✓

Type: SAMLUnicoreSoapIdP
Type description: SAML 2 UNICORE identity provider web endpoint
Paths:
1: https://localhost:2443/unicore-soapidp-oidc/metadata **Description:** Metadata of the SAML 2 identity provider web endpc
2: https://localhost:2443/unicore-soapidp-oidc/saml2unicoreidp-soap **Description:** SAML 2 UNICORE identity provider web endpoint
Binding: webservice-cxf2
Context address: /unicore-soapidp-oidc
Authenticators:
1: oidc

UNITY UNICORE SOAP SAML service Status: ✓

Example: authentication assertion



Unity

User

1. authenticate

1.1. return attributes

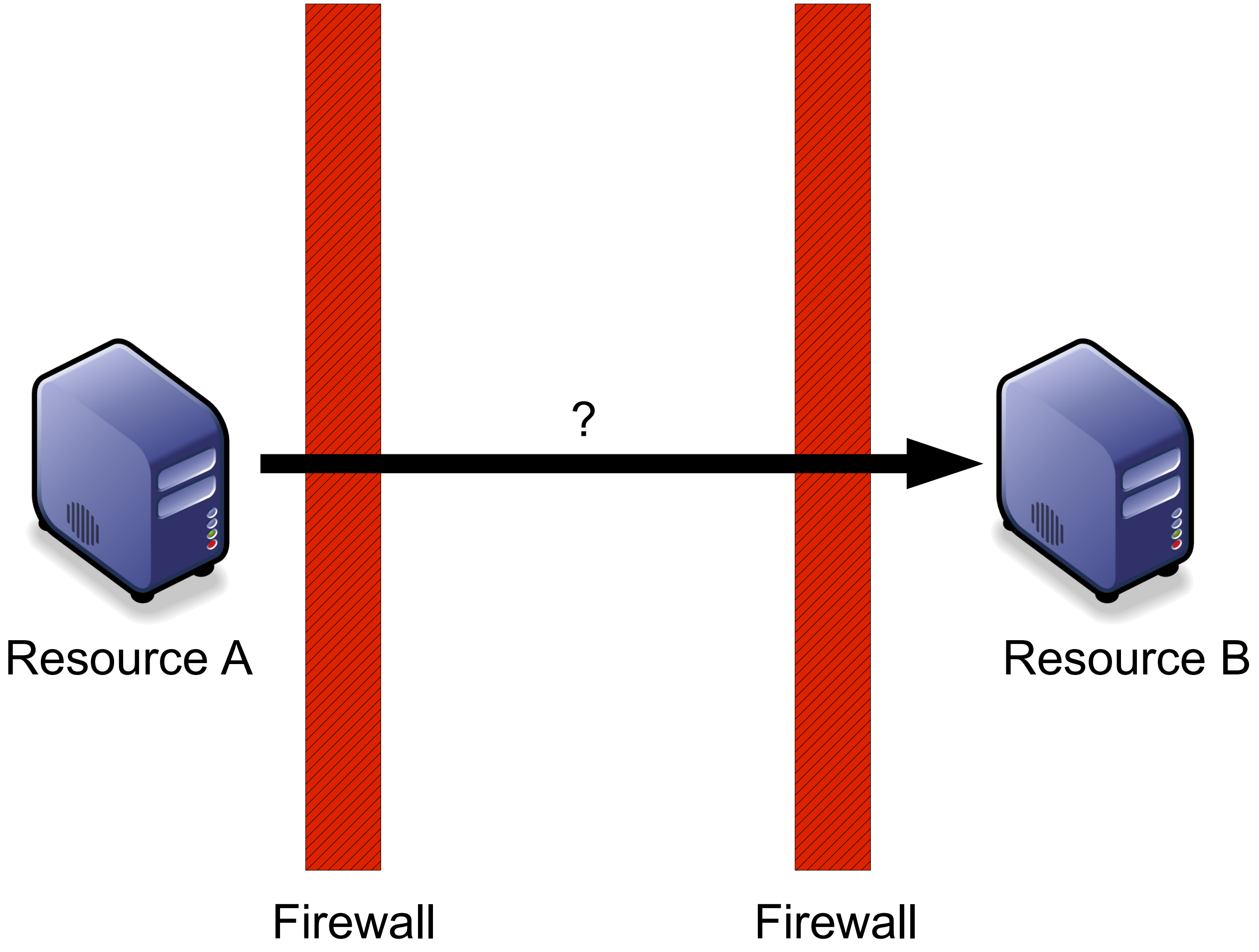
X.509

Name: …
Attributes: ...

```
<urn:Assertion>...
  <dsig:Signature... </dsig:Signature>
  <urn:Subject>
    <urn:NameID
       Format="urn:oasis:names:tc:SAML:1.1:nameid-format:X509SubjectName">CN=Demo User,O=UNICORE,C=EU</urn:NameID>
    <urn:SubjectConfirmation Method="urn:oasis:names:tc:SAML:2.0:cm:sender-vouches">
      <urn:SubjectConfirmationData NotOnOrAfter="2014-11-16T10:30:23.334Z"/>
    </urn:SubjectConfirmation>
  </urn:Subject>
  <urn:AttributeStatement>
    <urn:Attribute Name="cn">
      <urn:AttributeValue>Demo User</urn:AttributeValue>
    </urn:Attribute>
    <urn:Attribute Name="email">
      <urn:AttributeValue>test@example.com</urn:AttributeValue>
    </urn:Attribute>
    <urn:Attribute Name="memberOf">
      <urn:AttributeValue>/portal</urn:AttributeValue>
      <urn:AttributeValue>/</urn:AttributeValue>
    </urn:Attribute>
  </urn:AttributeStatement>
</urn:Assertion>
```

UFTP – high performance data transfer

Requirement: efficient data transfer through firewalls



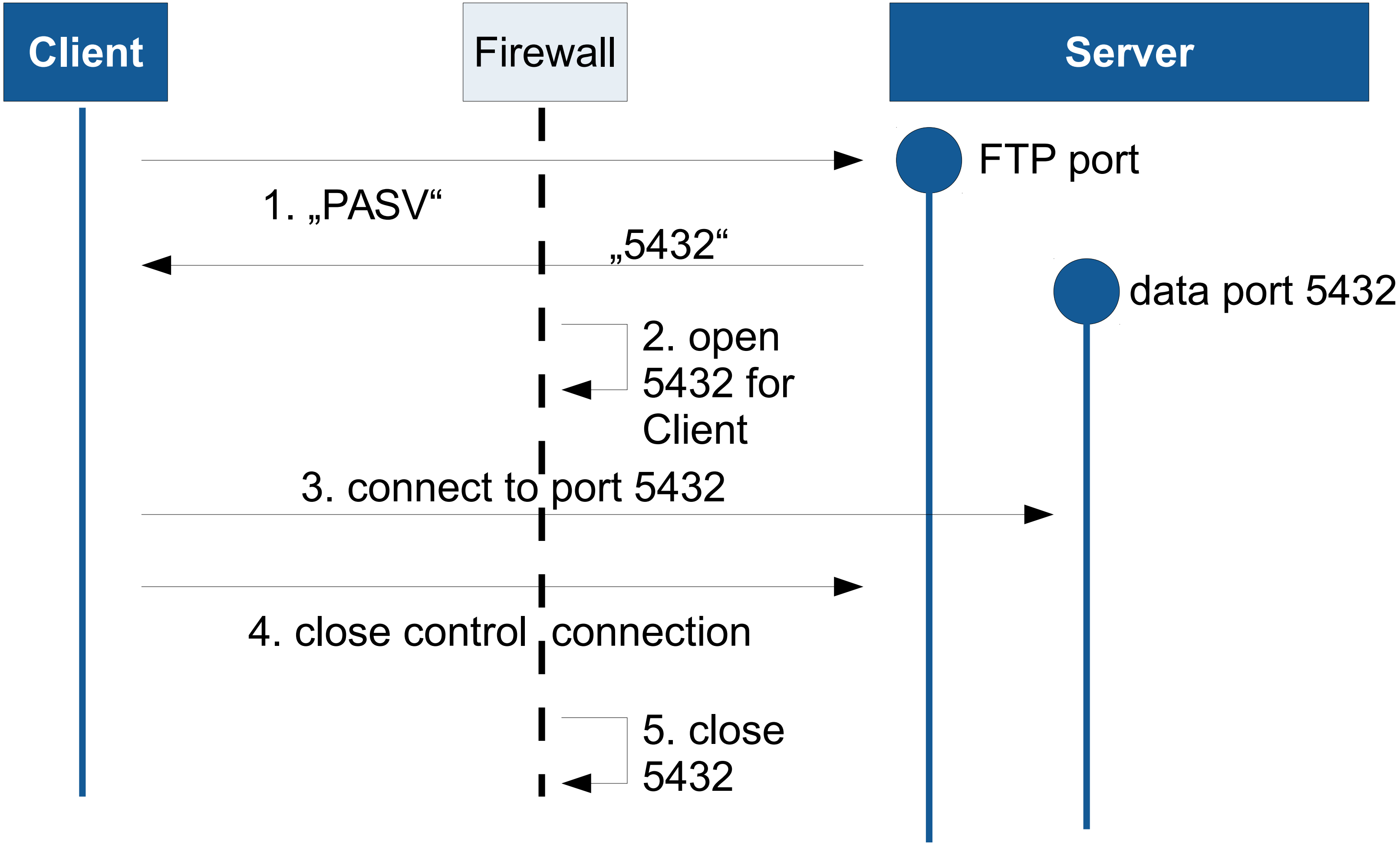
Common data transfer issues

- Firewall
  - Direct connections from the outside to the login node(s) are usually not allowed
  - Statically opening ports (or worse, port ranges) is a security risk

  → *need dynamic port opening technique*

- User management
  - Authentication and authorization
  - User ID / group IDs mapping
  - External / anonymous users

Solving the firewall issue: using passive FTP to open ports



UFTP = passive FTP plus separate AuthN

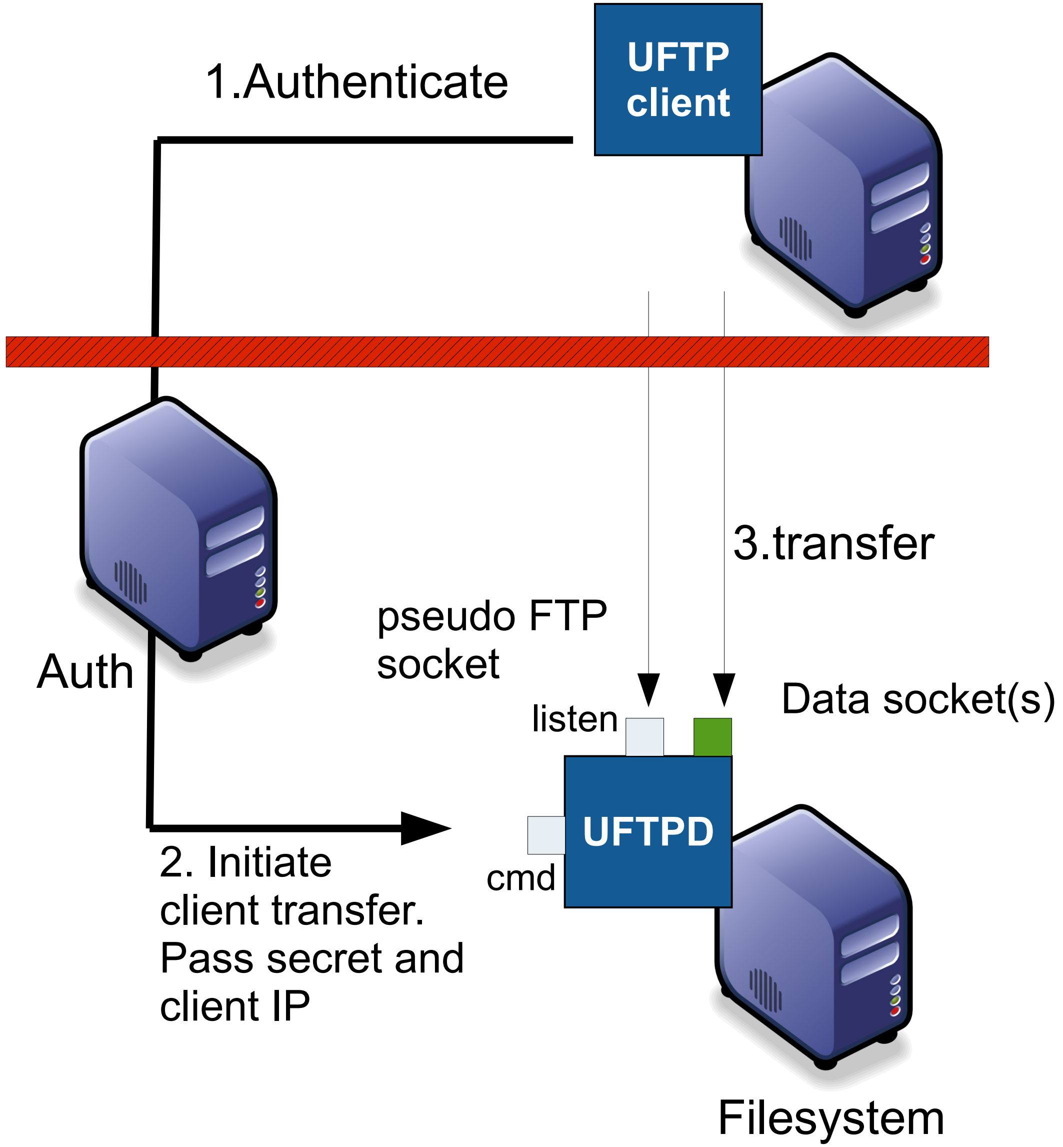
- FTP by itself is insecure:
  - Users log in using username/password

- UFTP adds a **secure control channel** which is used for additional security measures:
  - Authenticate clients
  - Map user ID / group IDs
  - Initiate data transfers

- Requires an secured „command port“ in addition to the open FTP port

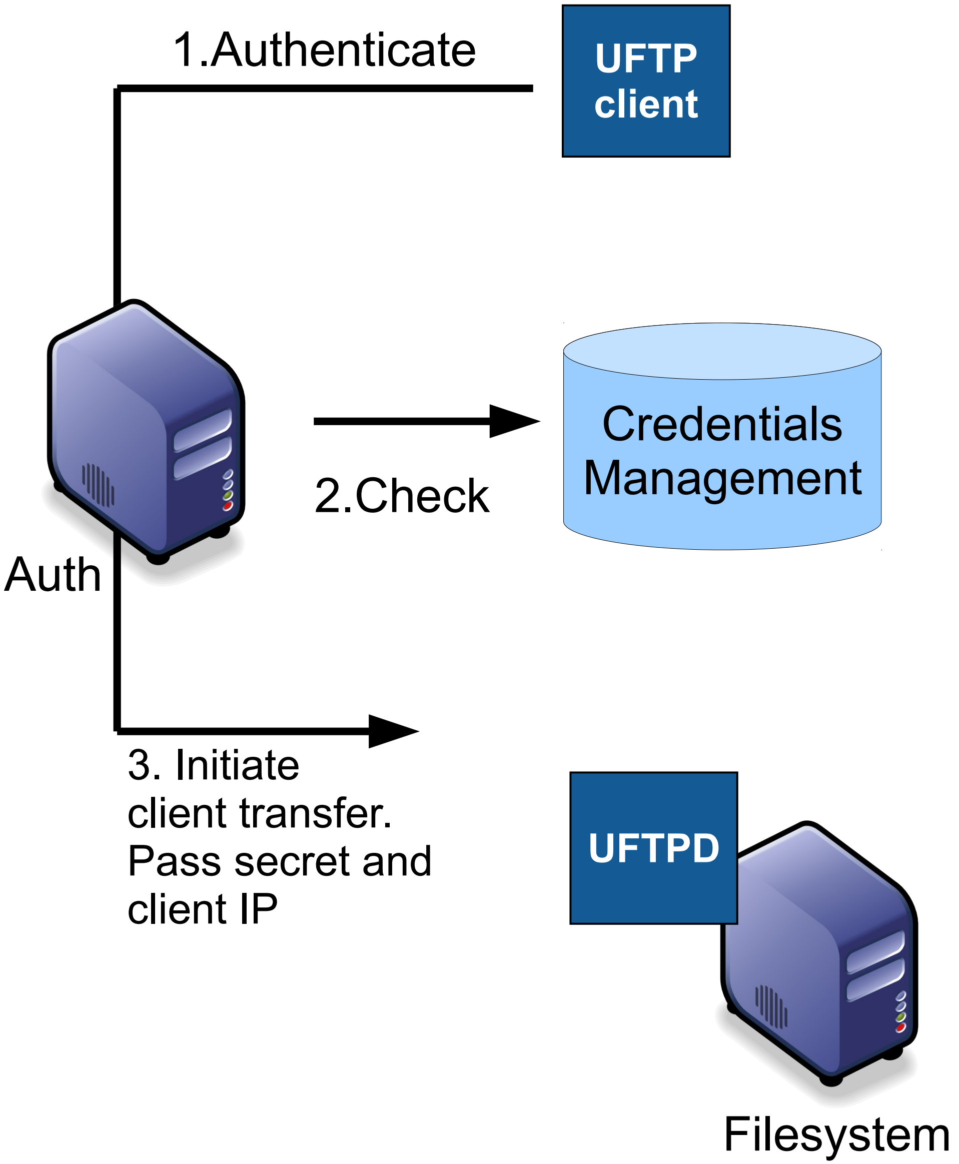
UFTP components

- UFTPD server
  - Pseudo-FTP port (open in firewall) for clients
  - Local command port (SSL protected) used by Auth server
  - Run as root w/ setuid
- UFTP client
  - Authenticate
  - Connect to UFTPD
  - Send/receive data
- Auth server
  - Client authentication
  - User ID mapping



Standalone „Auth server“

- Authentication
  - Password check
  - sshkey check
  - Unity is supported
- Attribute mapping
  - uid, gid
  - QoS e.g. rate limit
- RESTful service



Standalone UFTP Client

UFTP client

- Authentication
  - Username/password (HTTP basic auth)
  - sshkey incl. support for ssh-agent
- Commands
  - ls – list remote files
  - cp – copy file(s)
    - supports reading/writing parts of files (byte ranges)
  - sync – synchronize single remote/local files
- Requirements: Java 7
- Available as tgz archive

UFTP features

- Fast file transfer library similar to FTP
- Firewall friendly and secure
- Optional encryption and/or compression
- Multiple TCP streams per connection

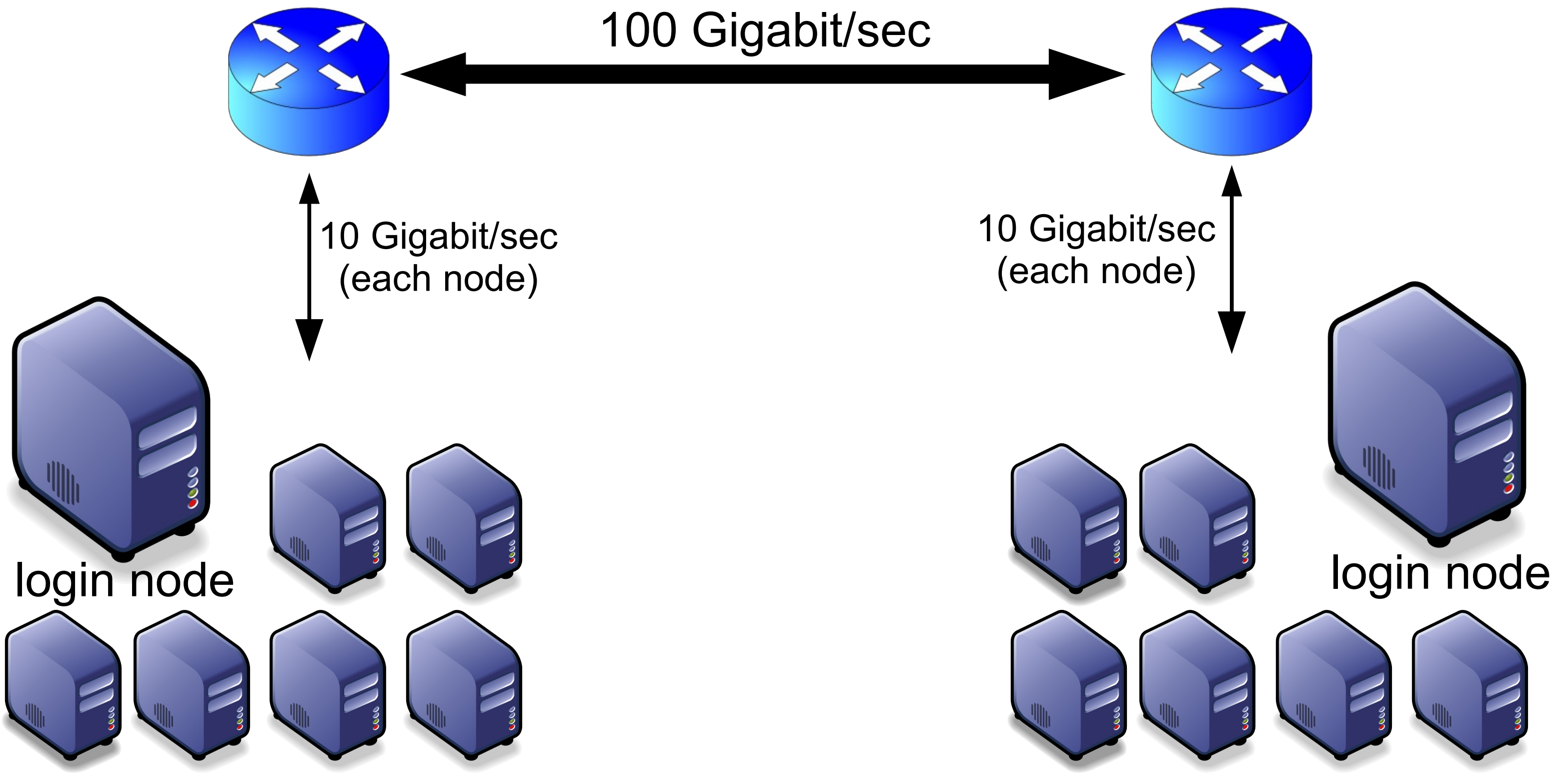
- Fully integrated into UNICORE for data staging and client/server data movement
- Standalone client is available
- Flexible integration options (portals, …) or separate authentication server

- Implemented in Java, available as tgz, rpm, deb

UFTP - Some applications and use cases

- File transfer and data staging in UNICORE
  - Built into standard UNICORE clients
  - Java applet for the UNICORE web portal
- Standalone use (client plus separate AuthN server)
  - Secure, high-performance data upload/download
- Integrate UFTP functionality into web applications
- Planned master thesis: Data access and sharing at JSC (UFTP+AAI+HPC storage cluster)

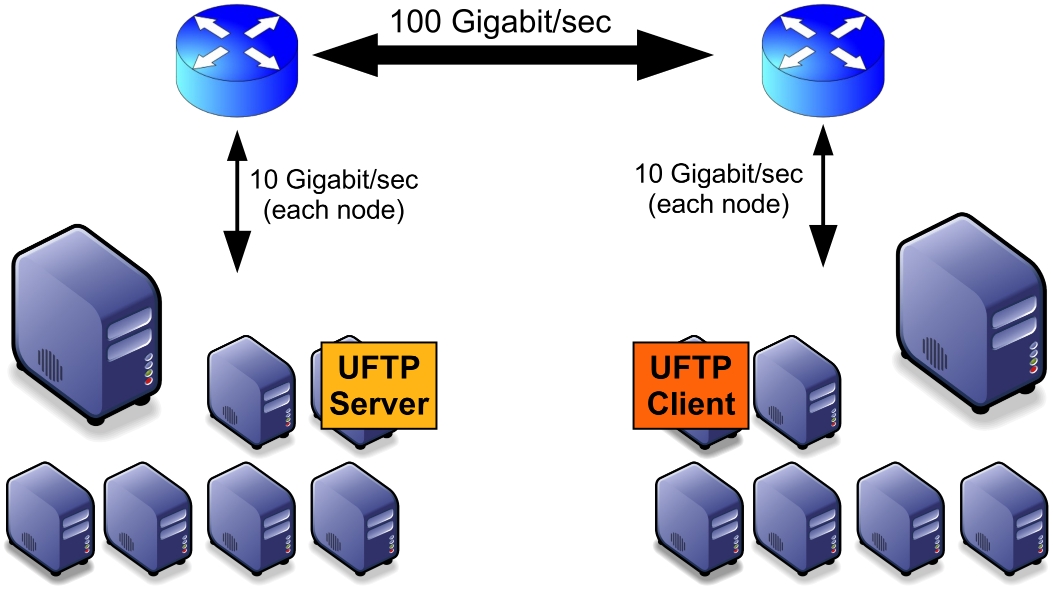
2012: Testing UFTP on a 100 GBit/s testbed TU Dresden – TU Freiberg

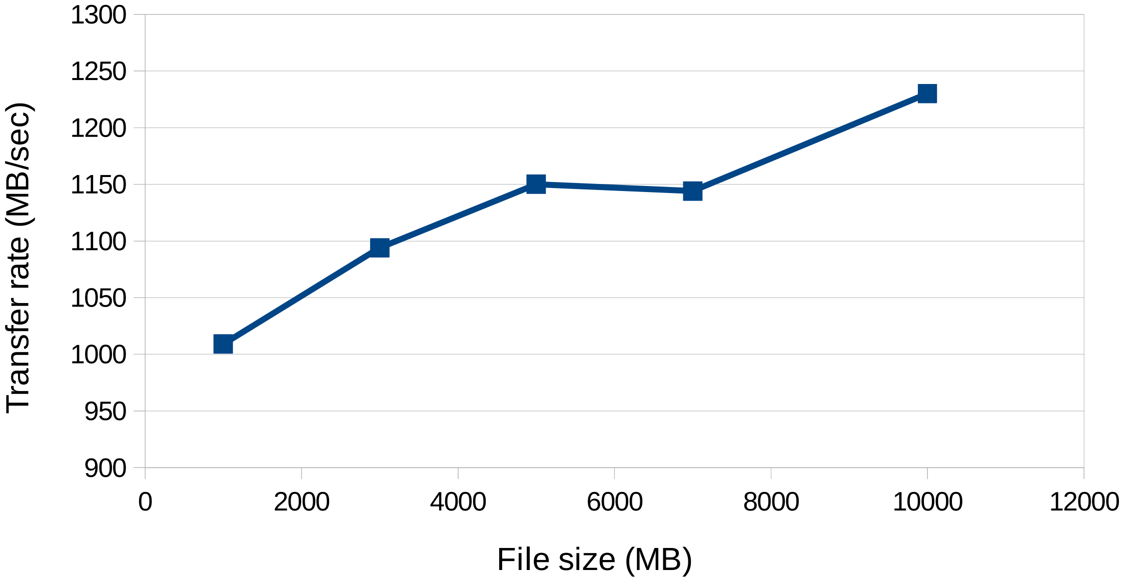


- Up to 10 GBit/sec per cluster node
- Up to 100 GBit/sec aggregated transfer rate

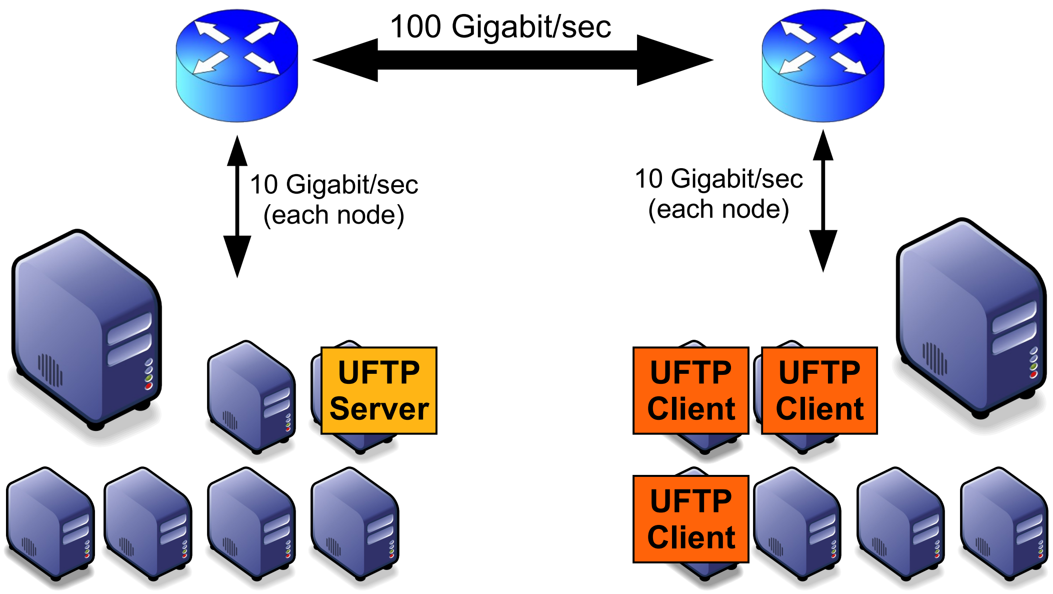
Single client, single server

- Up to 1.2GB/sec
- 98% of line rate

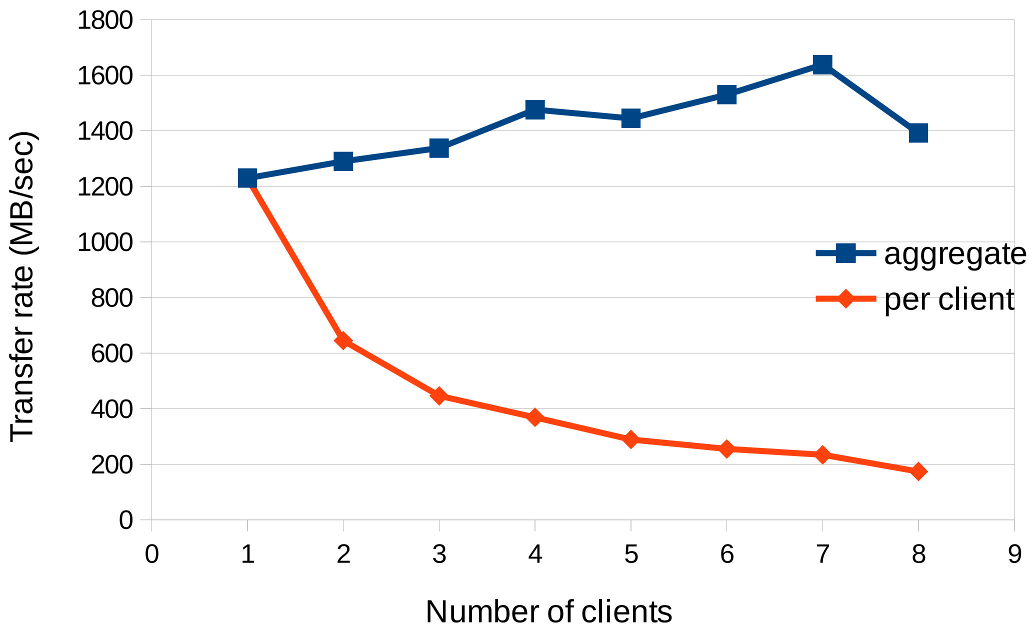




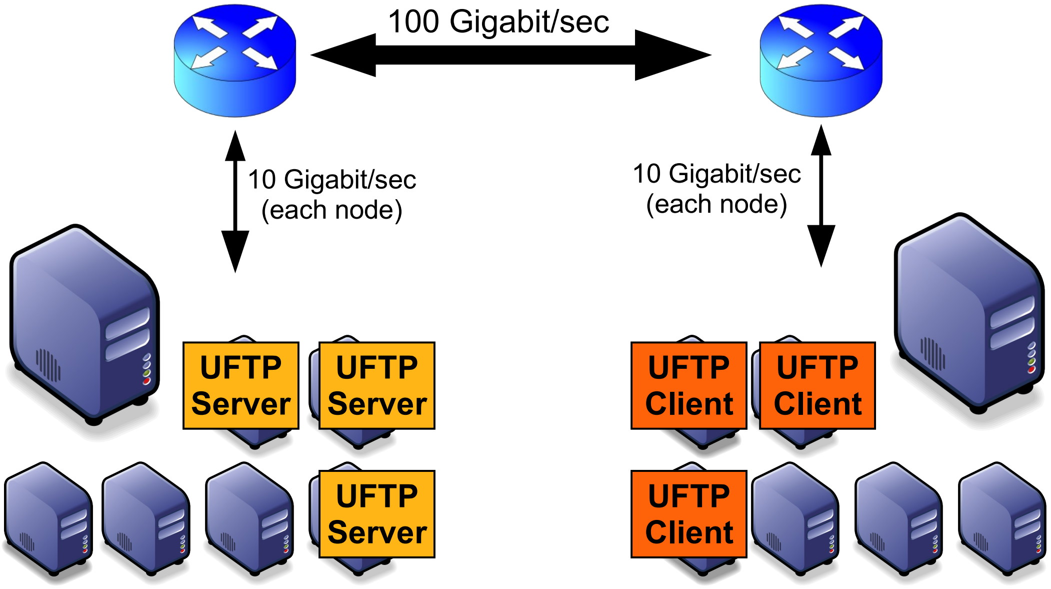
Multiple clients, single server



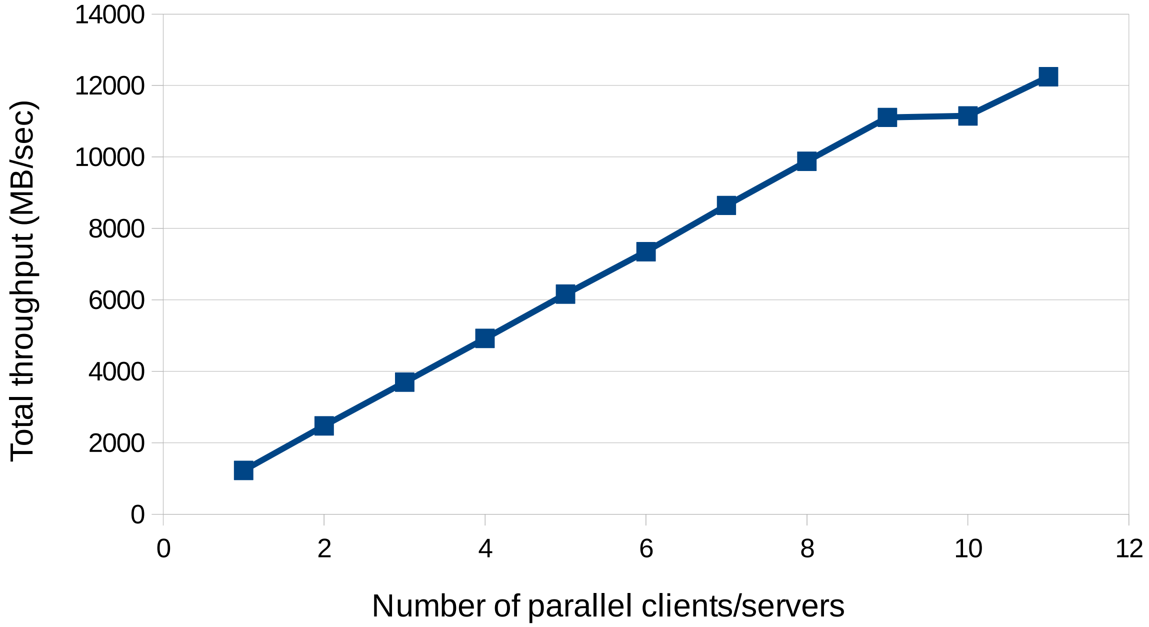
- Up to 8 clients
- (roughly!) parallel transfers (50GB each)



Multiple client/server pairs



- Up to 11 (roughly!) parallel transfers (50GB each)
- 12 GB/sec
- 98% of line rate

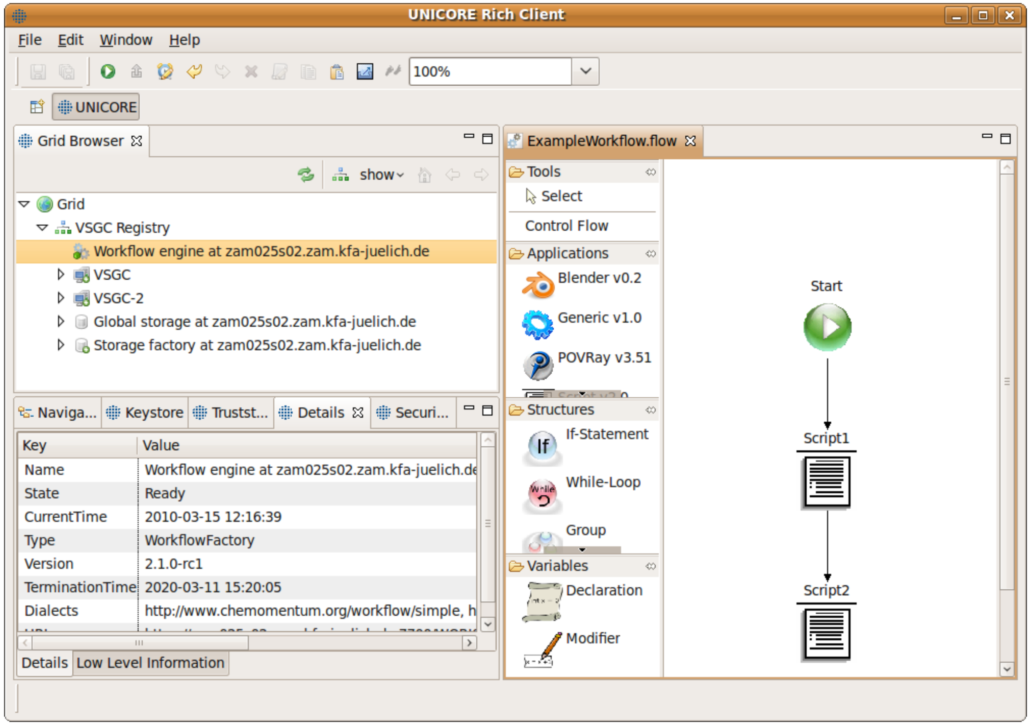


UNICORE Clients

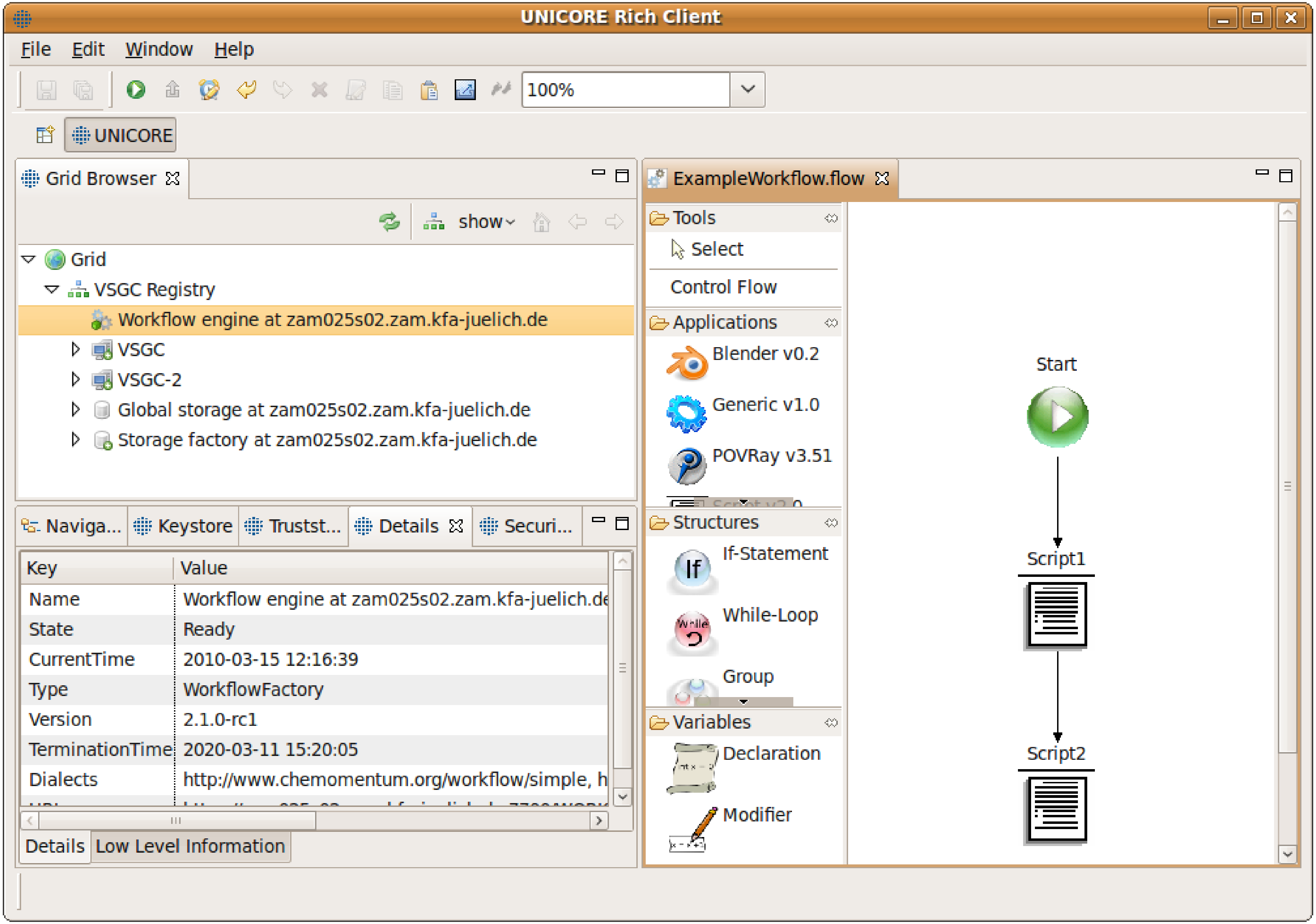
- „Rich client“ based on Eclipse
- Commandline client
- Web portal via Browser
- APIs
  - Java
  - RESTful (work in progress)

Rich client

- Building, submitting and monitoring jobs and workflows
- Integrated data and storage management
- X.509 and Unity for AuthN
- “Simple view” for novice users
- Based on the Eclipse framework
- Extensibility through plug-ins
- Installation/update mechanism for plug-ins and Application GUIs



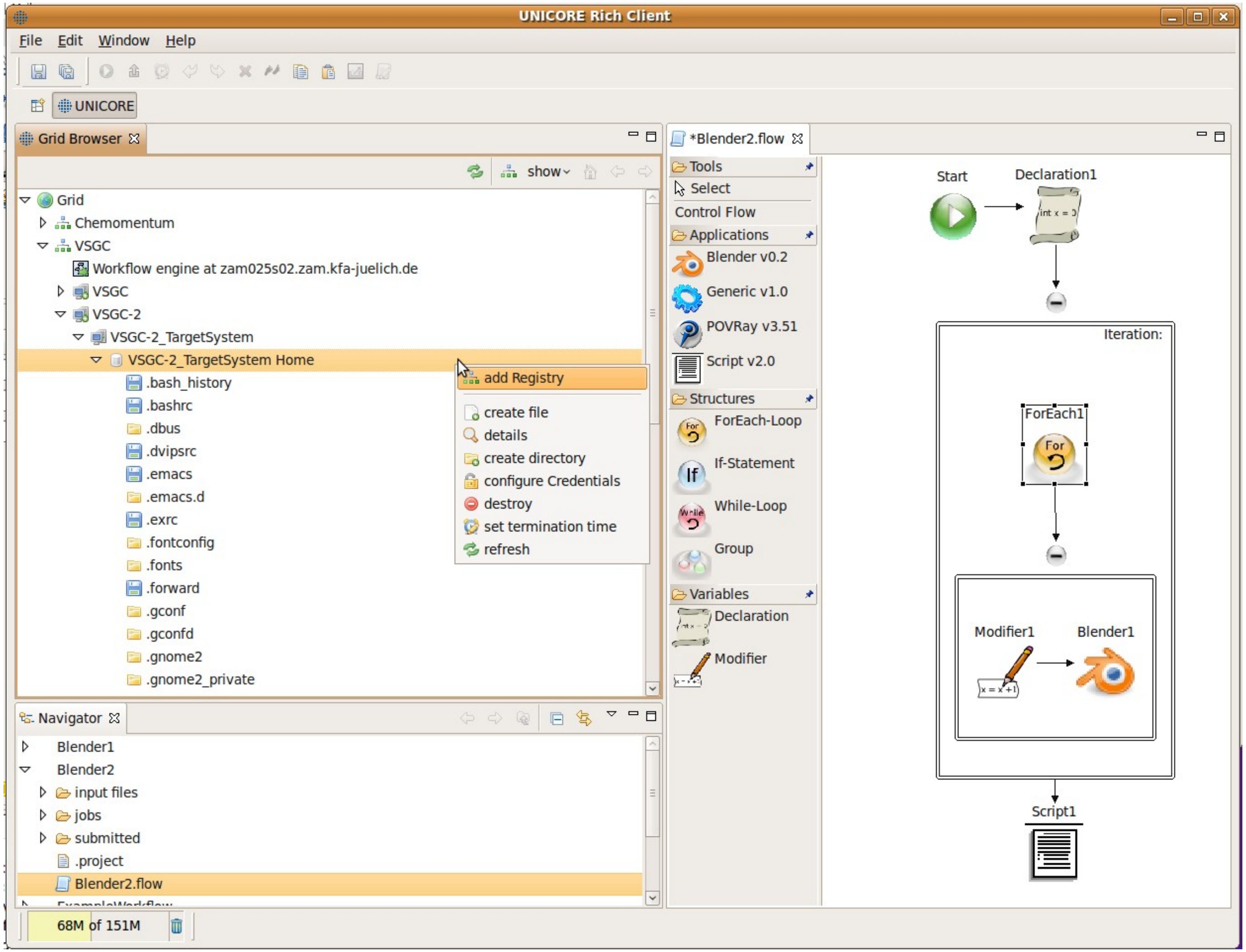
Key	Value
Name	Workflow engine at zam025s02.zam.kfa-juelich.de
State	Ready
CurrentTime	2010-03-15 12:16:39
Type	WorkflowFactory
Version	2.1.0-rc1
TerminationTime	2020-03-11 15:20:05
Dialects	http://www.chemomentum.org/workflow/simple, h



Key	Value
Name	Workflow engine at zam025s02.zam.kfa-juelich.de
State	Ready
CurrentTime	2010-03-15 12:16:39
Type	WorkflowFactory
Version	2.1.0-rc1
TerminationTime	2020-03-11 15:20:05
Dialects	http://www.chemomentum.org/workflow/simple, h

Integrated storage management in the UNICORE Rich client Grid browser

- Create files
- Drag and drop from/to desktop environment
- Copy and paste
- Remote file editing

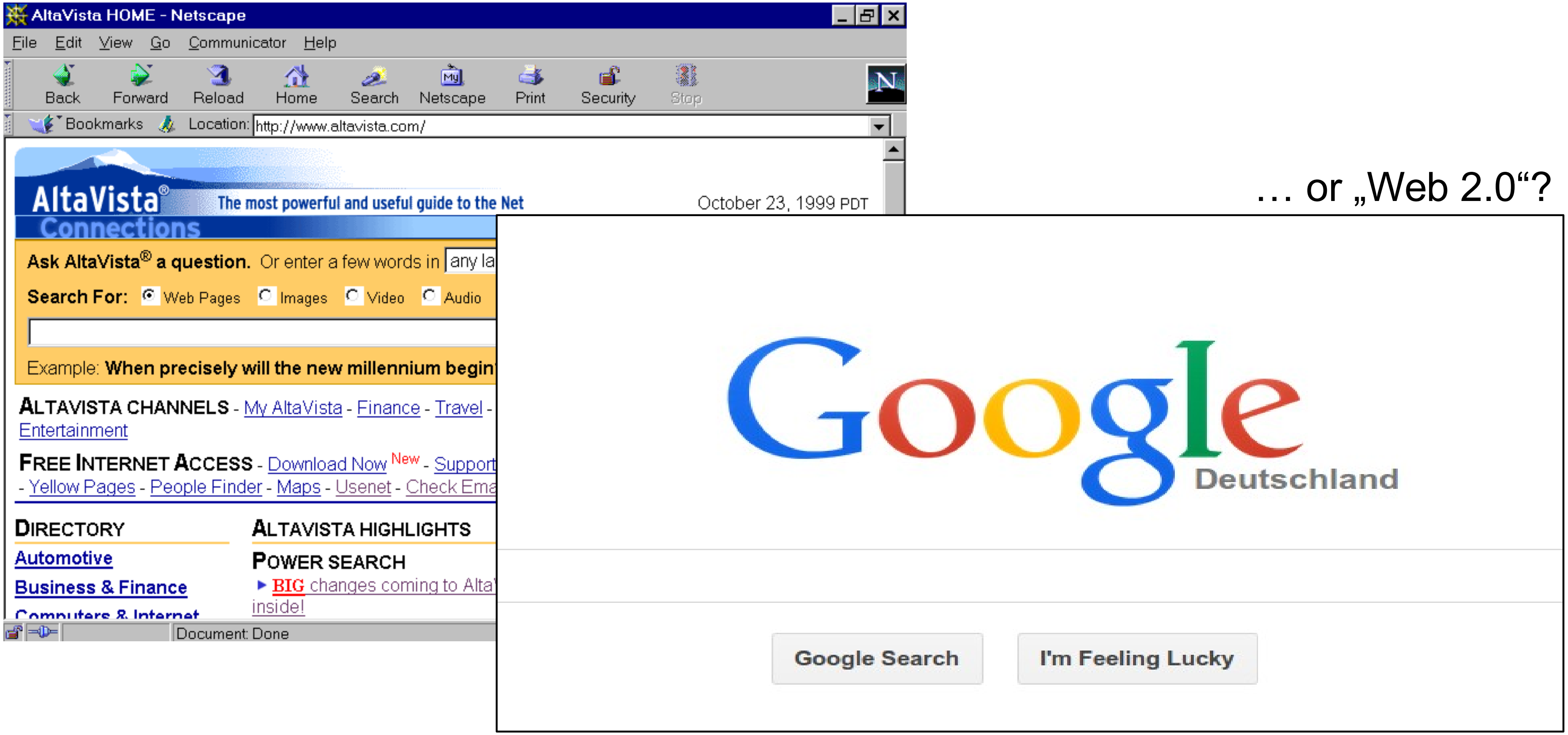


Portal / Web client

- What is a „portal“ anyway?

Back to the 1990s?

… or „Web 2.0“?


AltaVista HOME - Netscape
File Edit View Go Communicator Help
Back Forward Reload Home Search Netscape Print Security Stop
Bookmarks Location: http://www.altavista.com/
AltaVista® Connections
Search For: ● Web Pages ○ Images ○ Video ○ Audio
Document: Done
Google Deutschland
Google Search
I'm Feeling Lucky

The most powerful and useful guide to the Net

October 23, 1999 PDT

Ask AltaVista® a question. Or enter a few words in any la

Example: When precisely will the new millennium begin

ALTAVISTA CHANNELS - My AltaVista - Finance - Travel - Entertainment

FREE INTERNET ACCESS - Download Now New - Support - Yellow Pages - People Finder - Maps - Usenet - Check Ema

DIRECTORY

Automotive

Business & Finance

Computers & Internet

ALTAVISTA HIGHLIGHTS

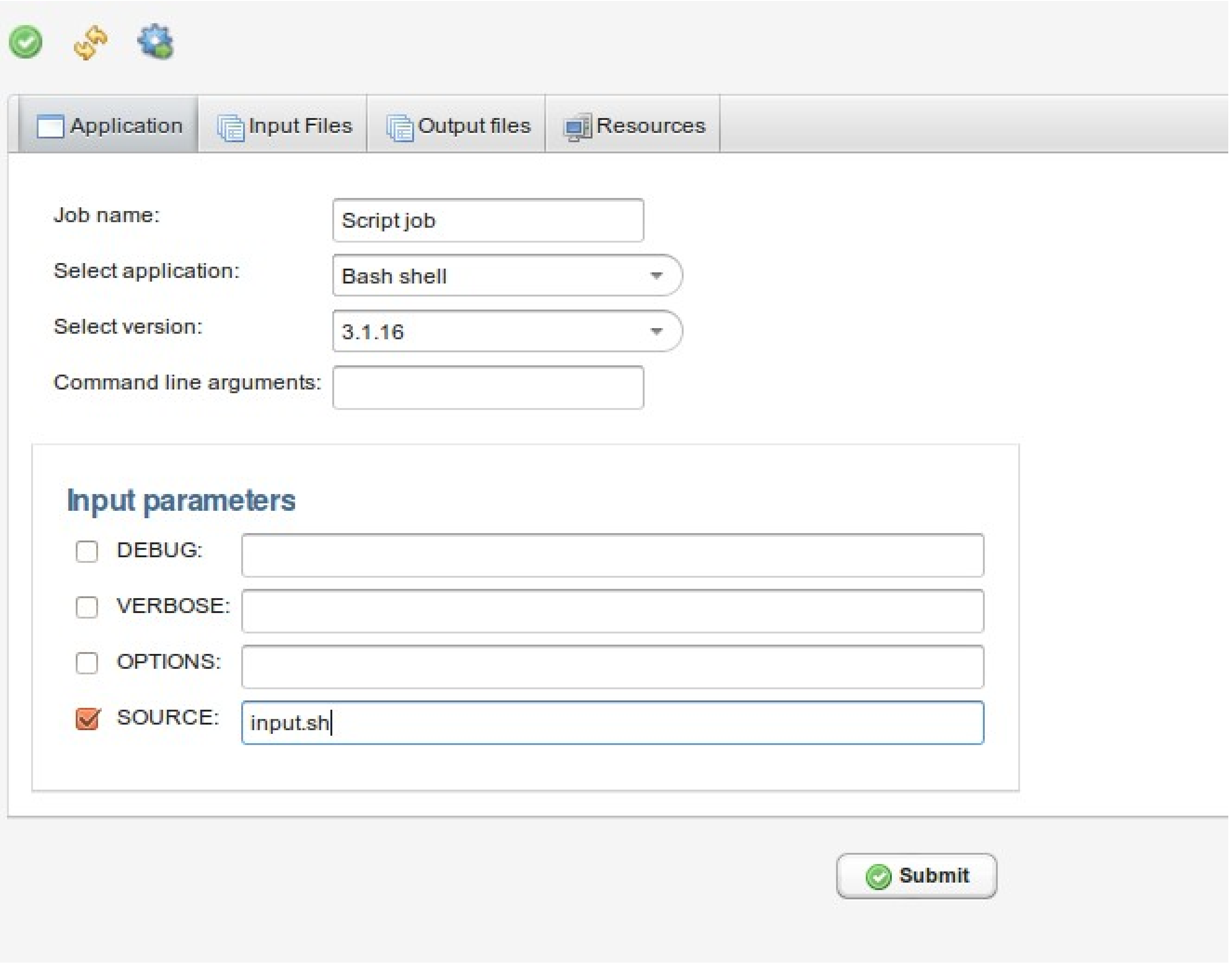
POWER SEARCH

▸ BIG changes coming to Alta inside!

UNICORE Portal

- Aim for a simple, easy-to-use web application
- Flexible authentication and user registration
  - support Unity
- Implementation choices
  - Java-based, VAADIN web framework
  - Use UNICORE Java APIs

UNICORE Portal – Job creation view



Job name: Script job

Select application: Bash shell

Select version: 3.1.16

Command line arguments:

Input parameters

□ DEBUG:

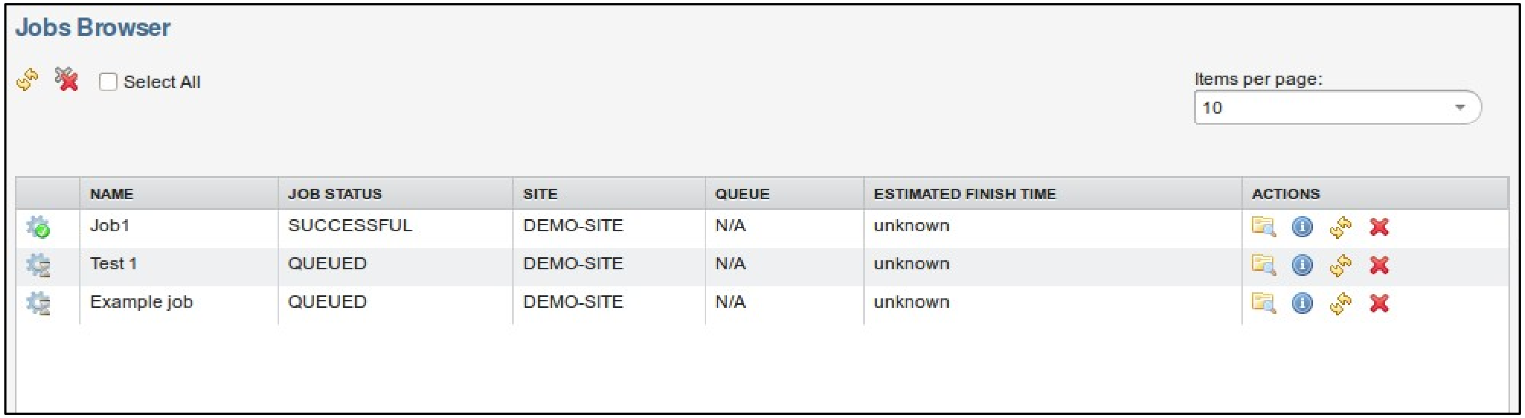
□ VERBOSE:

□ OPTIONS:

☑ SOURCE: input.sh

UNICORE Portal – various

- Several „list“ views, e.g. jobs, sites



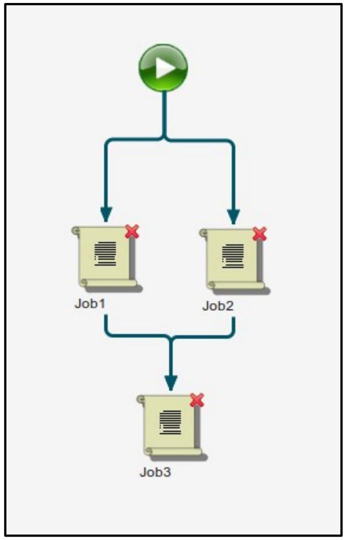
Jobs Browser

□ Select All

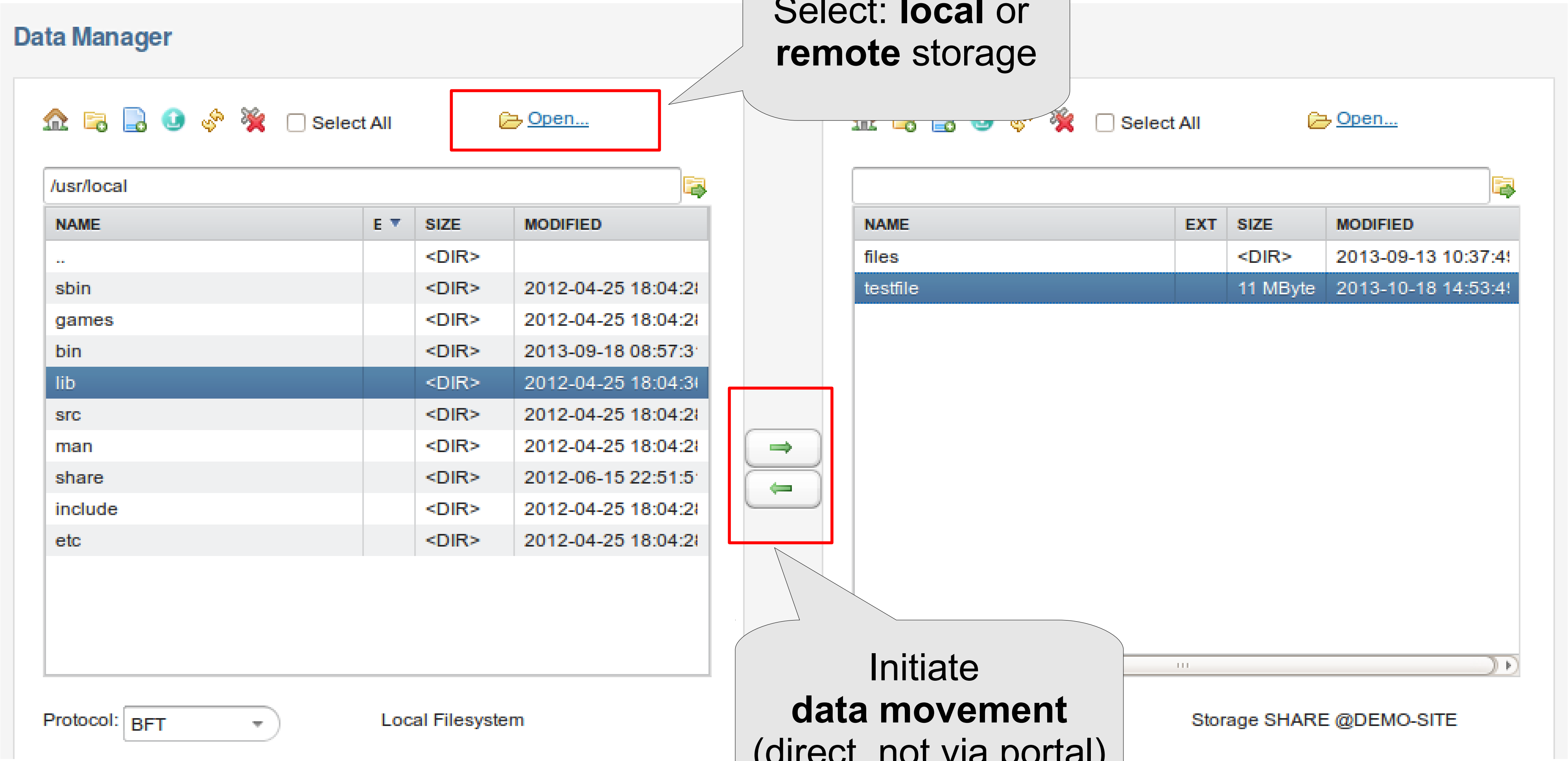
Items per page: 10

	NAME	JOB STATUS	SITE	QUEUE	ESTIMATED FINISH TIME	ACTIONS
	Job1	SUCCESSFUL	DEMO-SITE	N/A	unknown	
	Test 1	QUEUED	DEMO-SITE	N/A	unknown	
	Example job	QUEUED	DEMO-SITE	N/A	unknown	

- Workflow creation
- JavaScript
- Initially only simple graphs



UNICORE Portal: Data manager



Data Manager

/usr/local

NAME	E ▼	SIZE	MODIFIED
..		<DIR>	
sbin		<DIR>	2012-04-25 18:04:2
games		<DIR>	2012-04-25 18:04:2
bin		<DIR>	2013-09-18 08:57:3
lib		<DIR>	2012-04-25 18:04:3
src		<DIR>	2012-04-25 18:04:2
man		<DIR>	2012-04-25 18:04:2
share		<DIR>	2012-06-15 22:51:5
include		<DIR>	2012-04-25 18:04:2
etc		<DIR>	2012-04-25 18:04:2

Protocol: BFT

Local Filesystem

NAME	EXT	SIZE	MODIFIED
files		<DIR>	2013-09-13 10:37:4
testfile		11 MByte	2013-10-18 14:53:4

Storage SHARE @DEMO-SITE

Select: **local** or **remote** storage

Initiate
data movement
(direct, not via portal)

REST APIs

WS(RF) – in use since 2004/2005

- Pros
  - Strongly typed
  - Messages can be validated
  - SOAP: headers/envelope mechanism
  - WS-Security, SAML well established
- Cons
  - CPU intensive (XML processing, XML signatures)
  - Complex interface (look at a typical WSDL!)
  - Only Java and C# can be realistically used on the client side

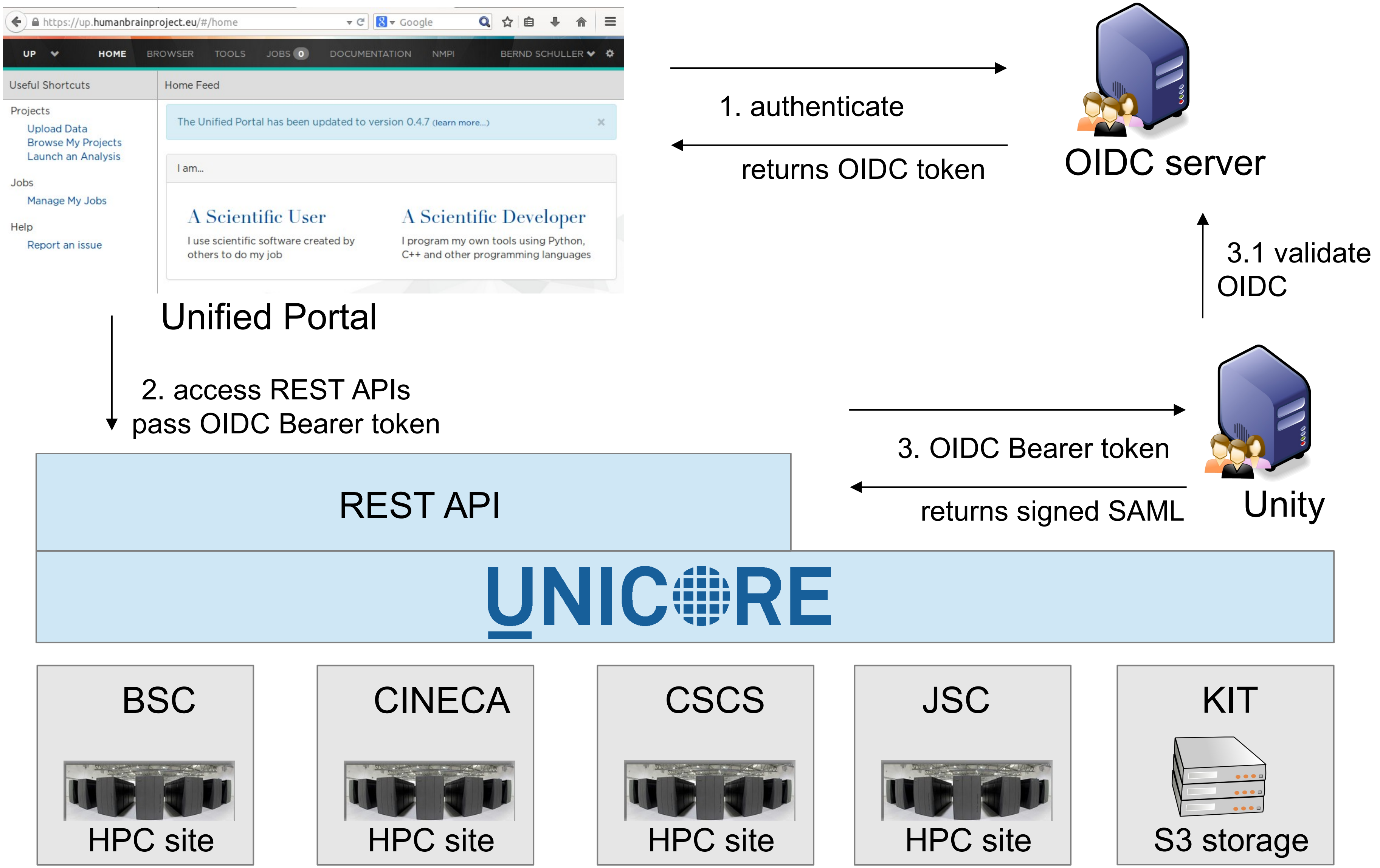
RESTful – pros and cons

- Pros
  - Weakly coupled
  - HTTP benefits (error codes, caching, …)
  - Several authentication options (HTTP basic, OAuth, ...)
  - Multiple message formats and resource representations can be used
    - *JSON, XML, HTML, ...*
  - Clients in all languages (even *curl* or *wget*)
- Cons
  - No standard solution for trust delegation (yet)

RESTful APIs

- Concrete requirements from the
  Human Brain Project
  - Authentication via OpenID Connect
  - Simple job submission and management
  - Data movement
- REST APIs available with UNICORE 7.1
- OIDC under development, will be available in UNICORE 7.2
- Dedicated talk tomorrow!

Putting it all together: the Human Brain Project's HPC platform



https://up.humanbrainproject.eu/#/home

UP | HOME | BROWSER | TOOLS | JOBS 0 | DOCUMENTATION | NMPI | BERND SCHULLER

Useful Shortcuts

Projects
- Upload Data
- Browse My Projects
- Launch an Analysis

Jobs
- Manage My Jobs

Help
- Report an issue

Home Feed

The Unified Portal has been updated to version 0.4.7 (learn more...)

I am...

A Scientific User

I use scientific software created by others to do my job

A Scientific Developer

I program my own tools using Python, C++ and other programming languages

Summary

- Main challenges
  - Concrete needs to access HPC compute and data resources through federations
  - More users and more diverse usage of HPC resource
  - Data sharing, open access and all that
- Solutions
  - UNICORE – compute and storage abstractions
  - Unity – federated identity management
  - UFTP – high-performance data transfer with sharing capabilities

Outlook

- Current and future trend: web-style
  - Authentication via OAuth2
  - RESTful APIs
  - Portals and science gateways
  - Data sharing
  - Maximise end-user friendliness, driven by applications
- Add/extend support for
  - Cloud resources (OpenStack, S3, EC2, …)
  - Hadoop / YARN jobs
  - Virtualised applications (Docker)

Team / Thank you

- Björn Hagemeier, Valentina Huber, André Giesler, Boris Orth, Mariya Petrova, Jedrzej Rybicki, Rajveer Saini and many others at JSC
- Krzysztof Benedyczak, Marcelina Borcz, Rafał Kluszczynski, Piotr Bała and others at ICM / Warsaw University
- Richard Grunzke and others at Technical University Dresden
- Students: Burak Bengi, Maciej Golik, Konstantine Muradov
- … many others who reported bugs, suggested features, contributed code and provided patches

The research leading to these results has received funding from the European Union Seventh Framework Programme (FP7/2007-2013) under grant agreement no. 604102 (Human Brain Project)