

# Verteiltes Rechnen mit UNICORE - Gegenwart und Zukunft

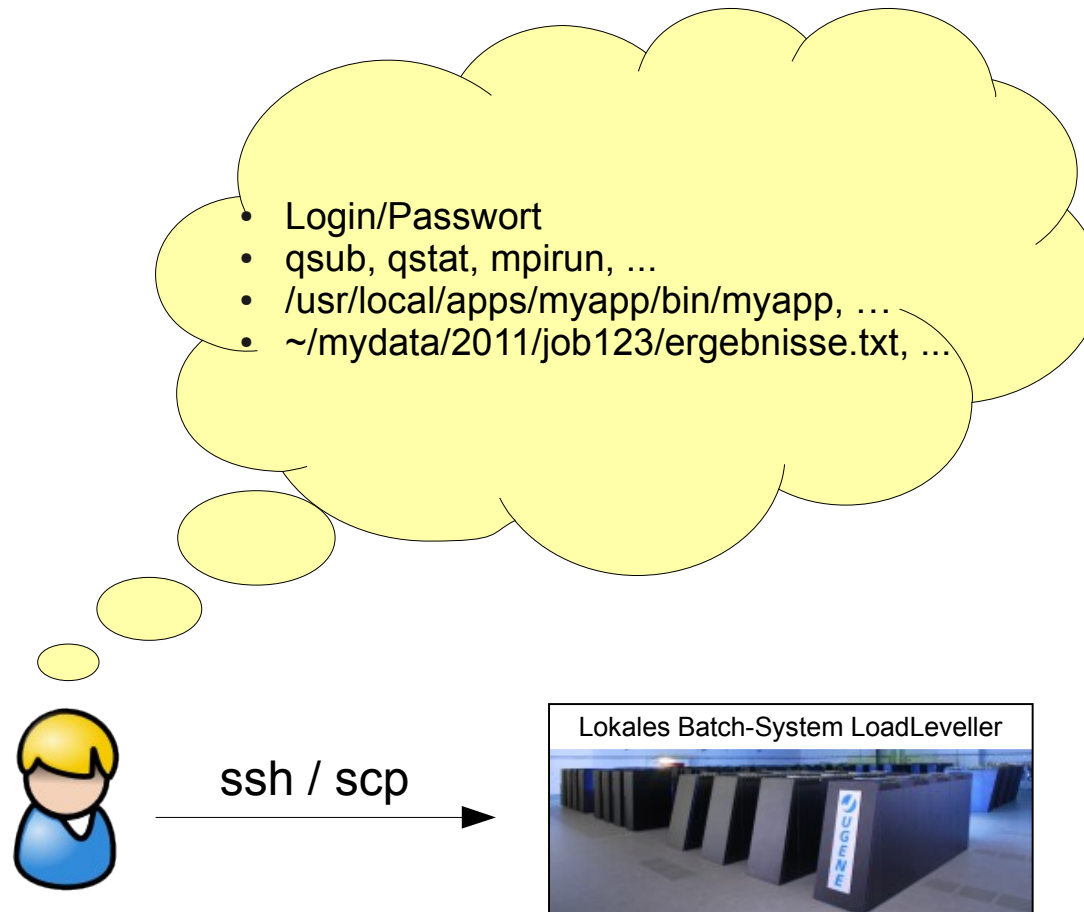
Dr. Bernd Schuller  
Federated Systems and Data division  
Jülich Supercomputer Centre  
Forschungszentrum Jülich GmbH

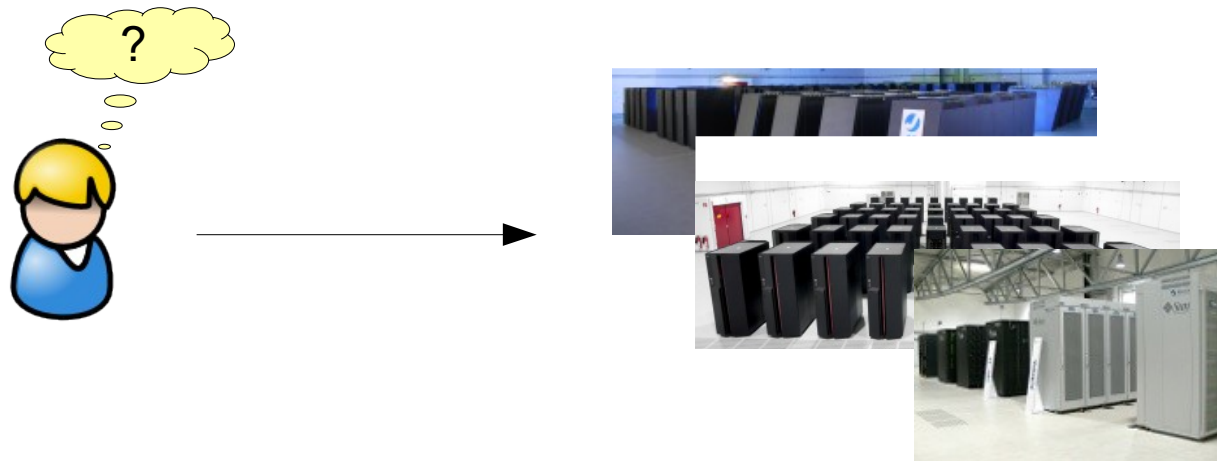
23. Juni 2011  
ZIH, Dresden

# Überblick

- Verteilte Ressourcen
  - Cluster, Clouds, Grid-Infrastrukturen ...
- Was ist UNICORE?
  - Jobs und Daten, Anwendungen, Workflows, Sicherheit
  - Anwendungsbeispiele
- Aktuelle Entwicklungen
  - Daten, Metadaten und Datentransfer
  - Virtuelle Systeme und Clouds
  - Kollaboratives Arbeiten

## Verteiltes Rechnen ...

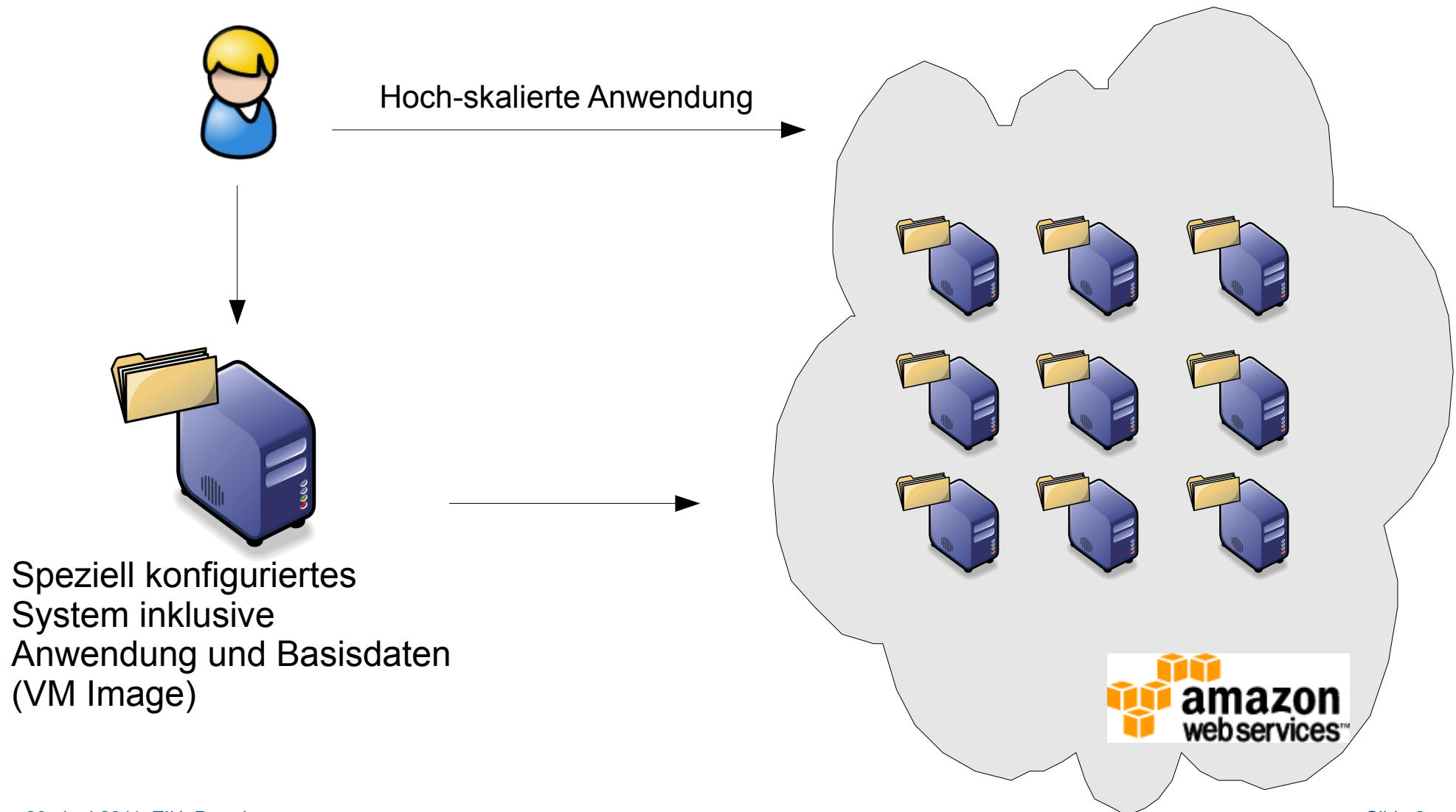




Wie kann ich ...

- ... mehrere, verschiedene Systeme möglichst einfach nutzen,
- ... meine Eingabedaten und Ergebnisse verwalten?
- ... und dies auch System-übergreifend? Workflows?
- Dies war die ursprüngliche Motivation für UNICORE (1997)

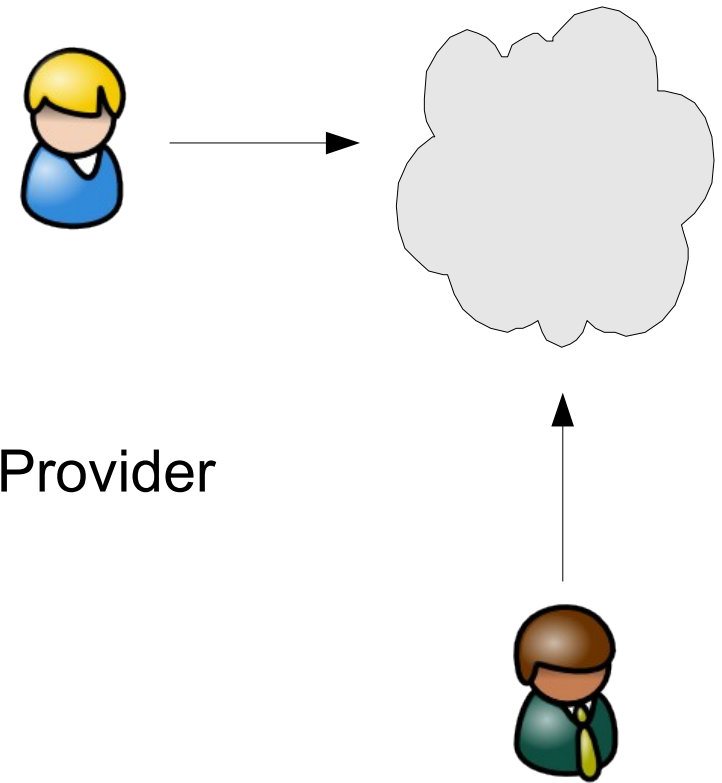
# Anwender-spezifische Umgebungen



## ... nicht schlecht, aber

Wie kann ich als Anwender ...

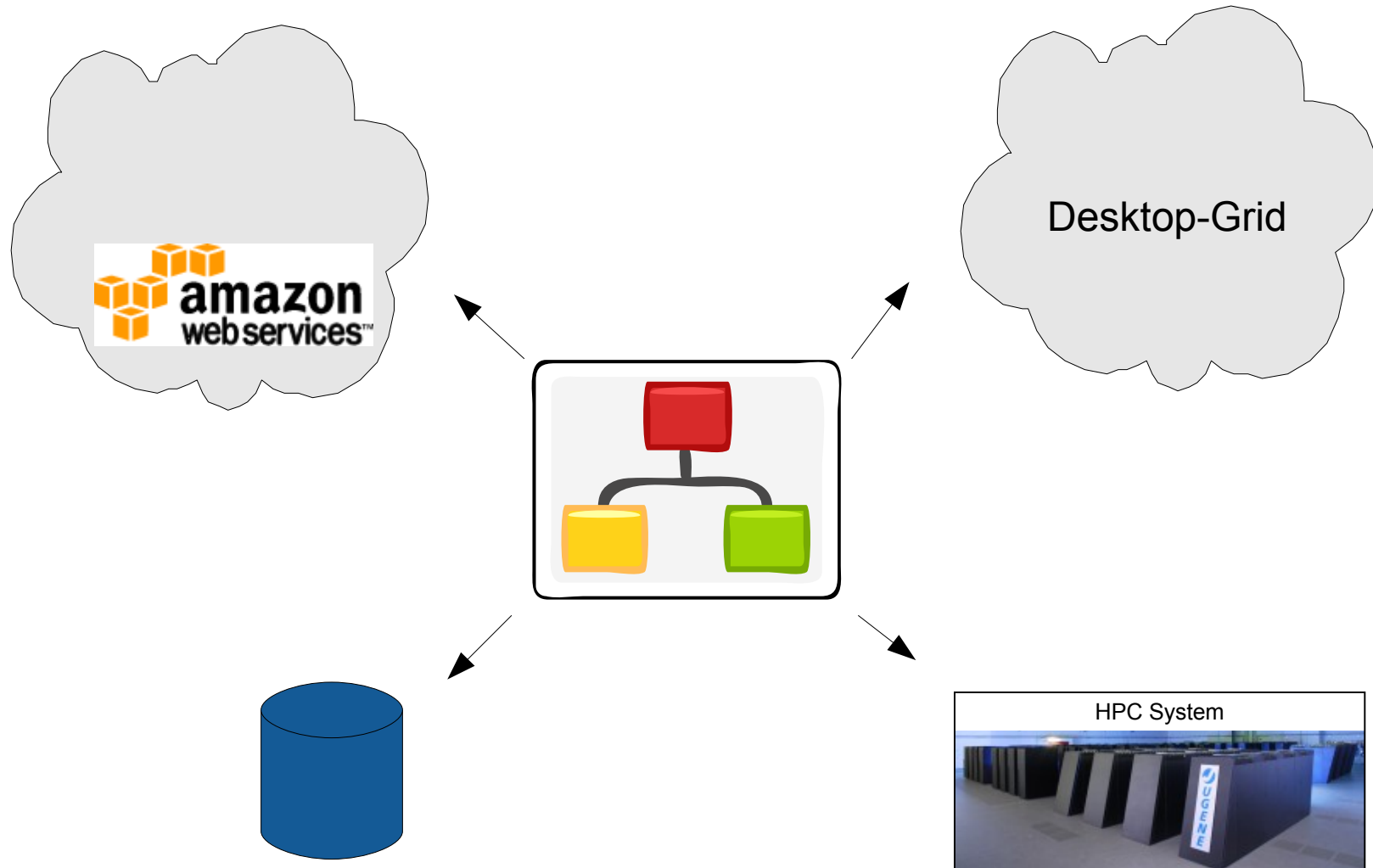
- ... meine Eingabedaten und Ergebnisse verwalten?
- ... meine Anwendungen unabhängig vom Provider gestalten?



Als möglicher Provider solcher Dienste ...

- ... meinen Kunden die Verwendung möglichst einfach machen?
- ... mein Angebot unabhängig von der Technologie gestalten?
- ... wie geht z.B. Sicherheit, Accounting, Logging, Monitoring?

## ... und was ist mit Workflows?

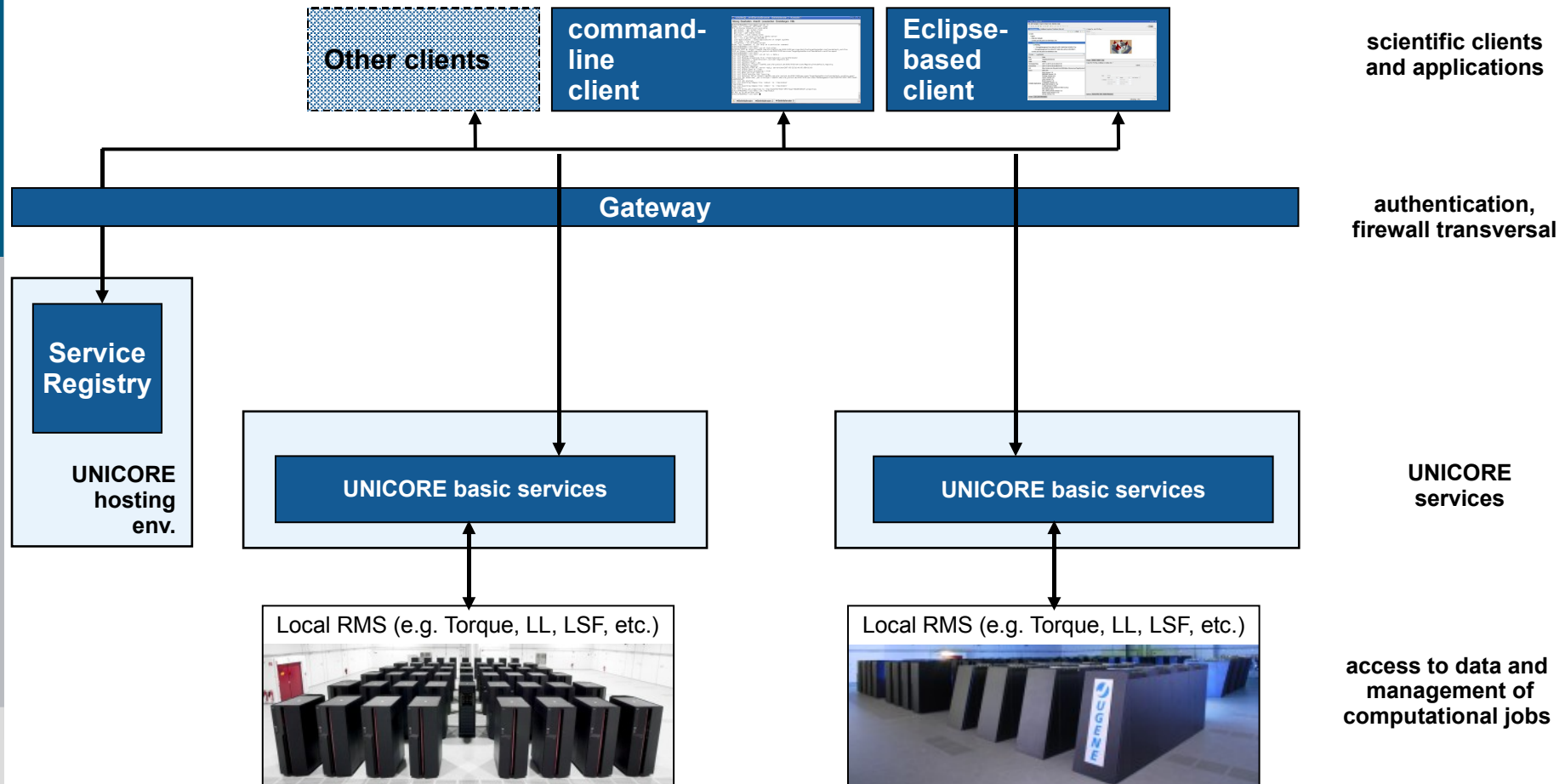




# UNICORE

... ist eine integrierte Middleware für sicheren Zugang zu Rechenressourcen und Daten

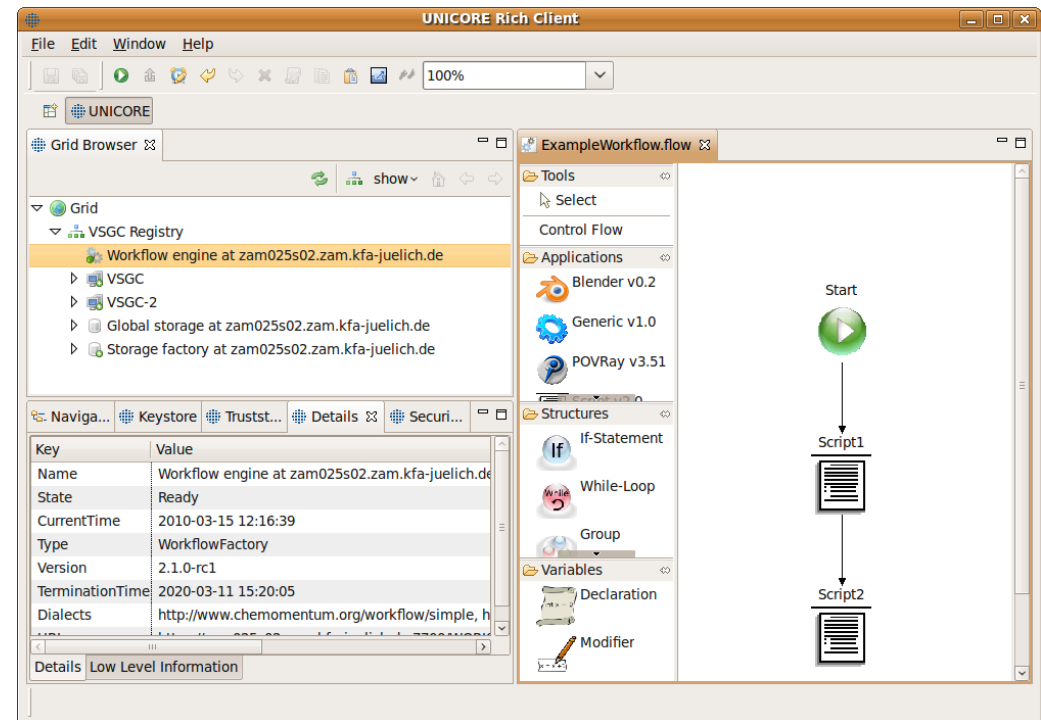
- Einfache Integration in existierende Umgebungen
- Implementiert in Java und Perl
  - UNIXe, MacOS X, Windows
- Unterstützt Torque, LoadLeveller, LSF, SGE, ...
- Effizienter Support, regelmässige Releases
- **Open source unter BSD Lizenz, siehe <http://www.unicore.eu>**



## Clients

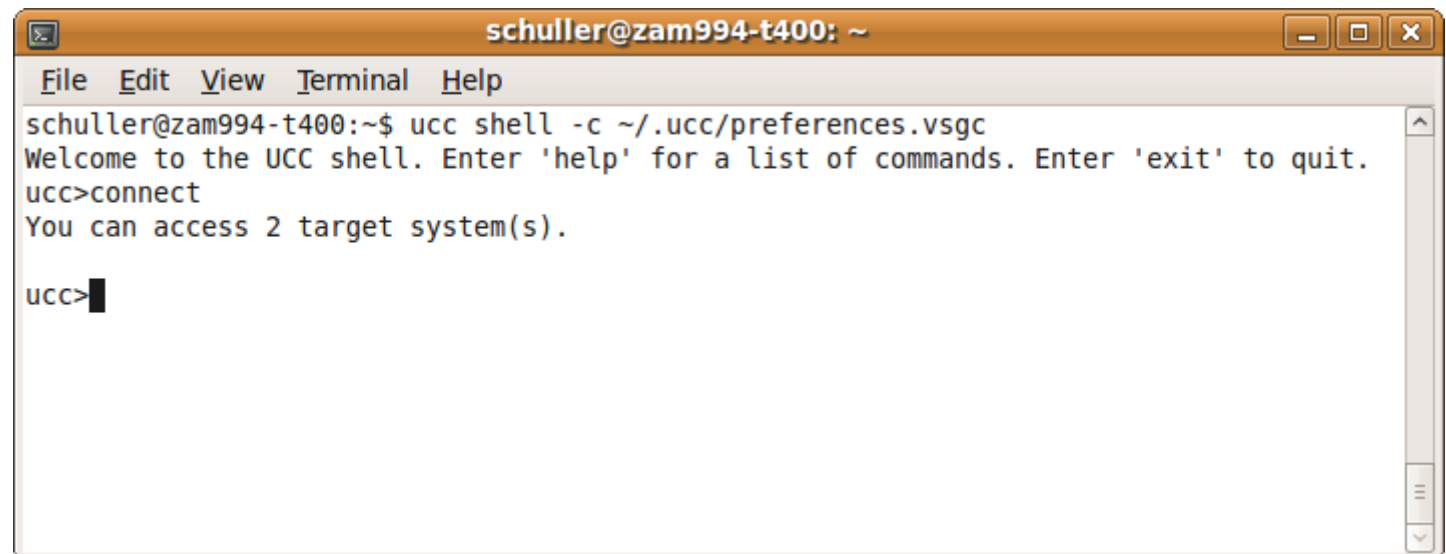
## Der URC – UNICORE Rich client

- Erstellen und Verwalten von Jobs und Workflows
- Integriertes Datenmanagement
- Basiert auf dem Eclipse Framework
- Erweiterbar



## Der UCC - UNICORE Commandline Client

- Bietet Zugang zur vollen UNICORE Funktionalität
- „Batch-Modus“ für high-throughput Szenarien
- Erweiterbar durch
  - neue Kommandos
  - selbstprogrammierte Skripten



```
schuller@zam994-t400: ~  
File Edit View Terminal Help  
schuller@zam994-t400:~$ ucc shell -c ~/.ucc/preferences.vsgc  
Welcome to the UCC shell. Enter 'help' for a list of commands. Enter 'exit' to quit.  
ucc>connect  
You can access 2 target system(s).  
  
ucc>
```

## Sicherheit

## Sicherheit - Überblick

Basiert auf offenen Industrie-Standards

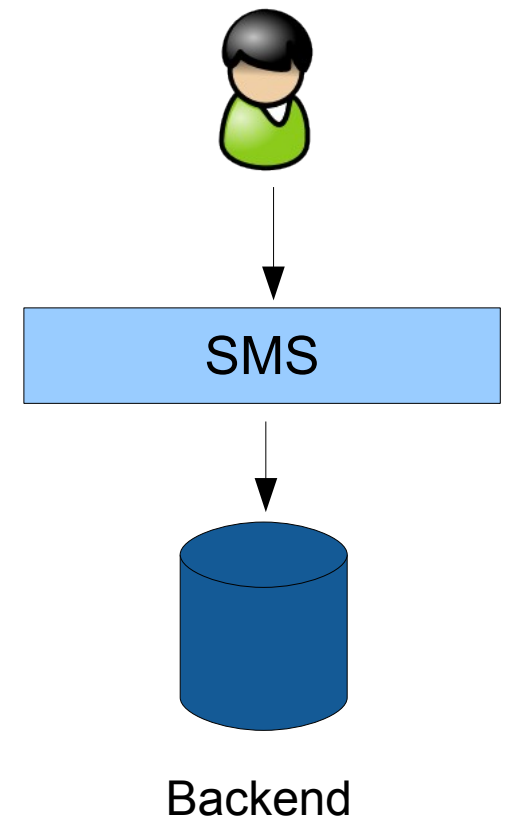
- **X.509** Zertifikate für Benutzer und Server
- **SSL** für die Kommunikation
- Signierte **SAML**-Dokumente (Security assertion markup language)
  - Sichere Delegation
- Verschiedene Quellen für Benutzerattribute zur Autorisierung
- Autorisierung basierend auf **XACML 2.0** Policies
- Erweiterbar, z.B. optionale, eingeschränkte Unterstützung von Proxy-Zertifikaten



## Datenmanagement

## Anbindung von Speichersystemen

- Der UNICORE Storage Management Service („SMS“) ist die zentrale Abstraktion: eine Filesystem-artige Sicht auf Daten
- Typische Operationen
  - mkdir, delete, ls, etc
- Erlaubt das Anstossen von File-Transfers
  - Import/Export von Daten von/zur lokalen Maschine
  - Senden/Empfangen von Daten anderer Server
  - Verschiedene Protokolle möglich

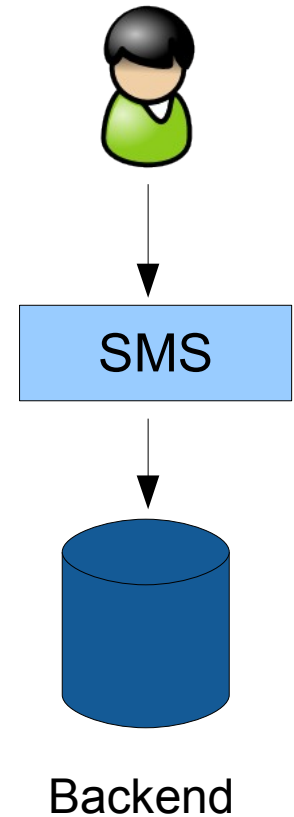


## Existierende SMS-Implementierungen

- Filesystem
- Apache HDFS (Hadoop distributed file system)

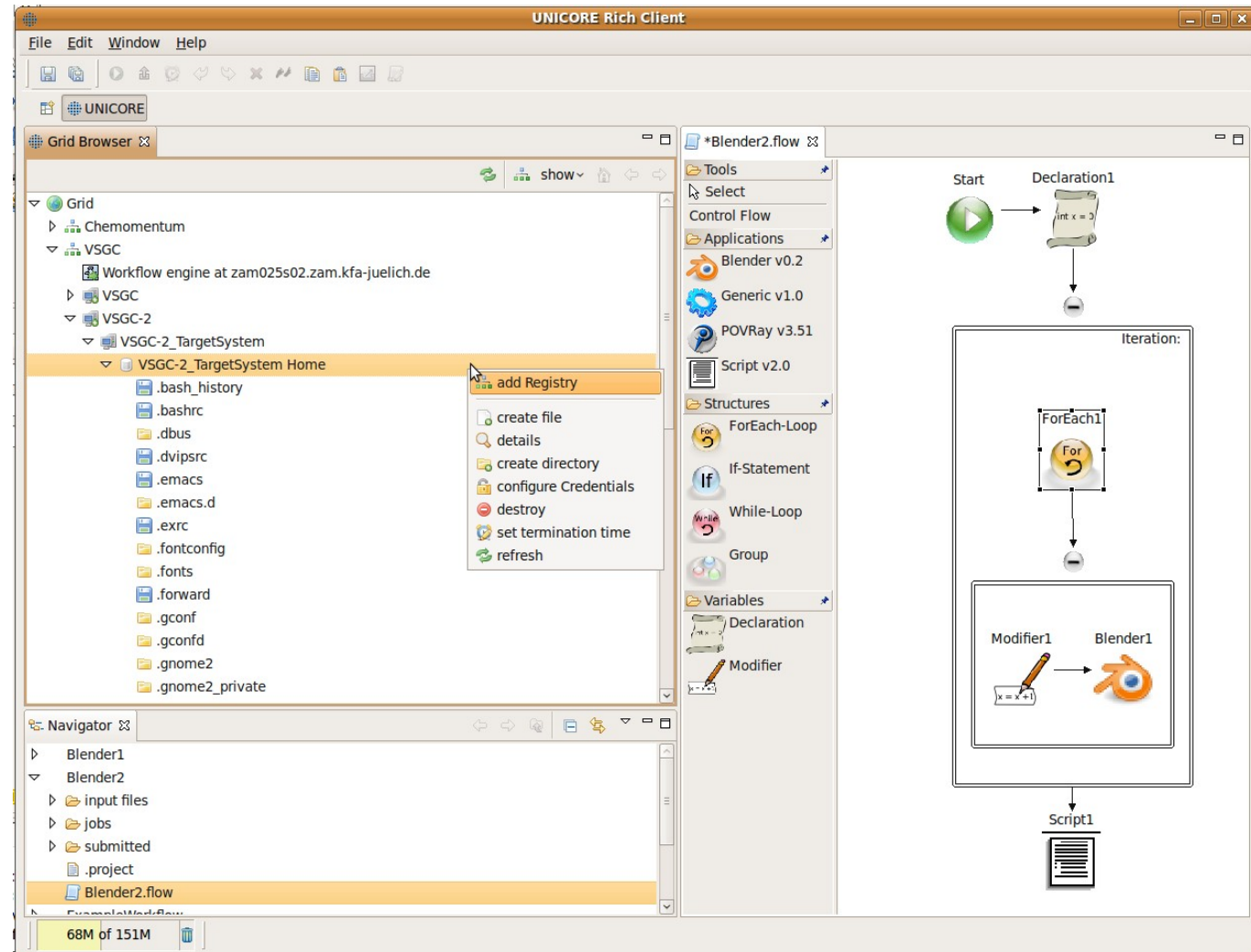


- iRODS (Prototyp)



## Integriertes Datenmanagement im UNICORE Rich Client

- Files erzeugen
- Drag & drop vom/zum Desktop
- Copy & paste

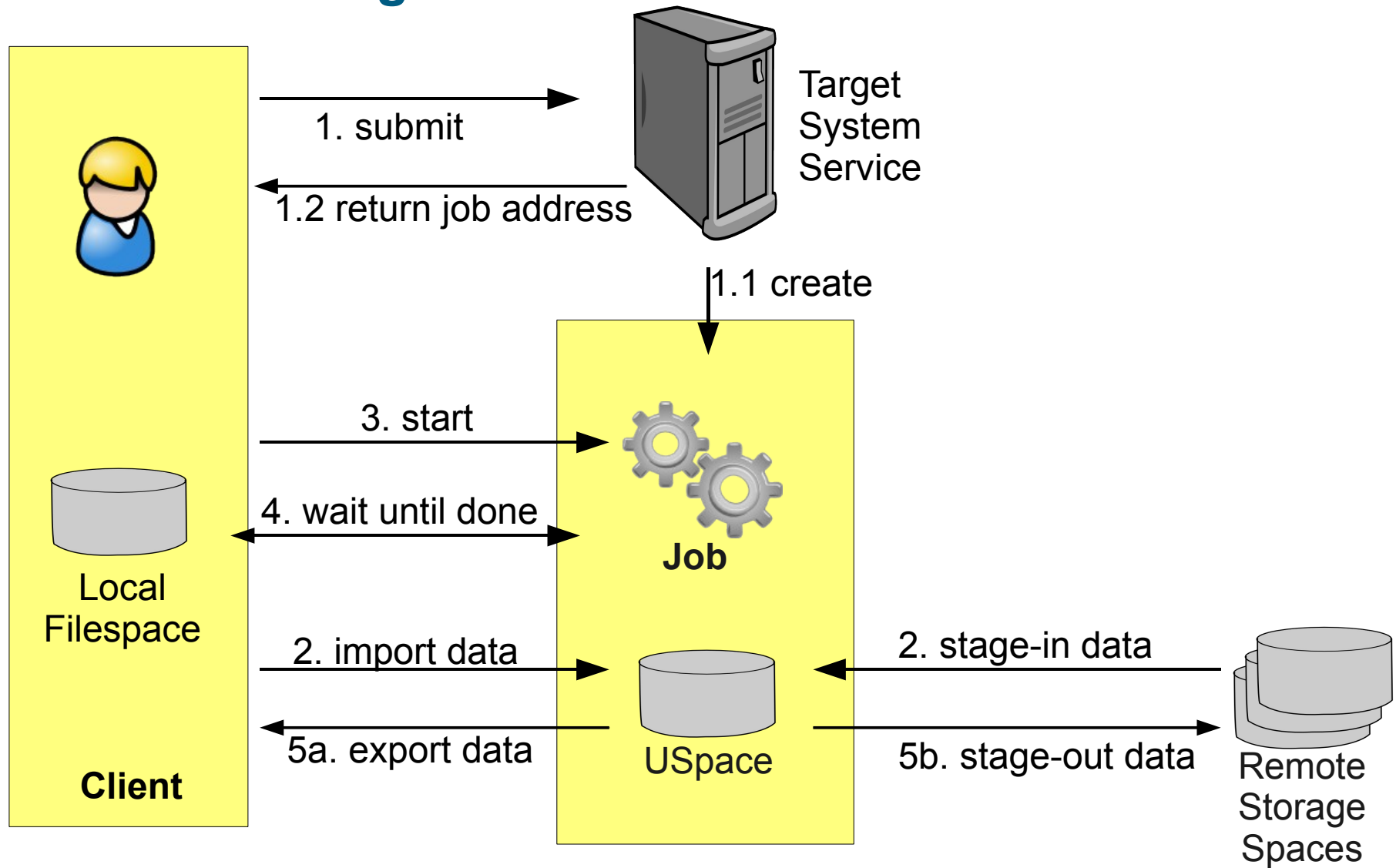


## Jobs

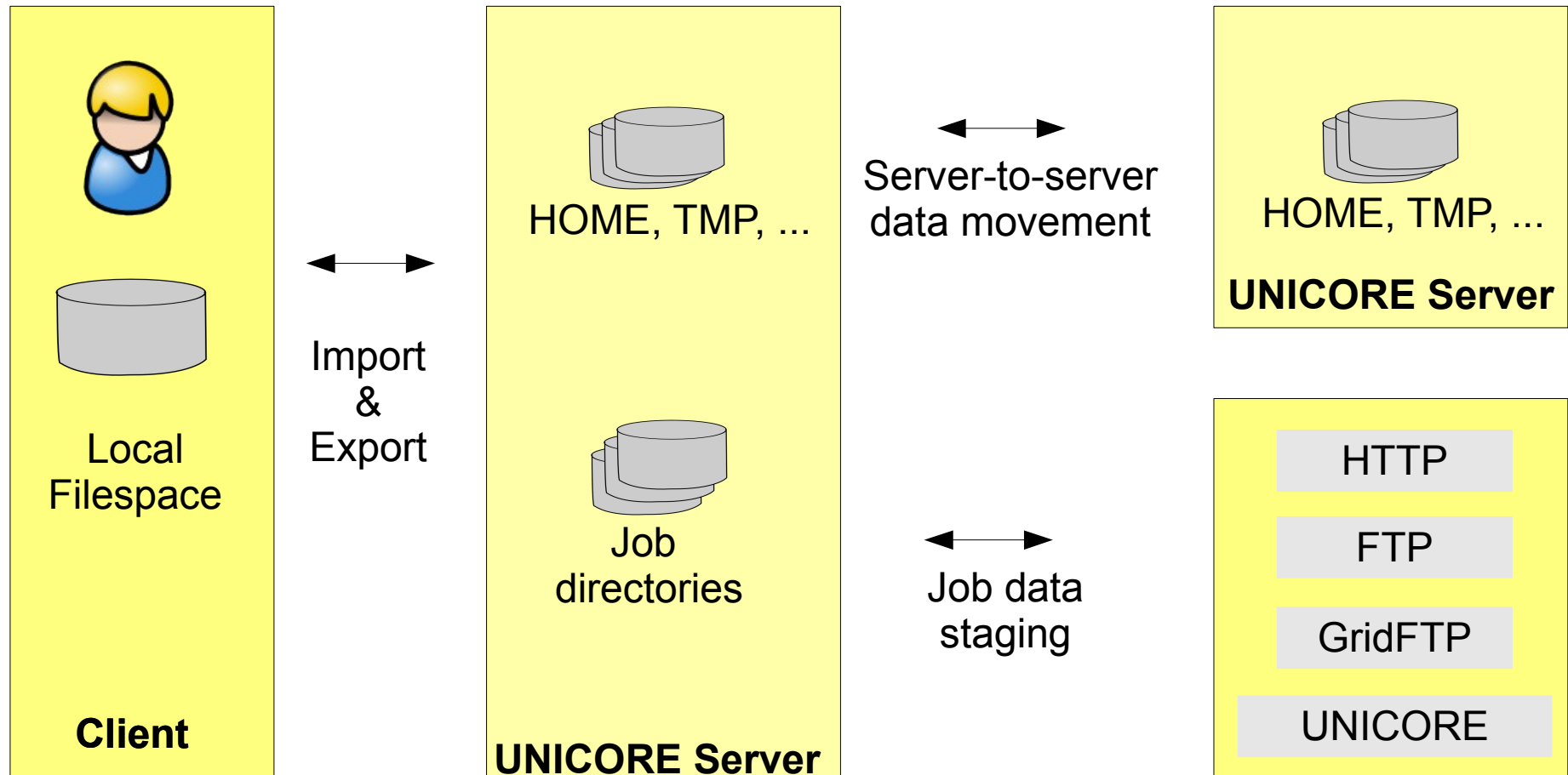
## Anbindung von Compute-Ressourcen

- „Target system service“ (TSS) und „Job management service“ (JMS) als Abstraktion
- Submission (TSS) und Management (JMS) von Einzeljobs
  - Anwendung, oder direkt das Executable
  - Argumente, Environment-Variablen, stderr, stdout, ...
  - Eingabedaten und Resultate
  - Angeforderte Ressourcen (Cores, Speicher, ...)

## Job-Ausführung



## Storages, Jobs und Daten





## Anwendungs-Integration

## Anwendungs-Integration

- Anwendung = Software-Paket, das im Batch-Betrieb ausführbar ist
- Installation auf dem Cluster durch einen Administrator
- Konfiguration im UNICORE-Server
- Unterstützung im grafischen Klienten
  - Anwendungs-spezifisches Plugin (ev. Entwicklungsaufwand)
  - „Generic“ Plugin (eingebaut)

## Beispiel: MOPAC

- Molecular Orbital PACkage
- Ein semi-empirisches Quantenchemie-Programm.
- Typische Anwendung: Optimierung von 3D – Molekülstrukturen

## Beispiel: MOPAC - 2

### Anwendungs- spezifisches Plugin

Input for mopac\_test

Job Preparation for MOPAC calculation

Job name: mopac\_test

Input file: input.slf

Output file: output.slf

Keywords Structures

AM1 EF GNORM=0.01 NOINTER

☐ Define custom keyword set

Predefined keyword sets

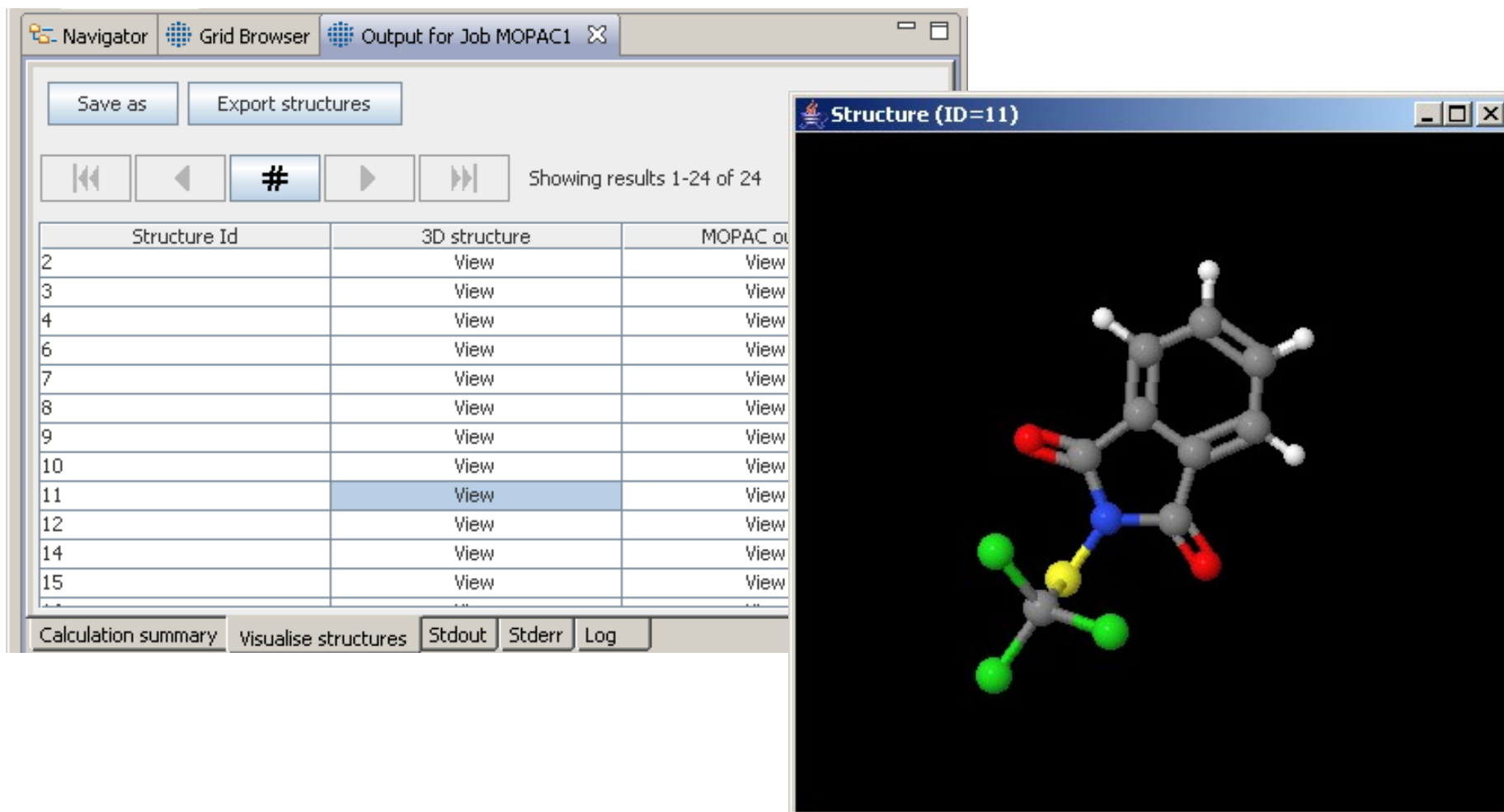
AM1 Opt EF GNORM=0.01

Add Remove

Input Files Variables Resources

## Beispiel: MOPAC - 3

Anzeige der Ergebnisse (Anwendungs-spezifisches Plugin)



The screenshot shows a software interface with a window titled "Output for Job MOPAC1". It contains a table of results and a 3D molecular structure visualization.

**Table of Results:**

Structure Id	3D structure	MOPAC out
2	View	View
3	View	View
4	View	View
6	View	View
7	View	View
8	View	View
9	View	View
10	View	View
11	View	View
12	View	View
14	View	View
15	View	View

The table shows results for 24 structures, with the first 15 displayed. Structure 11 is highlighted. The interface includes navigation buttons (first, previous, #, next, last) and a status bar at the bottom with tabs for "Calculation summary", "Visualise structures", "Stdout", "Stderr", and "Log".

**3D Molecular Structure Visualization:**

A separate window titled "Structure (ID=11)" displays a 3D ball-and-stick model of a molecule. The molecule features a benzene ring substituted with a sulfonamide group (SO<sub>2</sub>NH<sub>2</sub>) and a carboxylic acid group (COOH). The atoms are color-coded: carbon (grey), oxygen (red), nitrogen (blue), and sulfur (yellow).

## Beispiel: MOPAC - 4

Anzeige der Ergebnisse (Anwendungs-spezifisches Plugin)

The screenshot shows the UNICOREpro interface with the 'Output for Job MOPAC1' window open. The window title is 'MOPAC Output (ID=11)'. The output text is as follows:

```

20      H      2.2234   5.2809   0.0368

      ATOMIC ORBITAL ELECTRON POPULATIONS

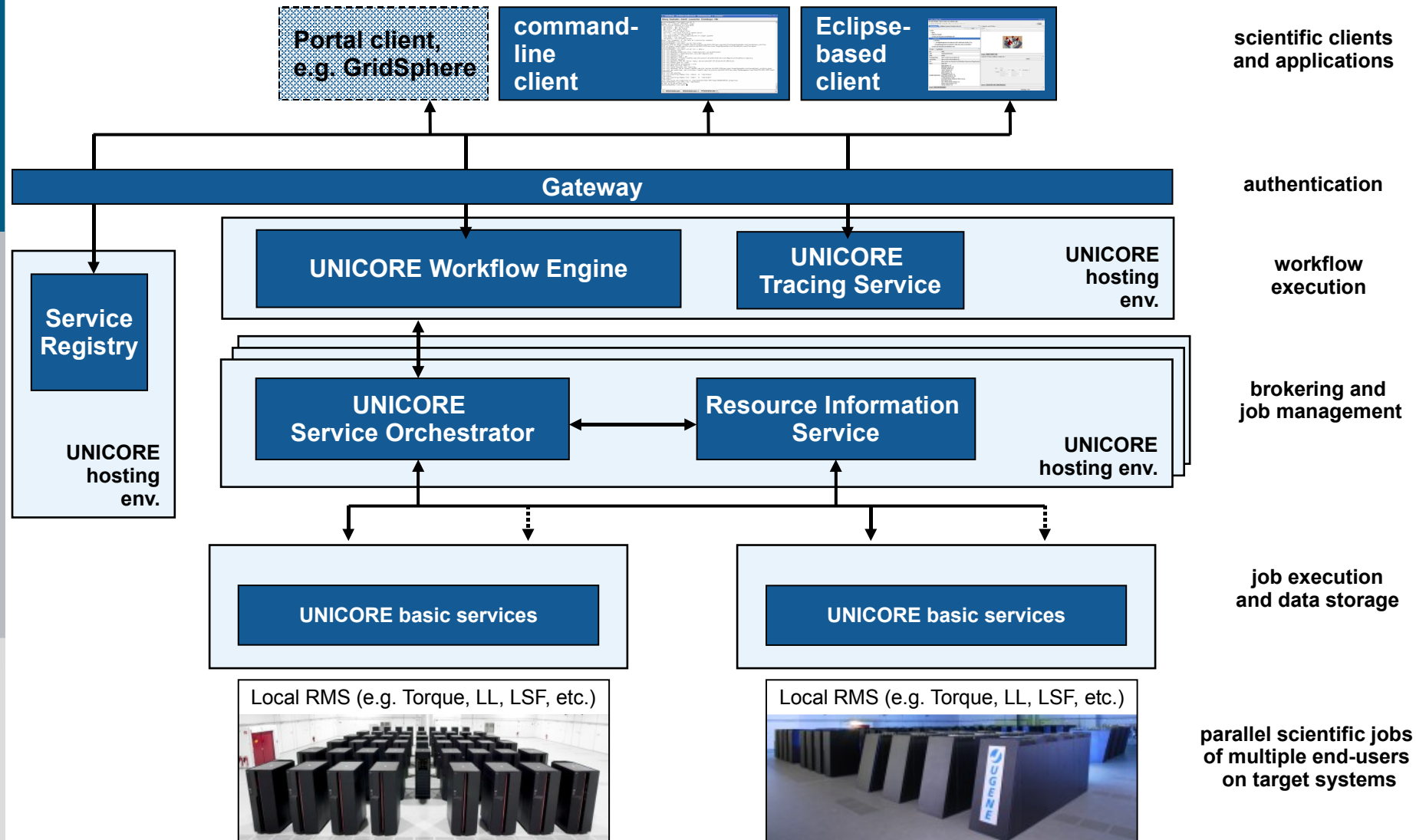
1.52289  1.11894  1.11465  1.73343  1.21920  0.80265  0.88922  0.73175
1.21915  0.88319  0.80871  0.73178  1.85333  1.56208  1.19455  0.94320
1.20254  0.93959  0.91776  1.05352  1.91230  1.58276  1.35867  1.39096
1.20254  0.92150  0.93591  1.05350  1.91231  1.16021  1.78123  1.39099
1.30671  0.90877  0.92043  1.03339  1.21627  0.95751  0.94315  0.94235
1.21627  0.97787  0.92282  0.94234  1.98658  1.74845  1.52952  1.76269
1.98661  1.68304  1.38681  1.94996  1.98657  1.35382  1.92554  1.76130
1.22319  0.98852  0.92831  0.97330  1.22318  0.91479  1.00205  0.97329
0.83746  0.83744  0.85233  0.85235

TOTAL CPU TIME:          1.22 SECONDS

== MOPAC DONE AT Fri Jul  3 16:00:24 2009 ==
  
```

The interface also shows a 'Navigator' panel on the left with buttons for 'Save as', 'Export structures', and a table of 'Structure Id' values (2, 3, 4, 6, 7, 8, 9, 10, 11, 12, 14, 15). At the bottom, there are buttons for 'Calculation summary' and 'Visualise structures'.

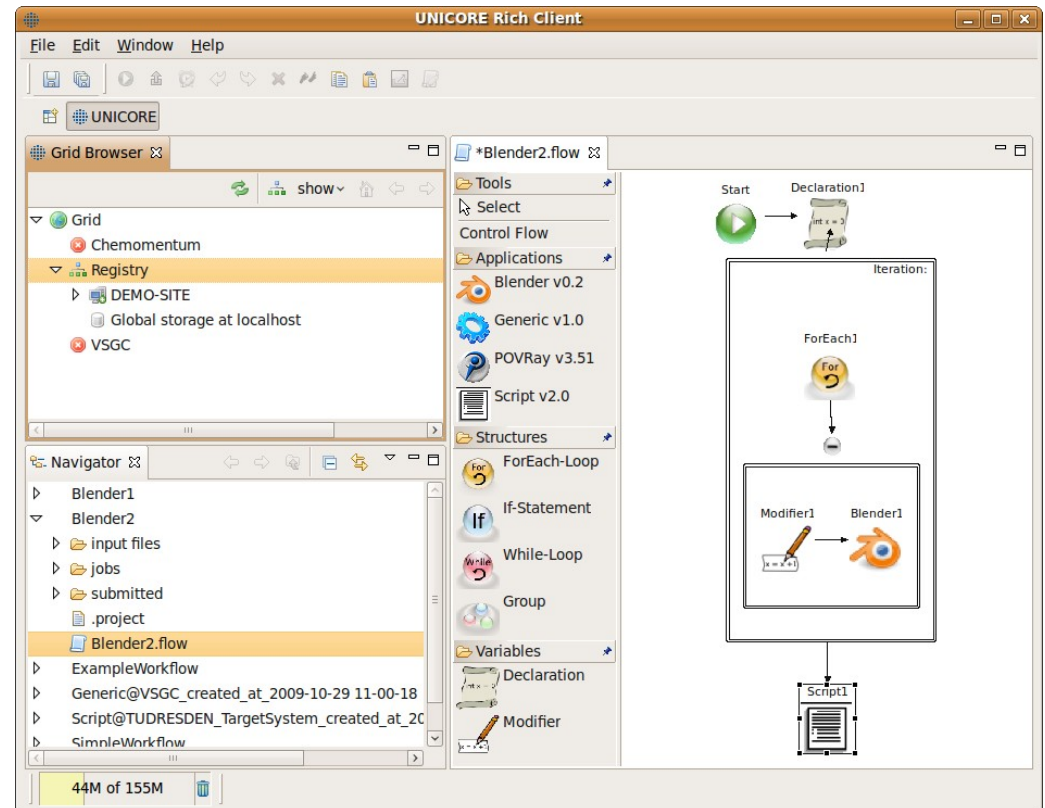
## Workflows





## Workflow features

- Simple graphs (DAGs)
- Workflow variables
- Loops and control constructs
  - while, for-each, if-else
- Conditions
  - Exit code, file existence, file size, workflow variables
- Clients
  - UNICORE Rich client
  - Commandline client



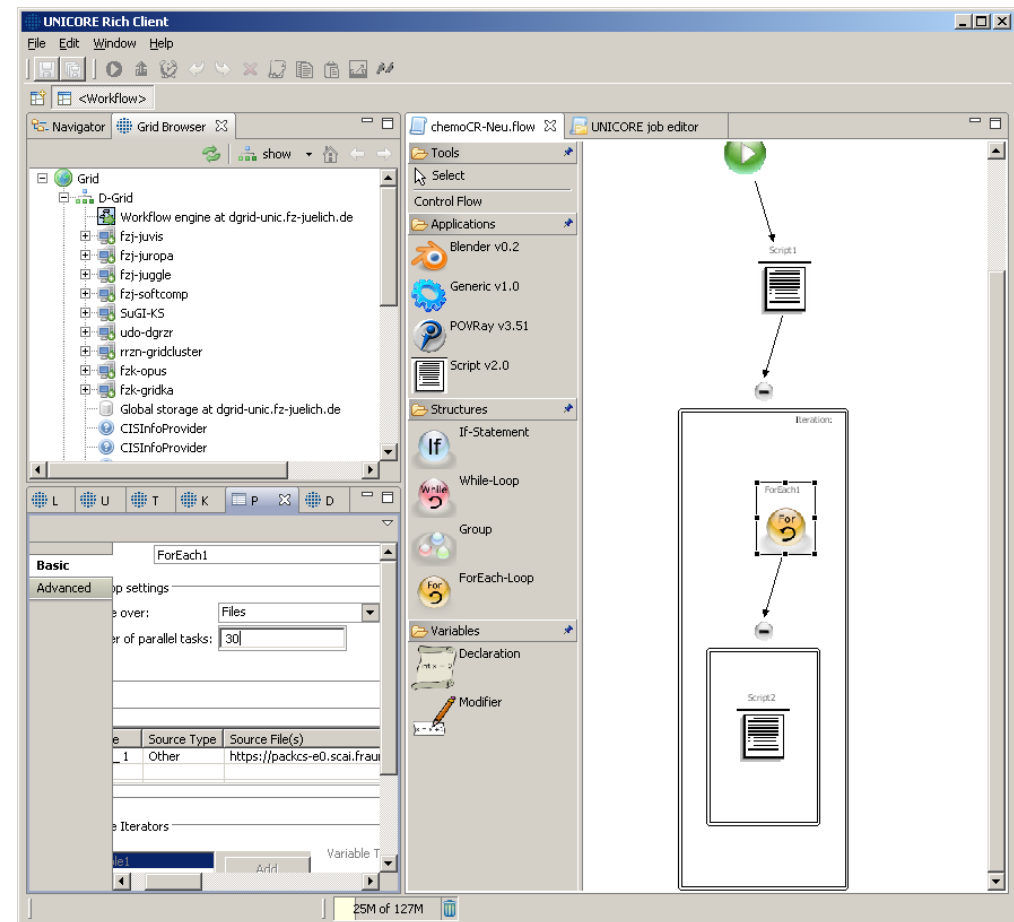
# Anwendungsbeispiel:

## The grand patent challenge

- Annotation aller chemischen Struktur-Beschreibungen in Patent-Dokumenten
- Verfahren
  - Identifikation von Molekülbeschreibungen in Dokumenten
    - *SMILES: CC(C)CC(=O)*
    - *2D-Zeichnungen oder Bilder*
    - *Molekülname*
  - Eingabe: Mehrseitige Dokumente (pdf, tif, ...)
  - Ausgabe: Datenbank
- High throughput Anwendung: Dieselbe Programm-Folge bearbeitet viele Dokumente unabhängig voneinander

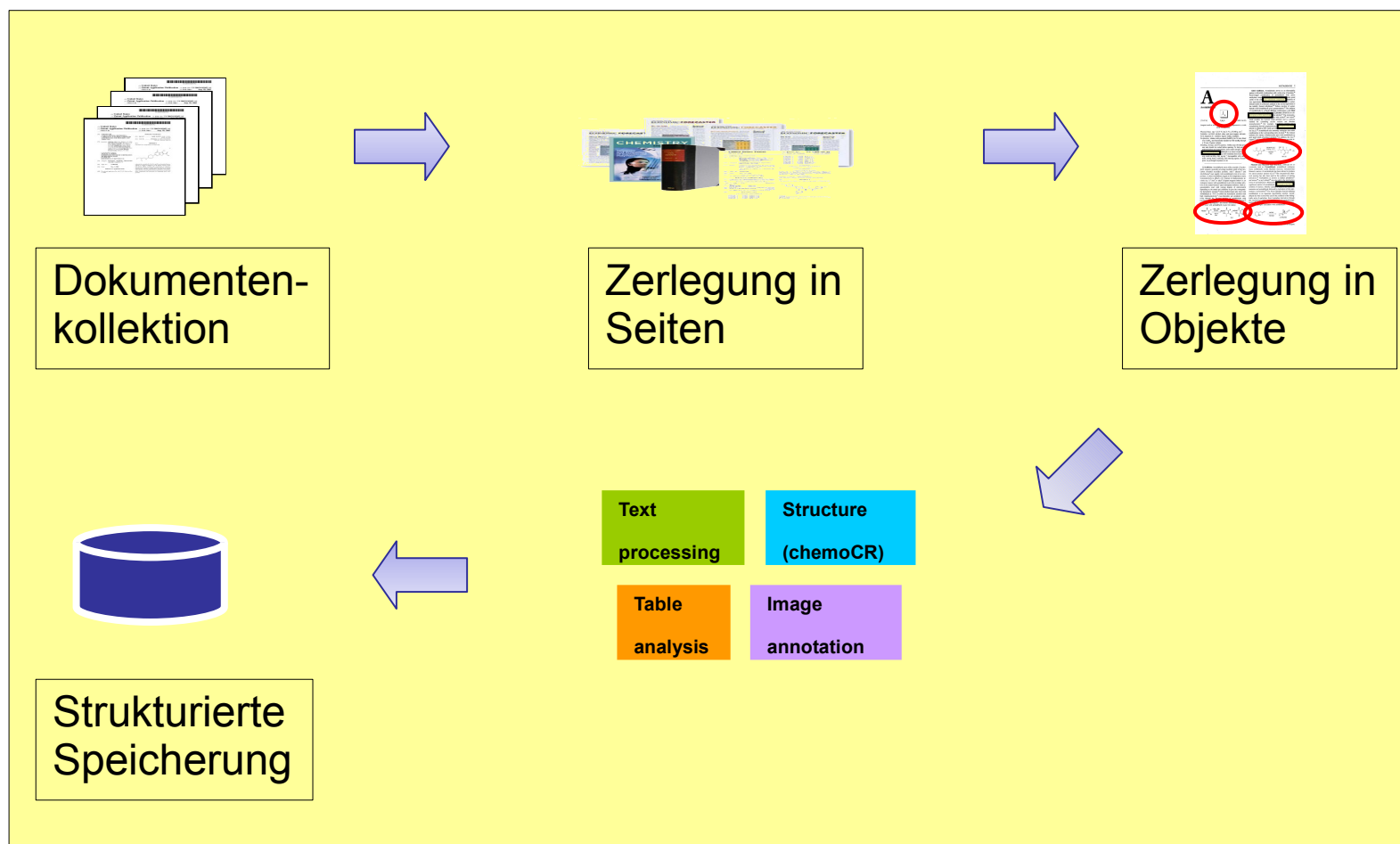
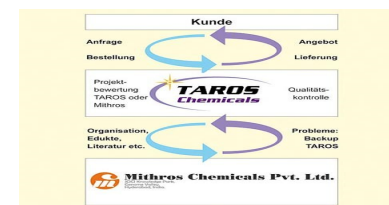
## Realisierung mittels UNICORE Workflows

- 16000 Dokumente von durchschnittlich 14,5 Seiten
- 747000 Moleküle rekonstruiert
- 100 Stunden Rechenzeit
- auf „Juggle“ Cluster (D-Grid)



# Aktuelles BMBF-Projekt

## UIMA-HPC



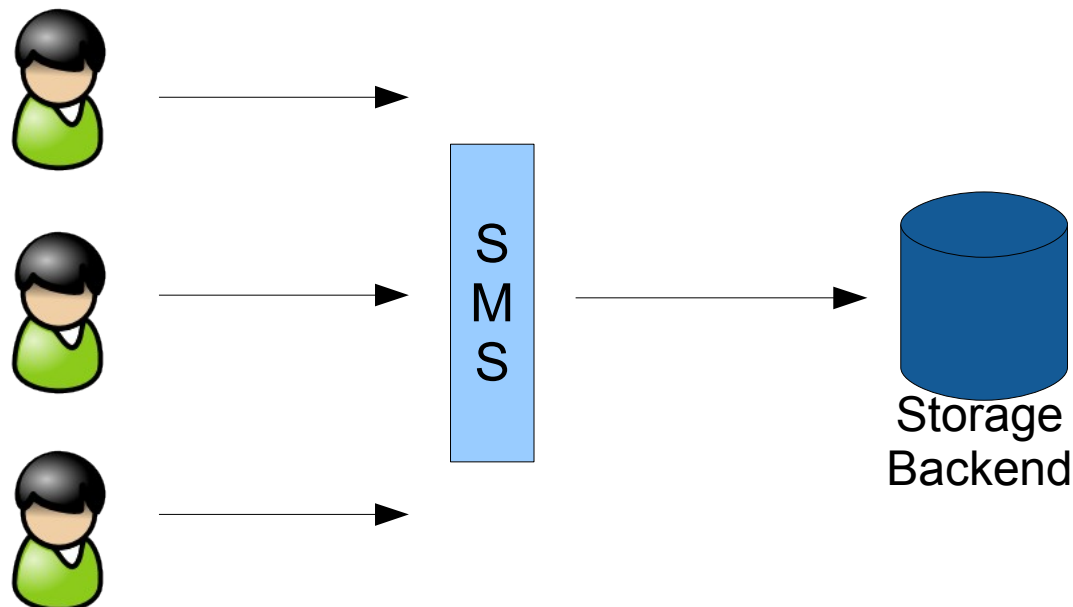
## Aktuelle Entwicklungen

- Daten, Metadaten und Datentransfer
- Virtuelle Systeme
- Kollaboratives Arbeiten

## Skalierbare Speicherdienste

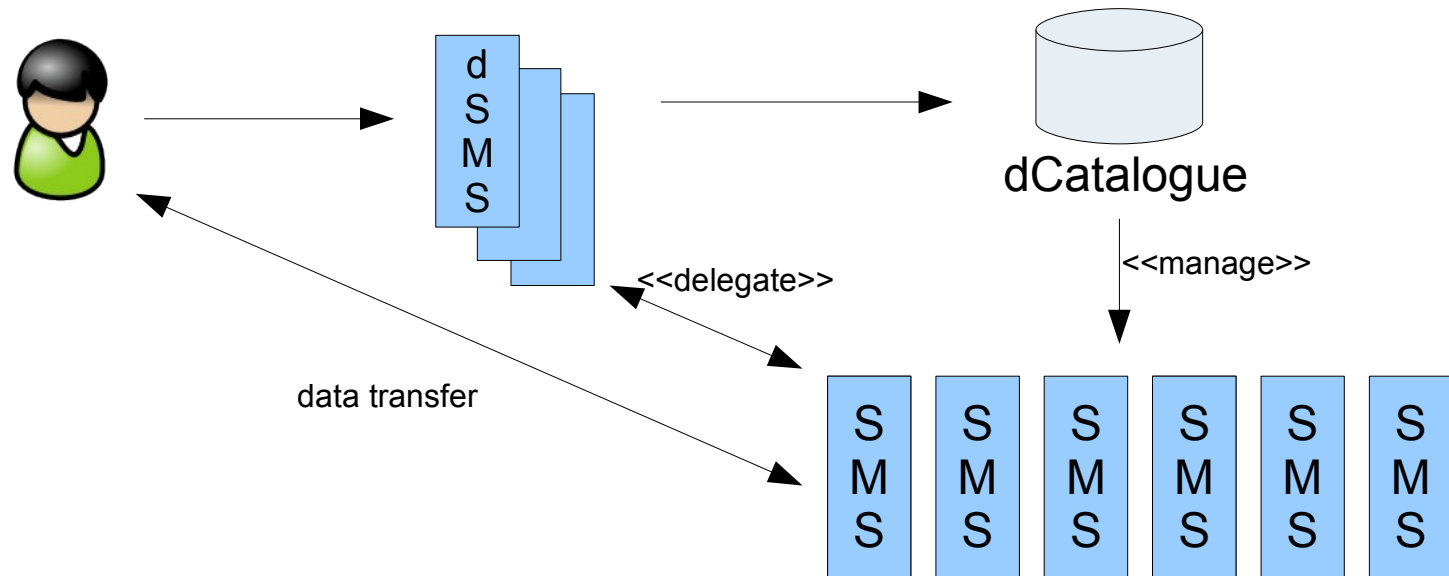
## Skalierbare Speicherdienste: Motivation

- Zentrale Abstraktion in UNICORE ist der Storage management service (SMS)
- Mögliche Engpässe bei grossen oder häufig benutzten Speichersystemen



# Ein Ansatz: distributed SMS

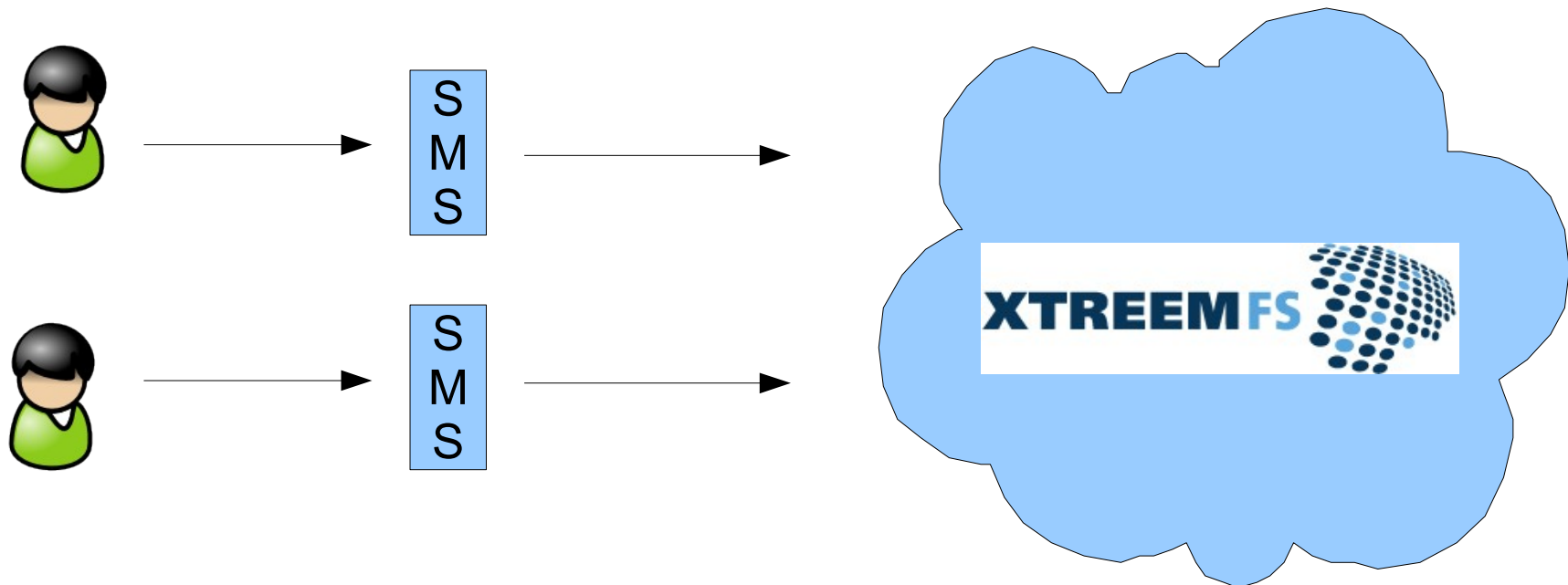
- Distributed Storage management service (dSMS):
  - Selbes Interface wie SMS, aber delegiert Operationen an “echte” SMS
  - Ermöglicht “beliebig viele” Zugangspunkte
  - Datentransfer direkt von/zu den SMS
  - Ähnliche Architektur wie z.B. Apache Hadoop
- File-Katalog(dCatalogue), verwaltet Dateien und angebundene SMS





# Alternative: XtreemFS und UNICORE

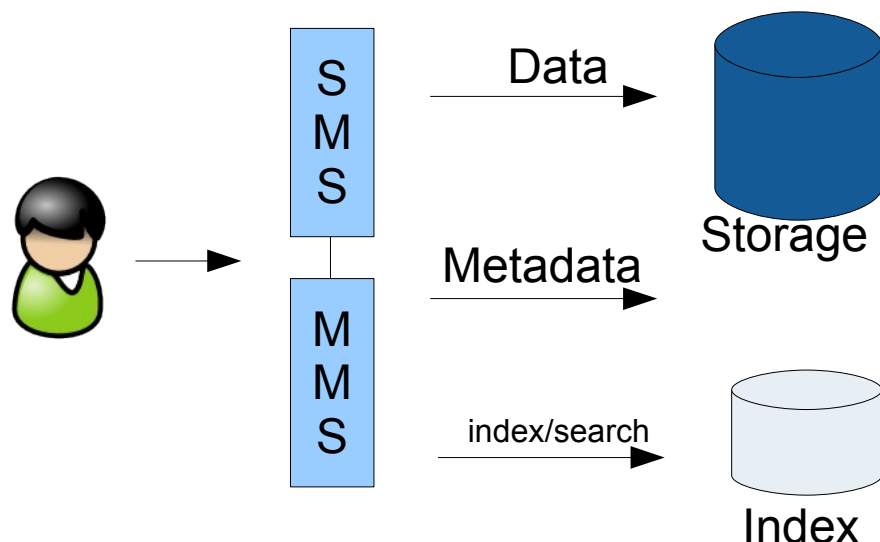
- XtreemFS
  - *Verteiltes Filesystem (WAN/Internet), lokal gemountet (FUSE)*
  - *Bietet globalen Namensraum*
- Mehrere Einstiegspunkte über mehrere SMS-Instanzen



## Metadaten

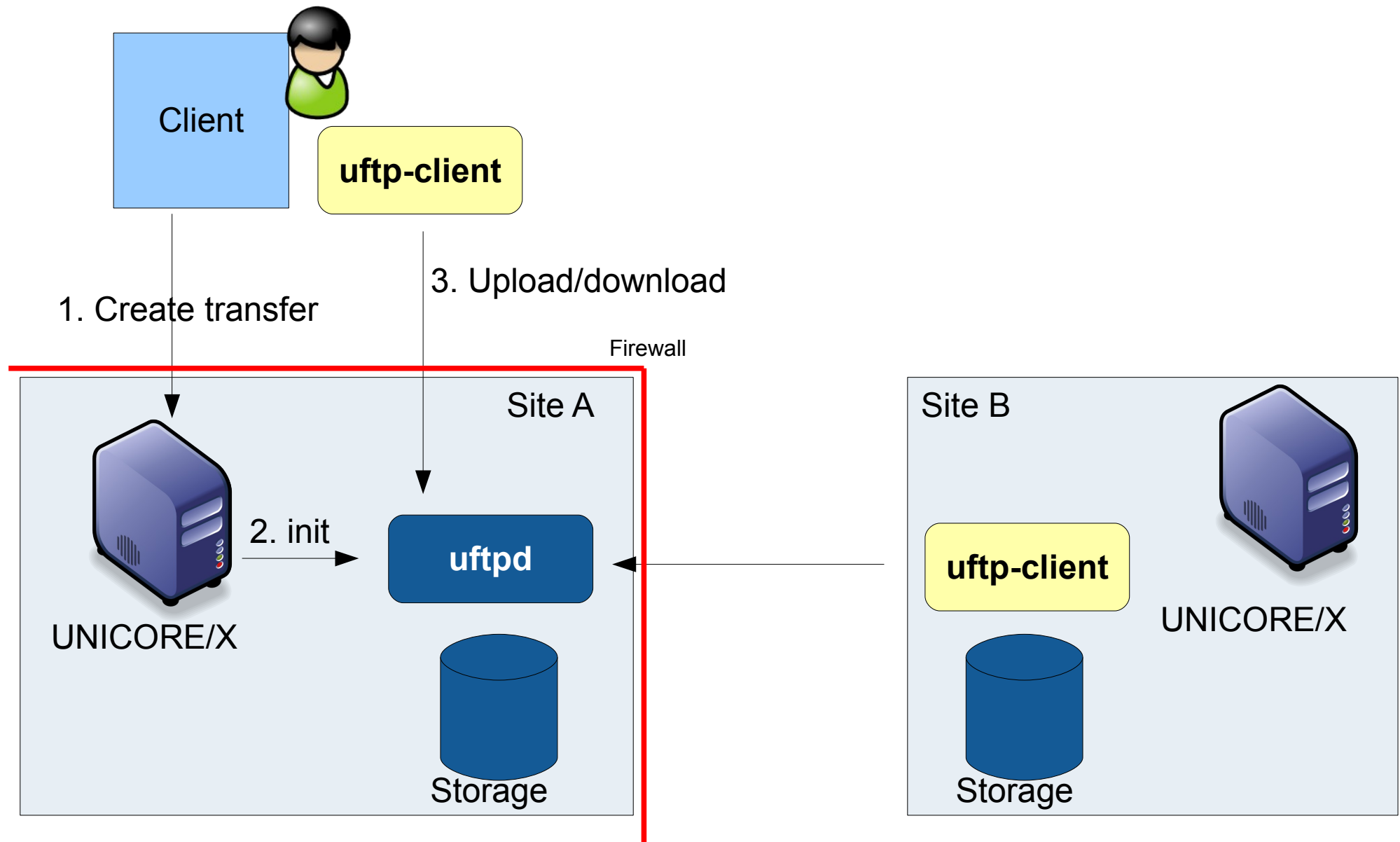
# Integriertes Metadaten-Management

- Dezentral: Je ein „metadata management service“ (MMS) pro Speicher (SMS)
- Schema-frei: Metadaten als Schlüssel/Wert-Paare
- Metadaten werden direkt „neben“ den Daten gespeichert
- Benutzer-definiert und (halb-)automatische Extraktion



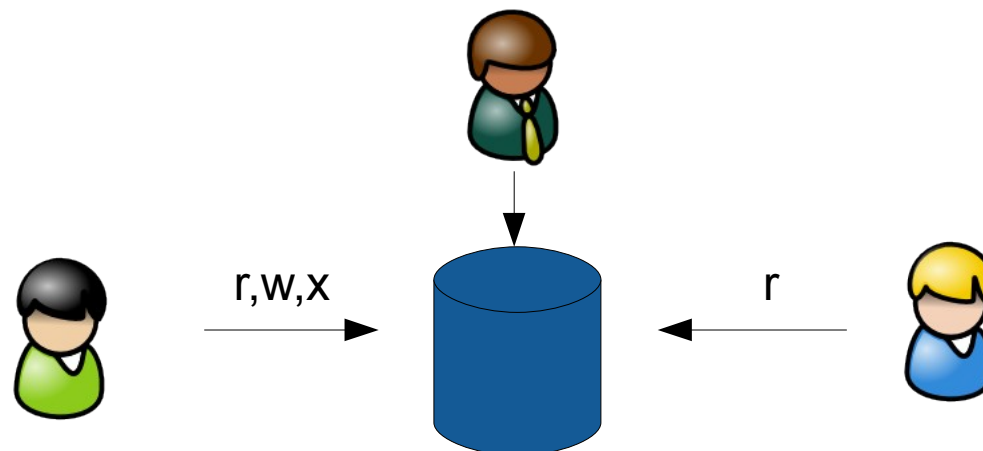
```
schuller@zam994-t400:/$ ucc-vsgc ls -l u6://VSGC-2/
-rw-          1670701 2011-02-24 09:26 /Documents/ref
{
  "Content-Type": "application/pdf",
  "Creation-Date": "2010-09-23T16:25:05Z",
  "Last-Modified": "2010-09-23T16:25:11Z",
  "created": "Thu Sep 23 18:25:05 CEST 2010",
  "creator": "Adobe InDesign CS5 (7.0)",
  "producer": "Adobe PDF Library 9.9",
  "resourceName": "/Documents/refcard-hadoop.pdf",
  "trapped": "False",
  "xmpTPg:NPages": "6"
}
```

# File-Transfer mit UFTP

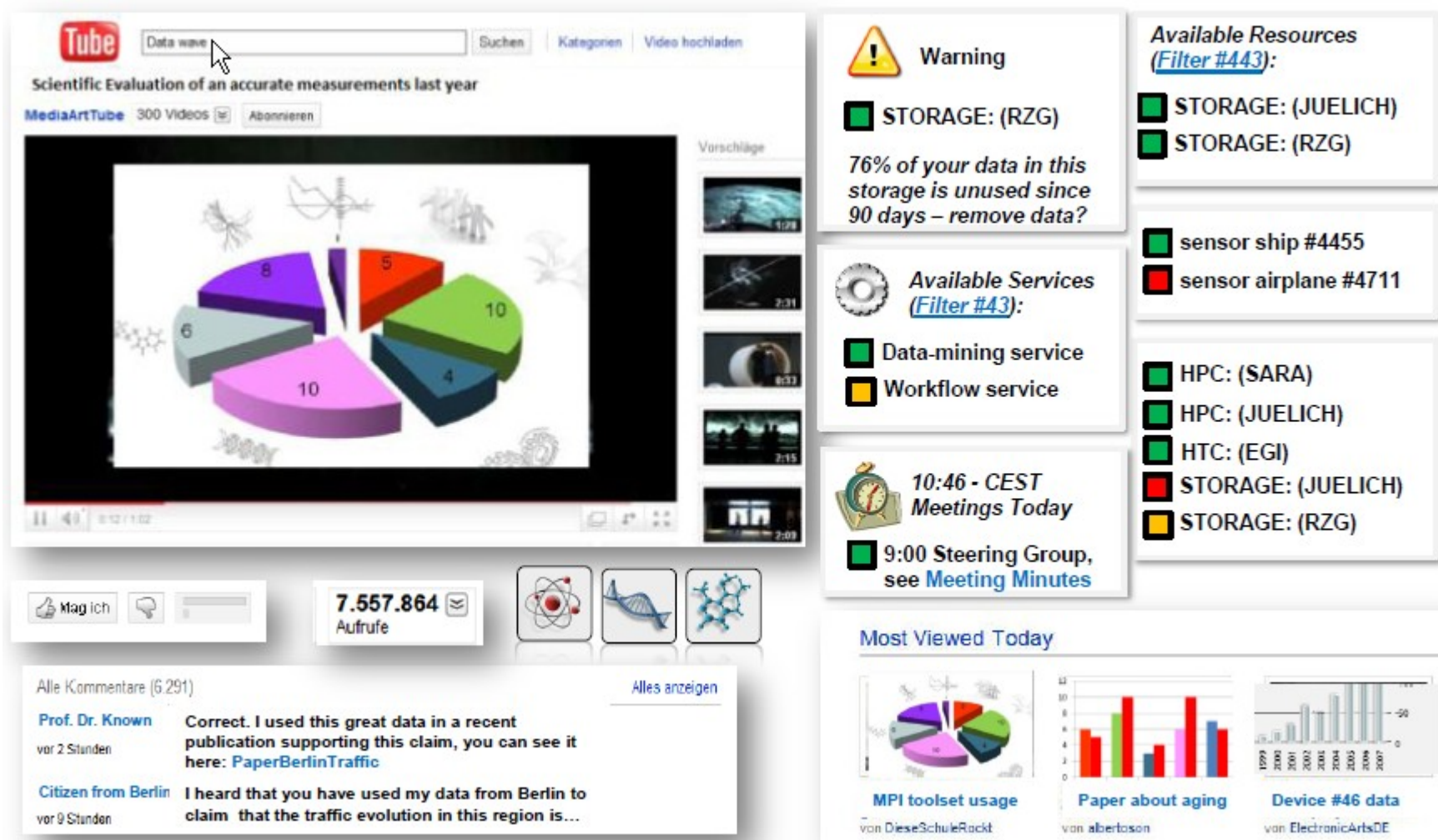


## Kollaboratives Arbeiten

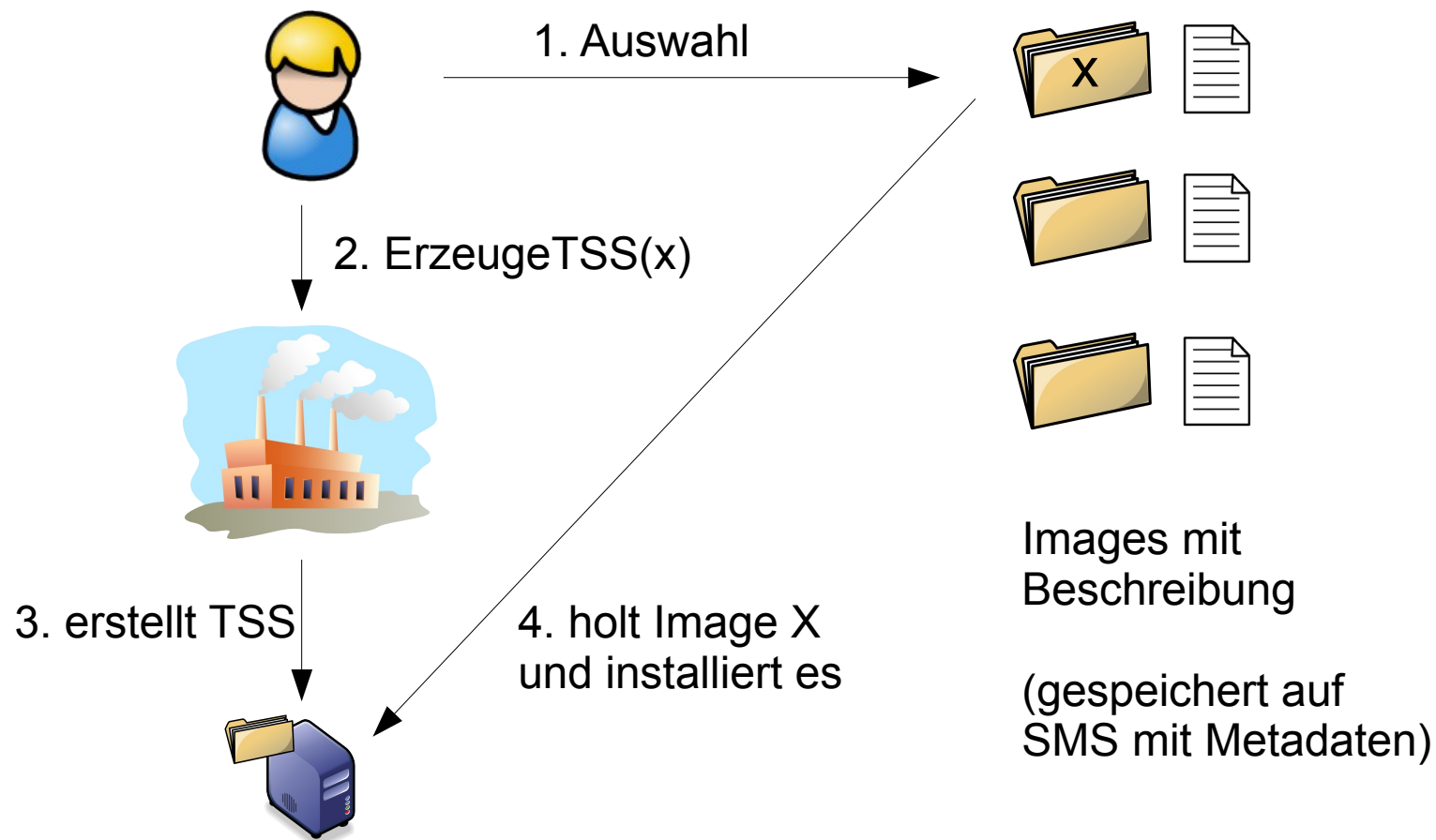
- Bisher:
  - für Dateien: meist Unix-artig (-rwxr--r--)
  - für Dienste: in UNICORE sind nur recht einfache Regeln möglich (z.B. Zugriff für alle Benutzer, nur für eine Gruppe, oder gar nur für den Besitzer)
- In „traditionellen“ Grids (EGEE / EGI, aber auch D-Grid)
  - „Grosse“ virtuelle Organisationen (VOs), abgebildet auf Unix-Gruppen. Hoher Erstellungsaufwand.



## Ziel: flexible Kollaborationen, Vorbild: das Web 2.0!



# Unterstützung von Anwender-spezifischen Umgebungen



.... mit kleinen  
Erweiterungen  
in UNICORE  
realisierbar!



**... wie schnell sind eigentlich Datentransfers?**

## UFTP: ein hoch-performanter Datentransfer

- UNICORE FTP (durchaus auch als Gegenpol zu Globus GridFTP)
- Performant (mehrere parallele TCP-Verbindungen)
- Client-Server, Server-Server
- Sicher, einfach einzurichten und zu verwenden
- Öffnen der benötigten Ports in der Firewall („FTP“ Protokol)
- Plattformunabhängig: Java
- Verfügbar mit UNICORE Version 6.4.1
- ... aber sicher werden Filetransfers ein Thema bleiben. Vielleicht sollte man sie ganz vermeiden?

- Aktuelle und zukünftige Anwendungen ...
  - erfordern die Verwendung immer mehr diversifizierter Systeme
  - verlangen ein ausgefeiltes Daten- und Rechte-Management
  - realisieren neue Szenarien jenseits des traditionellen Cluster- und Gridcomputing
- UNICORE bietet ...
  - eine solide Grundlage mit gleichzeitig hoher Erweiterbarkeit
- Aktuelle Themen sind unter anderem...
  - Verbesserungen und neue Funktionalität im Datenmanagement
  - Realisierung einer flexibleren Rechteverwaltung

- Danke
  - ICM: Krzysztof Benedyczak, Tomas Rekawek, ...
  - JSC: Sandra Bergmann, Jason Daivandy, Bastian Demuth, Björn Hagemeyer, Valentina Huber, Daniel Mallmann, Shiraz Memon, Michael Rambadt, Morris Riedel, Mathilde Romberg, Jedrzej Rybicki
  - ZIH: Richard Grunzke, Christian Loeschen
  - ...