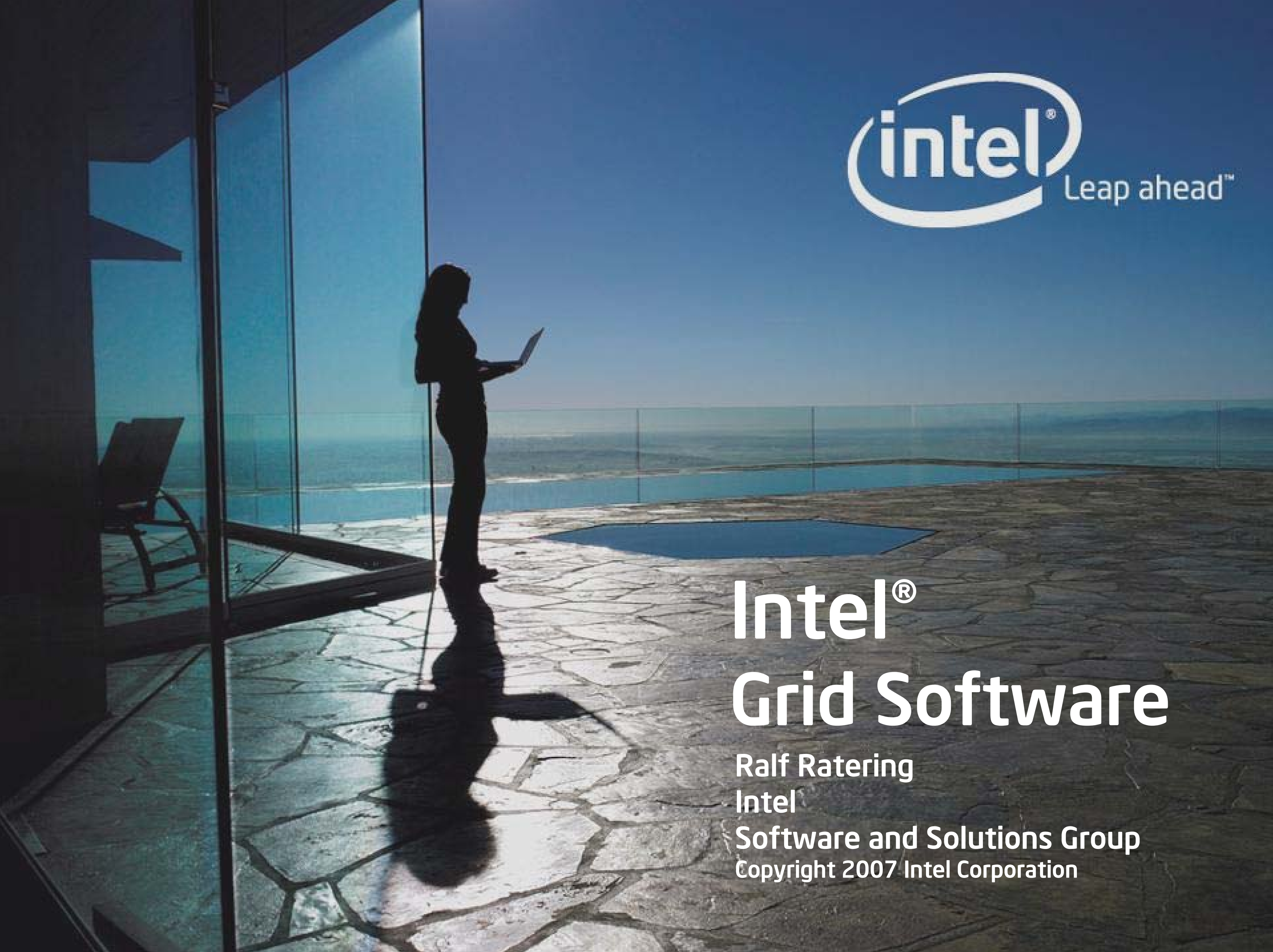


Leap ahead™

Intel® Grid Software

Ralf Ratering
Intel
Software and Solutions Group

Agenda

Part 1: Introduction

- Intel Software and the Grid
- Grid Programming Environment
- GPE Servers (Globus and Unicore 6)
- Advanced Features and Outlook

Part 2: Hands-on session

- Use pre-configured VMware image
- Run jobs using different GridBeans
- Implement your own Grid Client

Legal Information

INFORMATION IN THIS DOCUMENT IS PROVIDED IN CONNECTION WITH INTEL® PRODUCTS. NO LICENSE, EXPRESS OR IMPLIED, BY ESTOPPEL OR OTHERWISE, TO ANY INTELLECTUAL PROPERTY RIGHTS IS GRANTED BY THIS DOCUMENT. EXCEPT AS PROVIDED IN INTEL'S TERMS AND CONDITIONS OF SALE FOR SUCH PRODUCTS, INTEL ASSUMES NO LIABILITY WHATSOEVER, AND INTEL DISCLAIMS ANY EXPRESS OR IMPLIED WARRANTY, RELATING TO SALE AND/OR USE OF INTEL® PRODUCTS INCLUDING LIABILITY OR WARRANTIES RELATING TO FITNESS FOR A PARTICULAR PURPOSE, MERCHANTABILITY, OR INFRINGEMENT OF ANY PATENT, COPYRIGHT OR OTHER INTELLECTUAL PROPERTY RIGHT. INTEL PRODUCTS ARE NOT INTENDED FOR USE IN MEDICAL, LIFE SAVING, OR
LIFE SUSTAINING APPLICATIONS.

Intel may make changes to specifications and product descriptions at any time, without notice.

All products, dates, and figures specified are preliminary based on current expectations, and are subject to change without notice.

Intel, processors, chipsets, and desktop boards may contain design defects or errors known as errata, which may cause the product to deviate from published specifications. Current characterized errata are available on request.

Intel, the Intel logo, Intel. Leap ahead., the Intel. Leap ahead. Logo, Centrino, Core, Intel Inside, Itanium, Viiv, vPro and Xeon are trademarks or registered trademarks of Intel Corporation or its subsidiaries in the United States and other countries.

* Other names and brands may be claimed as the property of others.

Intel Software and the Grid

Intel Software & Solutions Group

No, Intel is not a chip company only…

Intel SSG

- Over 2000 people worldwide
- Linux, compilers, tools, Java*, .NET*, SOA, Grid, …
- Development, enabling, products, standards

Cluster Software & Grid Technology

- HPC, parallelism, cluster, grid computing
- Development of technology and tools, world-class expertise

Intel® Software Development Products

Intel® Compilers

- The best way to get application performance on Intel processors

Intel® VTune™ Performance Analyzers

- Identify bottlenecks in source code to increase performance or solve problems

Intel® Performance Libraries

- Highly optimized, thread-safe, multimedia and HPC math functions

Intel® Threading Tools

- Find threading errors and optimize threaded applications for maximum performance

Intel® Cluster Tools

- Create, analyze, optimize and deploy cluster-based applications

Intel Fortran Compilers
Intel C++ Compilers
Intel VTune Performance Analyzers
Intel Math Kernel Library
Intel Integrated Performance Primitives
Intel Thread Profile
Intel Thread Checker
Intel Cluster Tooki
Intel MPI Library
Intel Trace Analyzer and Collector

Intel® - XML Software Products

Intel® XSLT Accelerator 1.1

- XSLT 1.0 compliance, drop-in replacement for JAXP based XSLT processors
- Support for EXSLT extension functions and custom Java extension

Intel® XML Parsing Accelerator Beta 1.0

- W3C XML 1.0 Support
- C++ on Linux* only
- Efficient and scalable

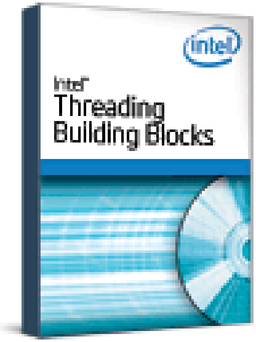
Intel® XML Software Suite Beta 1.0

- W3C XML Standards Compliant
- Java* and C/C++ environments on Linux* and Windows*
- XSLT, XML Parsing, Schema Validation, XPath navigation

more to come…

Latest Information: http://www.intel.com/software/xml/

Intel® Open Source Involvement
(incomplete, but Grid-related…)

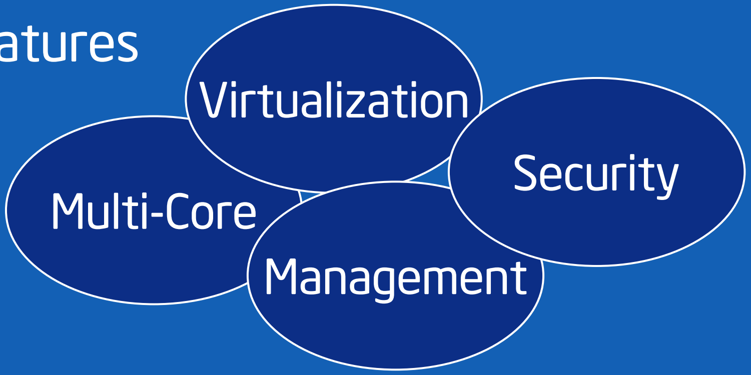


- **Threading Building blocks**
  - C++ runtime library that abstracts the low-level threading details necessary for optimal multi-core performance.
  - Uses common C++ templates and coding style to eliminate tedious threading implementation work
- **Apache Harmony**
  - A compatible, independent implementation of the Java SE 5 JDK under Apache License v2
  - A community-developed modular runtime (VM and class library) architecture.
- **Grid Programming Environment (GPE)**
  - Complete Grid Software Stack
  - Graphical user interfaces, Middleware Services, Execution Agents
  - Works with Globus Toolkit, Unicore and China Grid Support Platform

Grid Programming Environment (GPE)

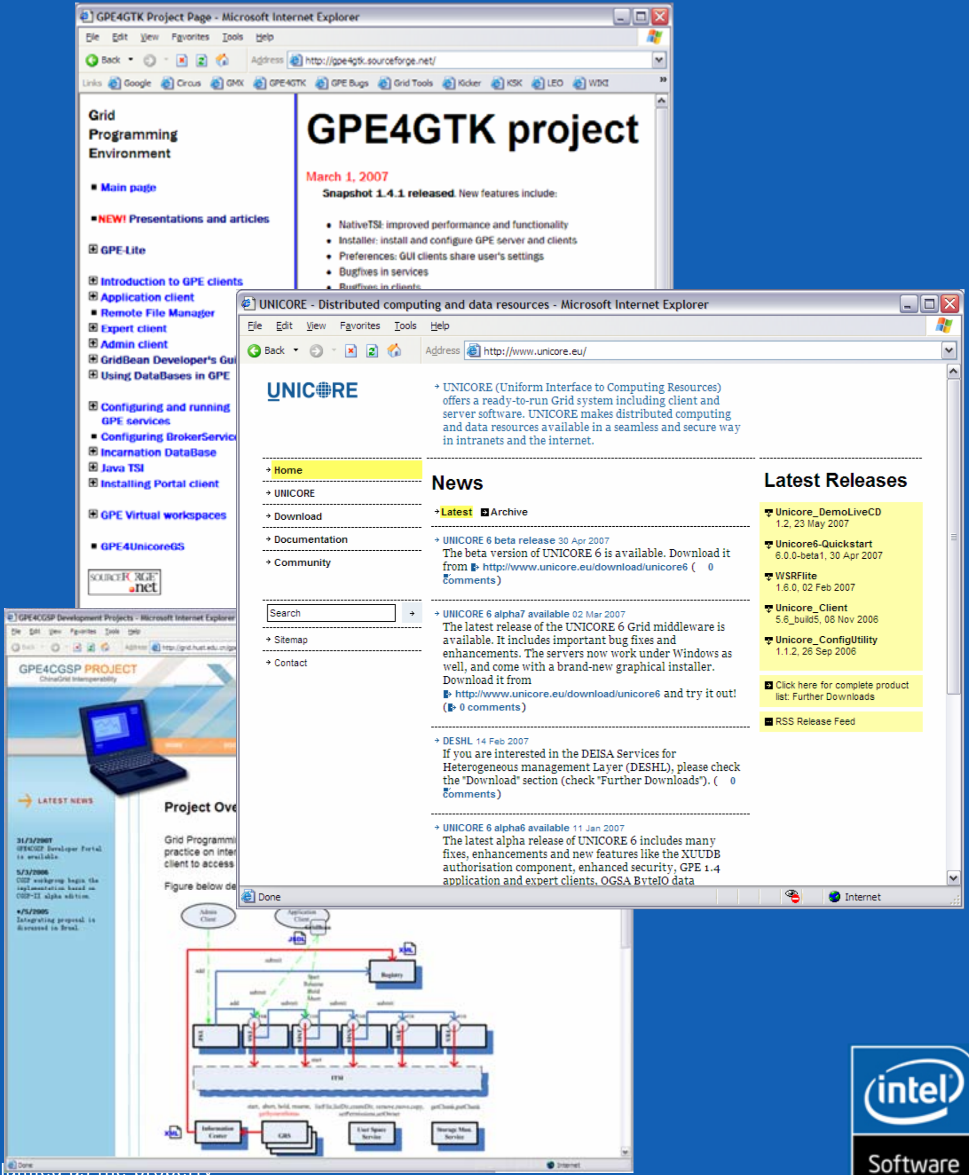
Grid Programming Environment

- **Open Source under BSD license**
  - SourceForge project (http://gpe4gtk.sourceforge.net)
- **Motivation**
  - Develop prototype Grid components
  - Simplify creation of Grid-enabled applications
  - Engage with Grid experts and users
  - Interact with ecosystem, standards bodies
  - Exploit Intel platform features



Current State: GPE 1.4 Released

- **Globus**
  - Main GPE development at GPE4GTK Sourceforge project
  - Complete Grid Software Stack
    - Clients, services, execution agents
- **UNICORE**
  - GPE integrated in UNICORE 6 release
  - Under consideration as future main platform for GPE
- **China Grid**
  - Prototype available for China Grid Supporting Platform



GPE4GTK Project Page - Microsoft Internet Explorer

http://gpe4gtk.sourceforge.net/

Grid Programming Environment

- Main page
- NEW! Presentations and articles
- GPE-Lite
- Introduction to GPE clients
- Application client
- Remote File Manager
- Expert client
- Admin client
- GridBean Developer's Gui
- Using DataBases in GPE
- Configuring and running GPE services
- Configuring BrokerServic
- Incarnation DataBase
- Java TSI
- Installing Portal client
- GPE Virtual workspaces
- GPE4UnicoreGS

GPE4GTK project

March 1, 2007

Snapshot 1.4.1 released. New features include:

- NativeTSI: improved performance and functionality
- Installer: install and configure GPE server and clients
- Preferences: GUI clients share user's settings
- Bugfixes in services
- Bugfixes in clients

UNICORE - Distributed computing and data resources - Microsoft Internet Explorer

http://www.unicore.eu/

UNICORE

UNICORE (Uniform Interface to Computing Resources) offers a ready-to-run Grid system including client and server software. UNICORE makes distributed computing and data resources available in a seamless and secure way in intranets and the internet.

- Home
- UNICORE
- Download
- Documentation
- Community
- Sitemap
- Contact

News

Latest Archive

UNICORE 6 beta release 30 Apr 2007
The beta version of UNICORE 6 is available. Download it from http://www.unicore.eu/download/unicore6 (0 comments)

UNICORE 6 alpha7 available 02 Mar 2007
The latest release of the UNICORE 6 Grid middleware is available. It includes important bug fixes and enhancements. The servers now work under Windows as well, and come with a brand-new graphical installer. Download it from http://www.unicore.eu/download/unicore6 and try it out! (0 comments)

DESHL 14 Feb 2007
If you are interested in the DEISA Services for Heterogeneous management Layer (DESHL), please check the "Download" section (check "Further Downloads"). (0 comments)

UNICORE 6 alpha6 available 11 Jan 2007
The latest alpha release of UNICORE 6 includes many fixes, enhancements and new features like the XUUDB authorisation component, enhanced security, GPE 1.4 application and expert clients, OGSA ByteIO data

Latest Releases

- Unicore_DemoLiveCD 1.2, 23 May 2007
- Unicore6-Quickstart 6.0.0-beta1, 30 Apr 2007
- WSRFlite 1.6.0, 02 Feb 2007
- Unicore_Client 5.6_build5, 08 Nov 2006
- Unicore_ConfigUtility 1.1.2, 26 Sep 2006

Click here for complete product list: Further Downloads

RSS Release Feed

GPE4CGSP Development Projects - Microsoft Internet Explorer

GPE4CGSP PROJECT
ChinaGrid Interoperability

LATEST NEWS

31/3/2007
GPE4CGSP Developer Portal is available

5/3/2006
CGSP workgroup begin the implementation based on CGSP-II alpha edition

*/5/2005
Integrating proposal is discussed in Brazil

Project Ove

Grid Programmi practice on inter client to access

Figure below de

Implementing and Influencing Standards with GPE

Open Grid Forum (OGF)

- JSDL (Job Submission Description Language)
- OGSA (Open Grid Services Architecture)
  - BES (Basic Execution Services)
  - DMI (Data Movement Interface)
- ByteIO (File Transfer and Streaming)

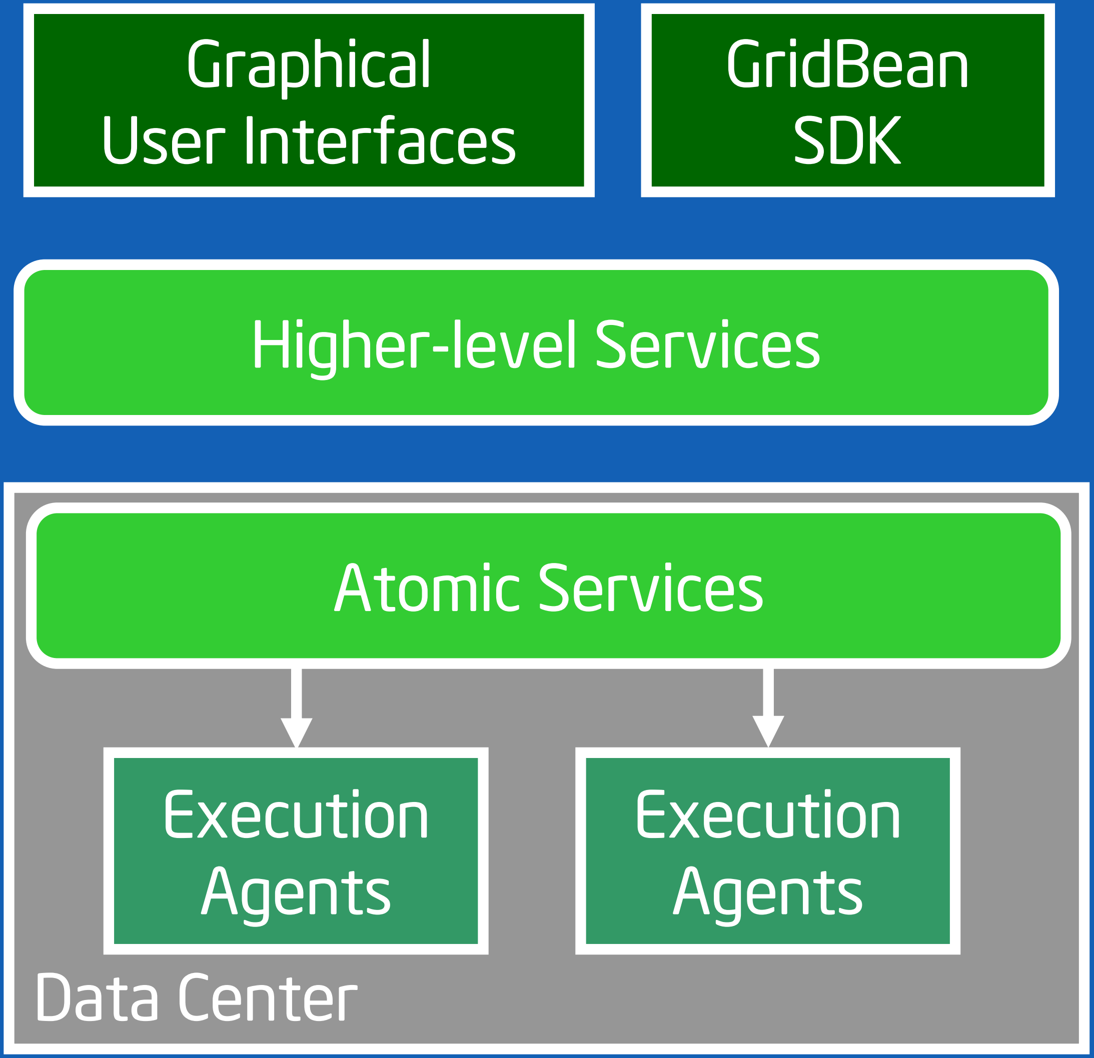
OASIS

- WSRF (Web Services Resource Framework)
- WSN (Web Services Notification)
- WSS (Web Services Security)
- WSBPEL (WS-Business Process Execution Language)
- SAML (Security Assertion Markup Language)

W3C

- WS-Addressing, SOAP, WSDL, XML, etc.

GPE Components


Graphical User Interfaces

GridBean SDK

Higher-level Services

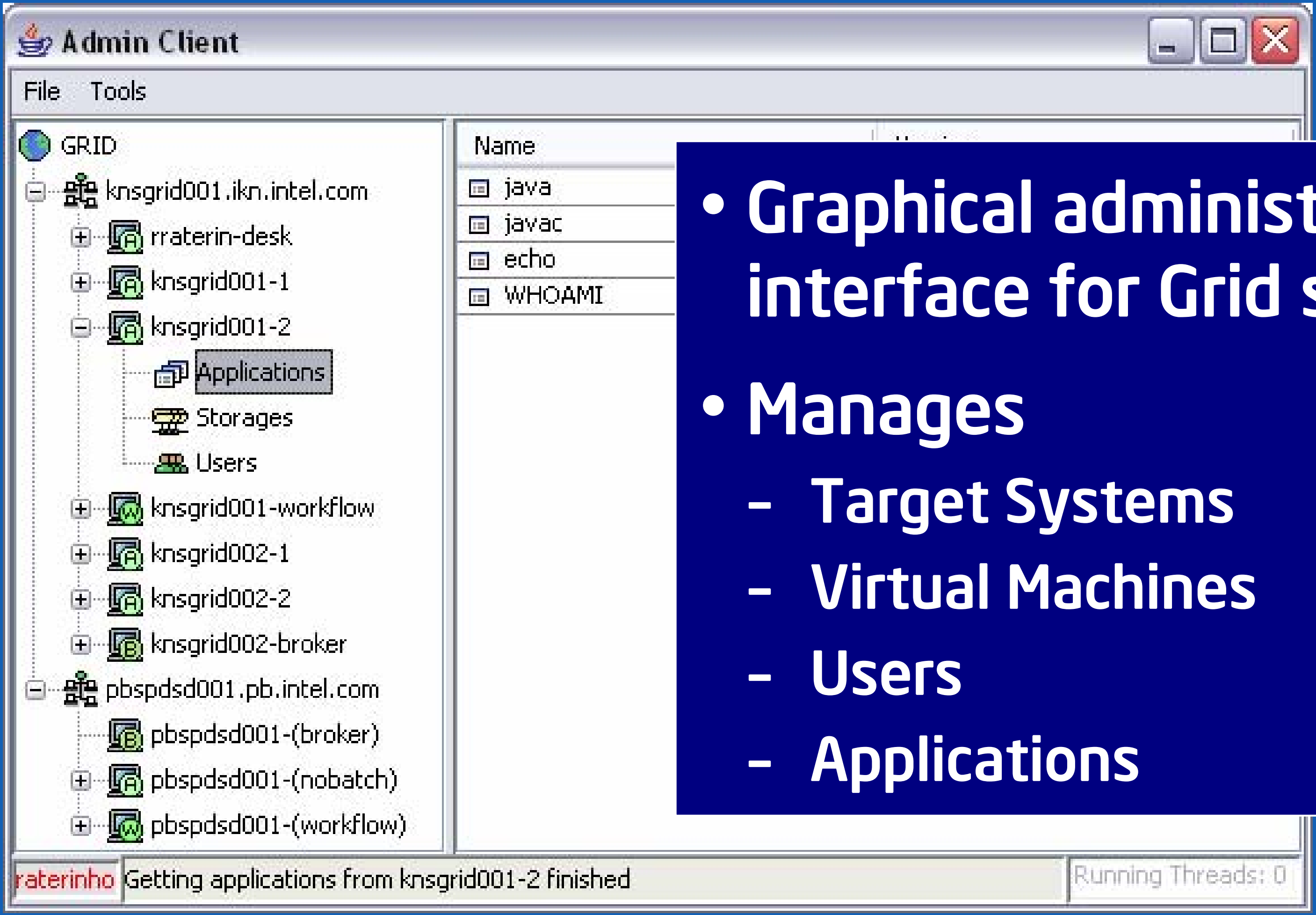
Atomic Services

Execution Agents

Execution Agents

Data Center

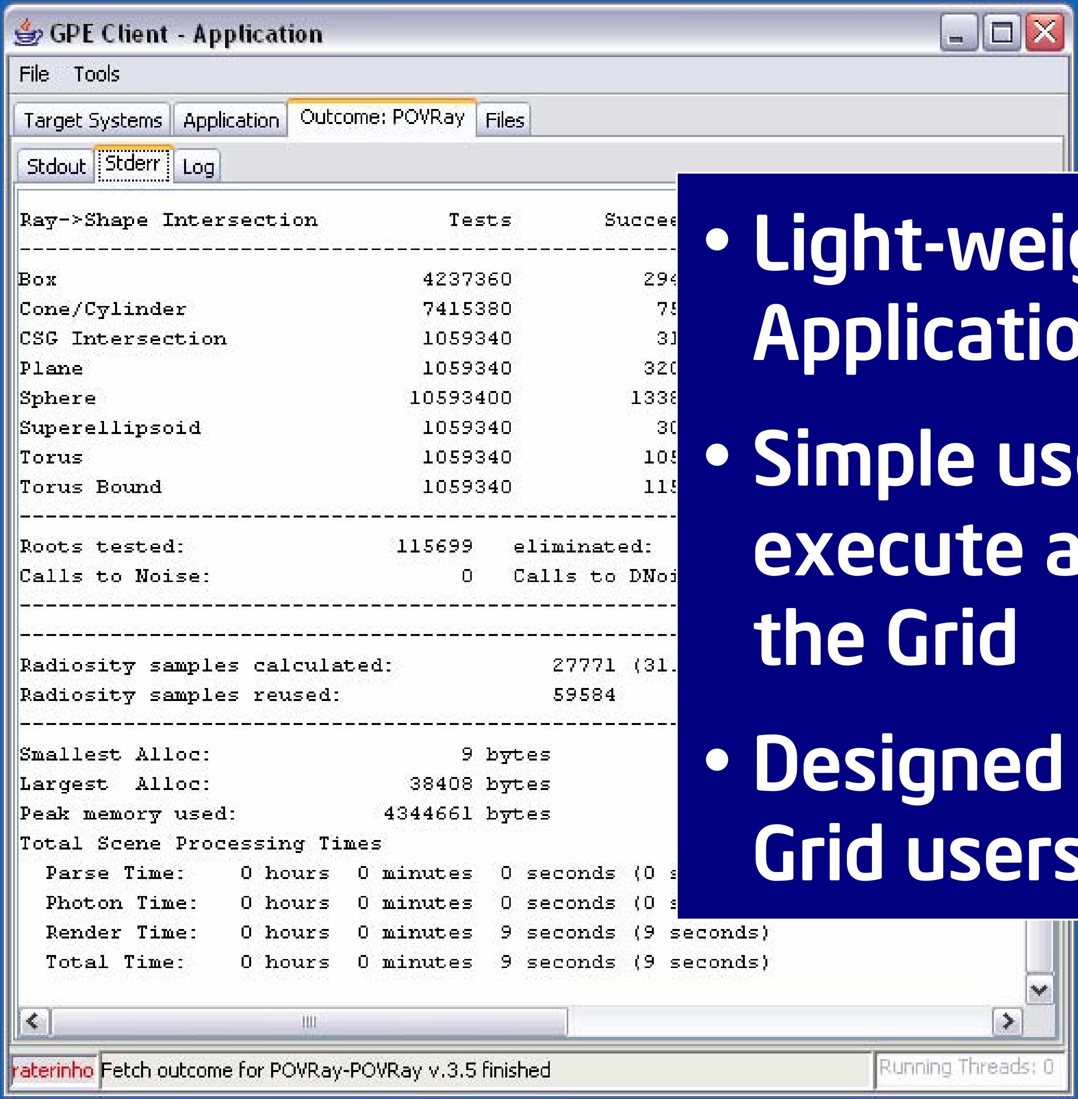
GPE Admin Client



Getting applications from knsgrid001-2 finished

- Graphical administration interface for Grid systems
- Manages
  - Target Systems
  - Virtual Machines
  - Users
  - Applications

GPE Application Client

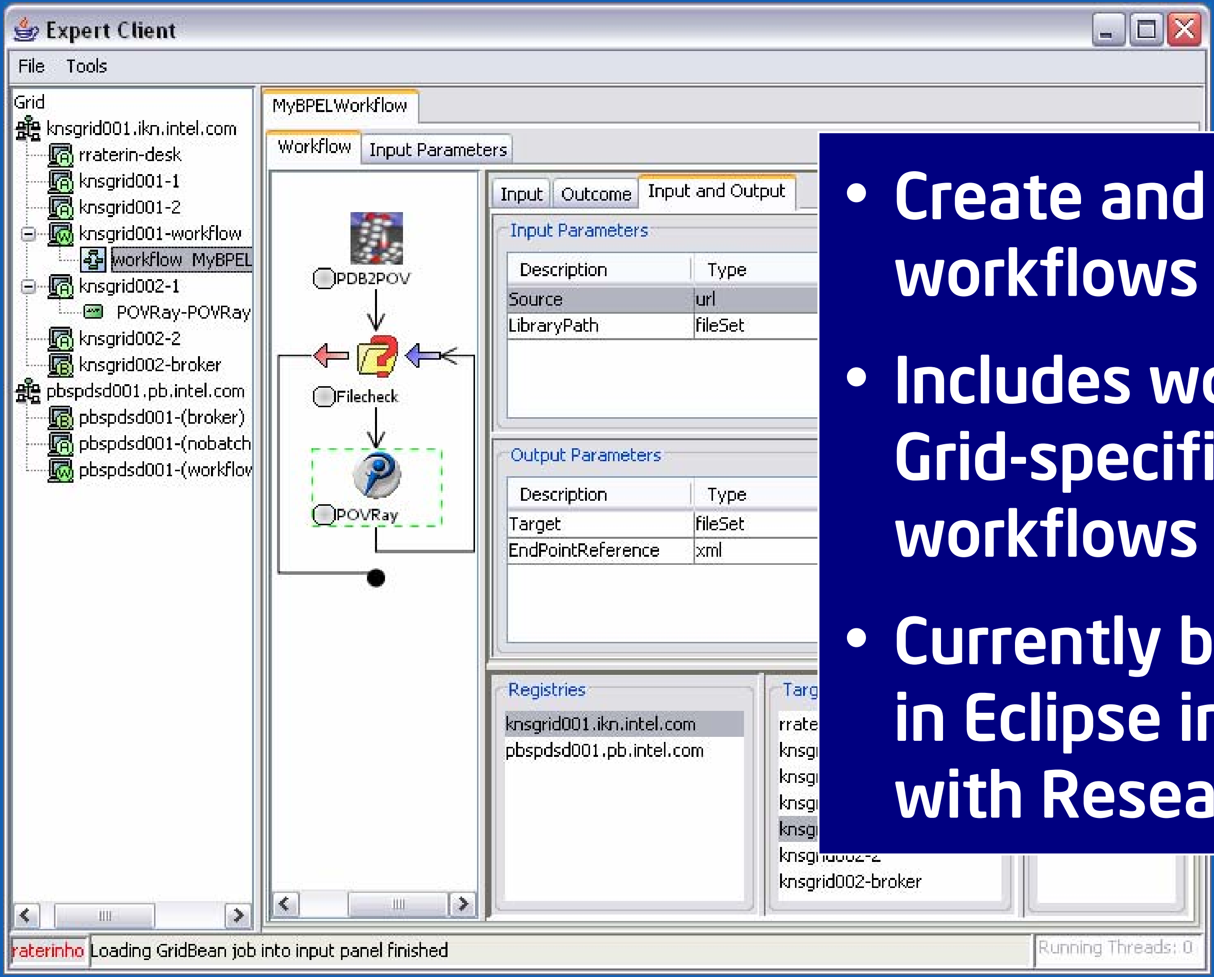


```
Ray->Shape Intersection          Tests       Succe
--------------------------------------------------
Box                             4237360        29
Cone/Cylinder                   7415380         7
CSG Intersection                1059340         3
Plane                           1059340        32
Sphere                         10593400       133
Superellipsoid                  1059340         3
Torus                           1059340        10
Torus Bound                     1059340        11
--------------------------------------------------
Roots tested:              115699   eliminated:
Calls to Noise:                 0   Calls to DNoi
--------------------------------------------------
--------------------------------------------------
Radiosity samples calculated:           27771 (31.
Radiosity samples reused:               59584
--------------------------------------------------
Smallest Alloc:                 9 bytes
Largest  Alloc:             38408 bytes
Peak memory used:         4344661 bytes
Total Scene Processing Times
  Parse Time:    0 hours  0 minutes  0 seconds (0 s
  Photon Time:   0 hours  0 minutes  0 seconds (0 s
  Render Time:   0 hours  0 minutes  9 seconds (9 seconds)
  Total Time:    0 hours  0 minutes  9 seconds (9 seconds)
```

raterinho Fetch outcome for POVRay-POVRay v.3.5 finished

- Light-weight Java Application
- Simple user interface to execute applications on the Grid
- Designed for standard Grid users

GPE Expert Client



Input Parameters

Description	Type
Source	url
LibraryPath	fileSet

Output Parameters

Description	Type
Target	fileSet
EndPointReference	xml

Registries

knsgrid001.ikn.intel.com
pbspdsd001.pb.intel.com

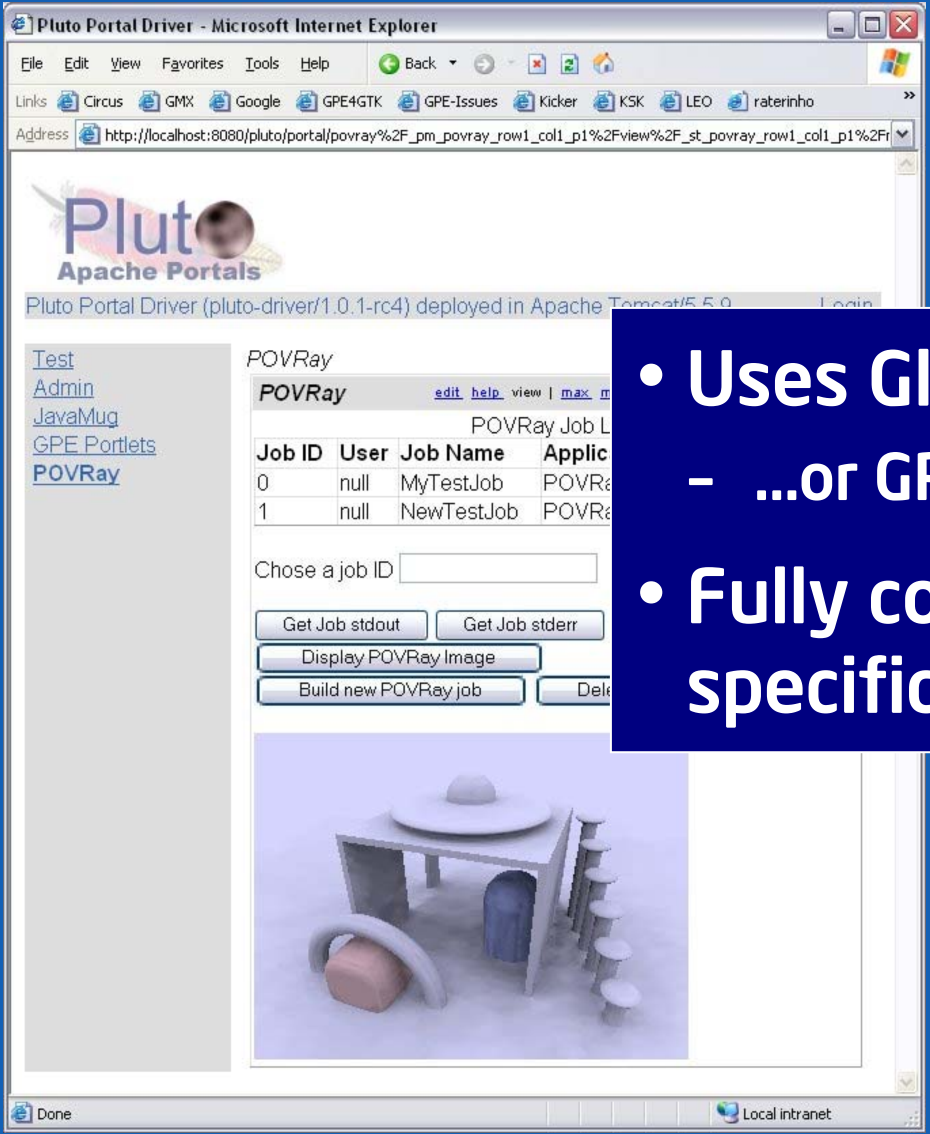
Targ

rrate
knsg
knsg
knsg
knsg
knsgrid002-2
knsgrid002-broker

raterinho Loading GridBean job into input panel finished

- Create and manage complex workflows on the Grid
- Includes workflow editor for Grid-specific BPEL-workflows
- Currently being integrated in Eclipse in cooperation with Research Center Juelich

GPE Portal



Pluto Portal Driver - Microsoft Internet Explorer

File Edit View Favorites Tools Help
Back
Links Circus GMX Google GPE4GTK GPE-Issues Kicker KSK LEO raterinho
Address

http://localhost:8080/pluto/portal/povray%2F_pm_povray_row1_col1_p1%2Fview%2F_st_povray_row1_col1_p1%2Fr

Pluto Portal Driver (pluto-driver/1.0.1-rc4) deployed in Apache Tomcat/5.5.9 Login

POVRay

POVRay edit help view | max m

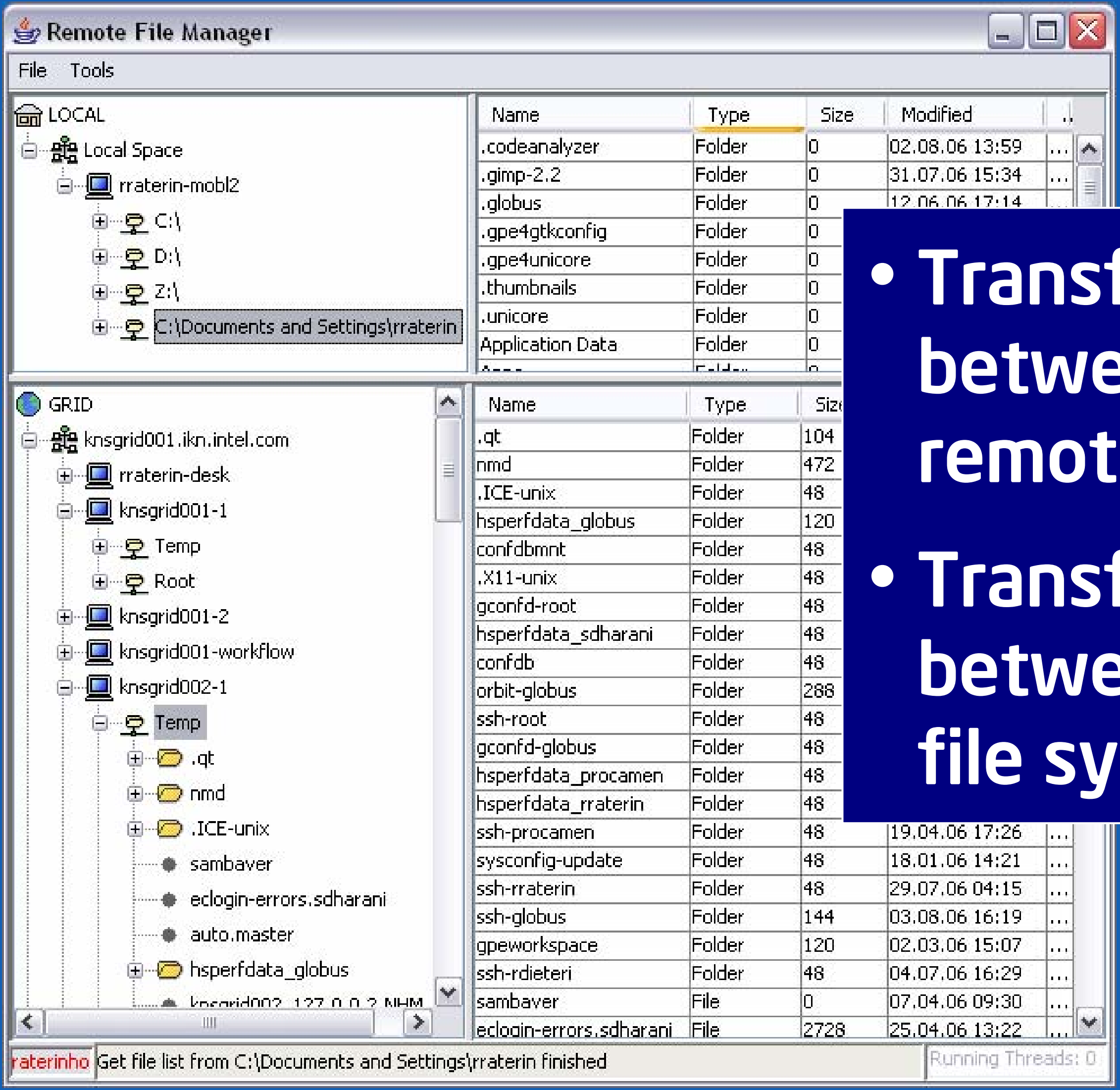
POVRay Job L

Job ID	User	Job Name	Applic
0	null	MyTestJob	POVRa
1	null	NewTestJob	POVRa

Chose a job ID

- Uses Globus MyProxy
  - …or GPE VO Credential Service
- Fully compliant to Portlet specification JSR 168

GPE Remote File Manager



Remote File Manager

File Tools

LOCAL
Local Space
rraterin-mobl2
C:\
D:\
Z:\
C:\Documents and Settings\rraterin

Name	Type	Size	Modified	..
.codeanalyzer	Folder	0	02.08.06 13:59	...
.gimp-2.2	Folder	0	31.07.06 15:34	...
.globus	Folder	0	12.06.06 17:14	...
.gpe4gtkconfig	Folder	0		
.gpe4unicore	Folder	0		
.thumbnails	Folder	0		
.unicore	Folder	0		
Application Data	Folder	0		

GRID
knsgrid001.ikn.intel.com
rraterin-desk
knsgrid001-1
Temp
Root
knsgrid001-2
knsgrid001-workflow
knsgrid002-1
Temp
.qt
nmd
.ICE-unix
sambaver
eclogin-errors.sdharani
auto.master
hsperfdata_globus
knsgrid002_127.0.0.2 NHM

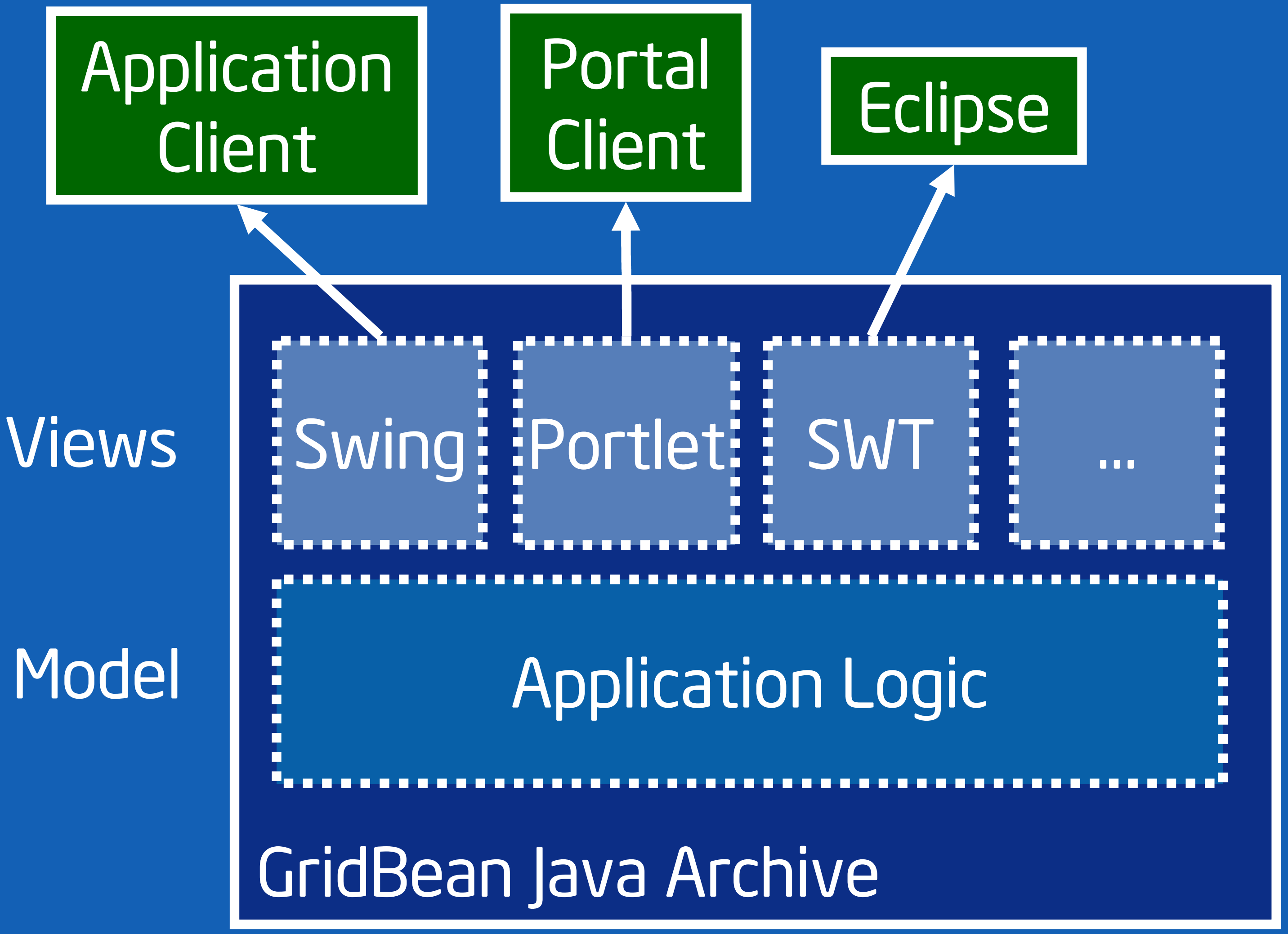
Name	Type	Size		
.qt	Folder	104		
nmd	Folder	472		
.ICE-unix	Folder	48		
hsperfdata_globus	Folder	120		
confdbmnt	Folder	48		
.X11-unix	Folder	48		
gconfd-root	Folder	48		
hsperfdata_sdharani	Folder	48		
confdb	Folder	48		
orbit-globus	Folder	288		
ssh-root	Folder	48		
gconfd-globus	Folder	48		
hsperfdata_procamen	Folder	48		
hsperfdata_rraterin	Folder	48		
ssh-procamen	Folder	48	19.04.06 17:26	...
sysconfig-update	Folder	48	18.01.06 14:21	...
ssh-rraterin	Folder	48	29.07.06 04:15	...
ssh-globus	Folder	144	03.08.06 16:19	...
gpeworkspace	Folder	120	02.03.06 15:07	...
ssh-rdieteri	Folder	48	04.07.06 16:29	...
sambaver	File	0	07.04.06 09:30	...
eclogin-errors.sdharani	File	2728	25.04.06 13:22	...

raterinho Get file list from C:\Documents and Settings\rraterin finished

Running Threads: 0

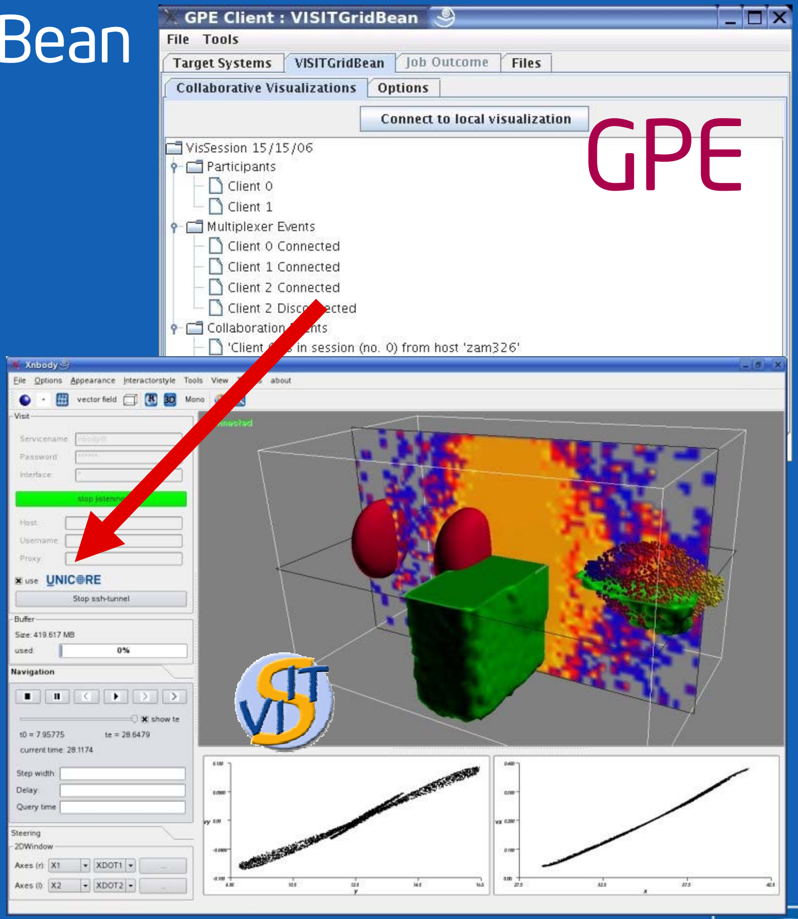
- Transfer files between local and remote file systems
- Transfer files between remote file systems

How do GridBeans work?

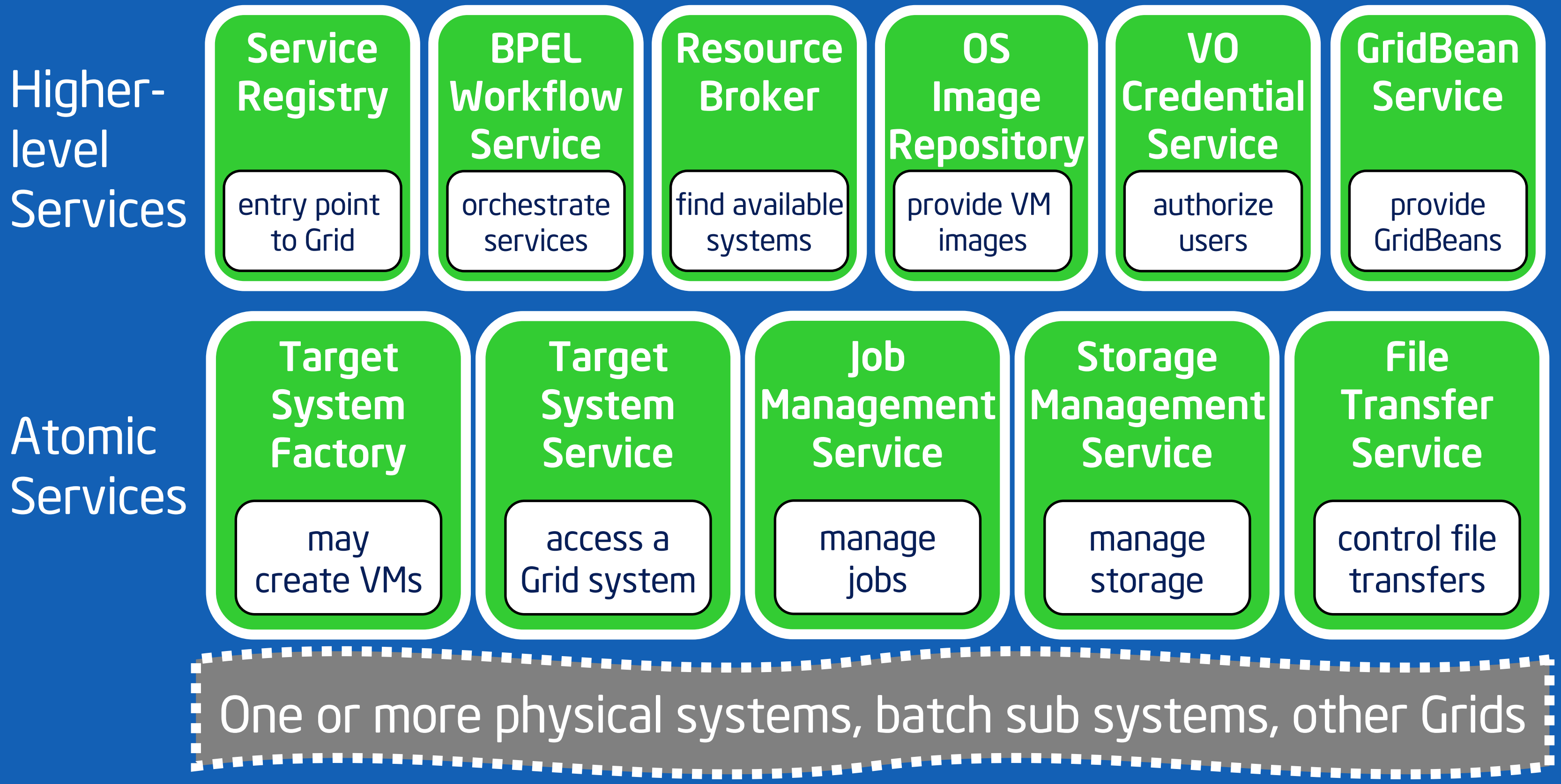


GridBean Examples

- GridBeans fo arbitrary applications
  - Generic GridBean, ScriptGridBean
- Application GridBeans
  - POVRay, PDBSearch, Ansys CFX*, Nastran*, Fluent*, Gaussian*, etc.
- Complex GridBeans
  - Collaborative Online Visualization and Steering (COVS) developed at Research Center Juelich



GPE Services



Higher-level Services

Service Registry — entry point to Grid

BPEL Workflow Service — orchestrate services

Resource Broker — find available systems

OS Image Repository — provide VM images

VO Credential Service — authorize users

GridBean Service — provide GridBeans

Atomic Services

Target System Factory — may create VMs

Target System Service — access a Grid system

Job Management Service — manage jobs

Storage Management Service — manage storage

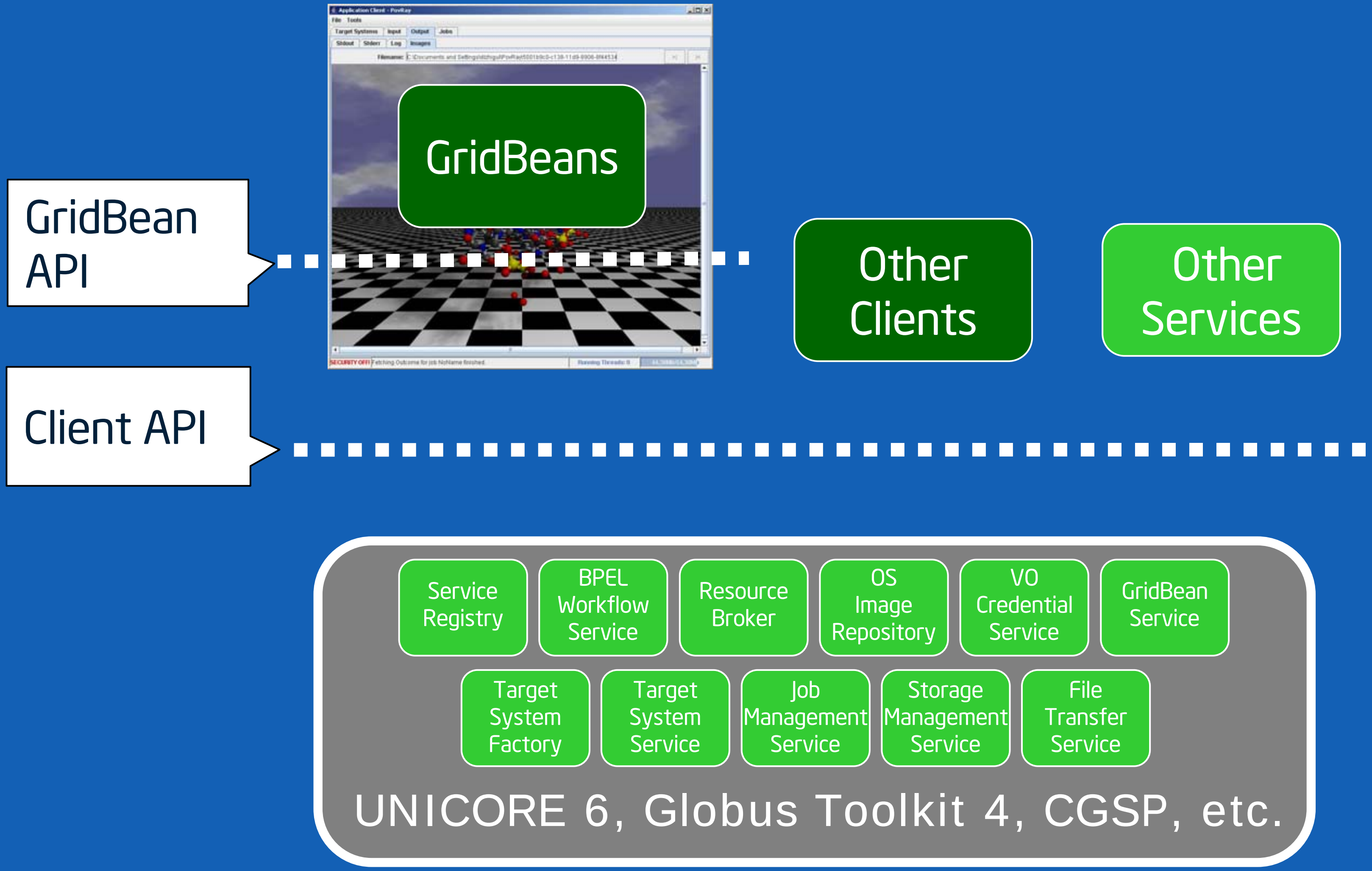
File Transfer Service — control file transfers

One or more physical systems, batch sub systems, other Grids

Atomic Services provide abstraction for underlying Hardware, Software, Batch Sub Systems, File transfer protocols etc.

- Based on OASIS Web Service Resource Framework (WSRF)

GPE Programming



GridBeans

GridBean API

Other Clients

Other Services

Client API

Service Registry

BPEL Workflow Service

Resource Broker

OS Image Repository

VO Credential Service

GridBean Service

Target System Factory

Target System Service

Job Management Service

Storage Management Service

File Transfer Service

UNICORE 6, Globus Toolkit 4, CGSP, etc.

Let’s see it in action…

A closer look at the server side…

GPE for Globus Toolkit 4 (*GPE4GTK*)

- **GPE services are hosted in GT4 container**
  - Only WS-Core package is needed
    - Bundled with GPE Server installer
    - GPE services are automatically deployed
    - Optionally GridFTP can be used for file transfers
- **Use underlying Globus security infrastructure**
  - GPE security provider creates proxy certificates on the client side
- **GPE provides**
  - Atomic Service implementations
  - Advanced TSI as replacement for GRAM
  - Easy-to-use installation packages

Supported File Transfer Protocols

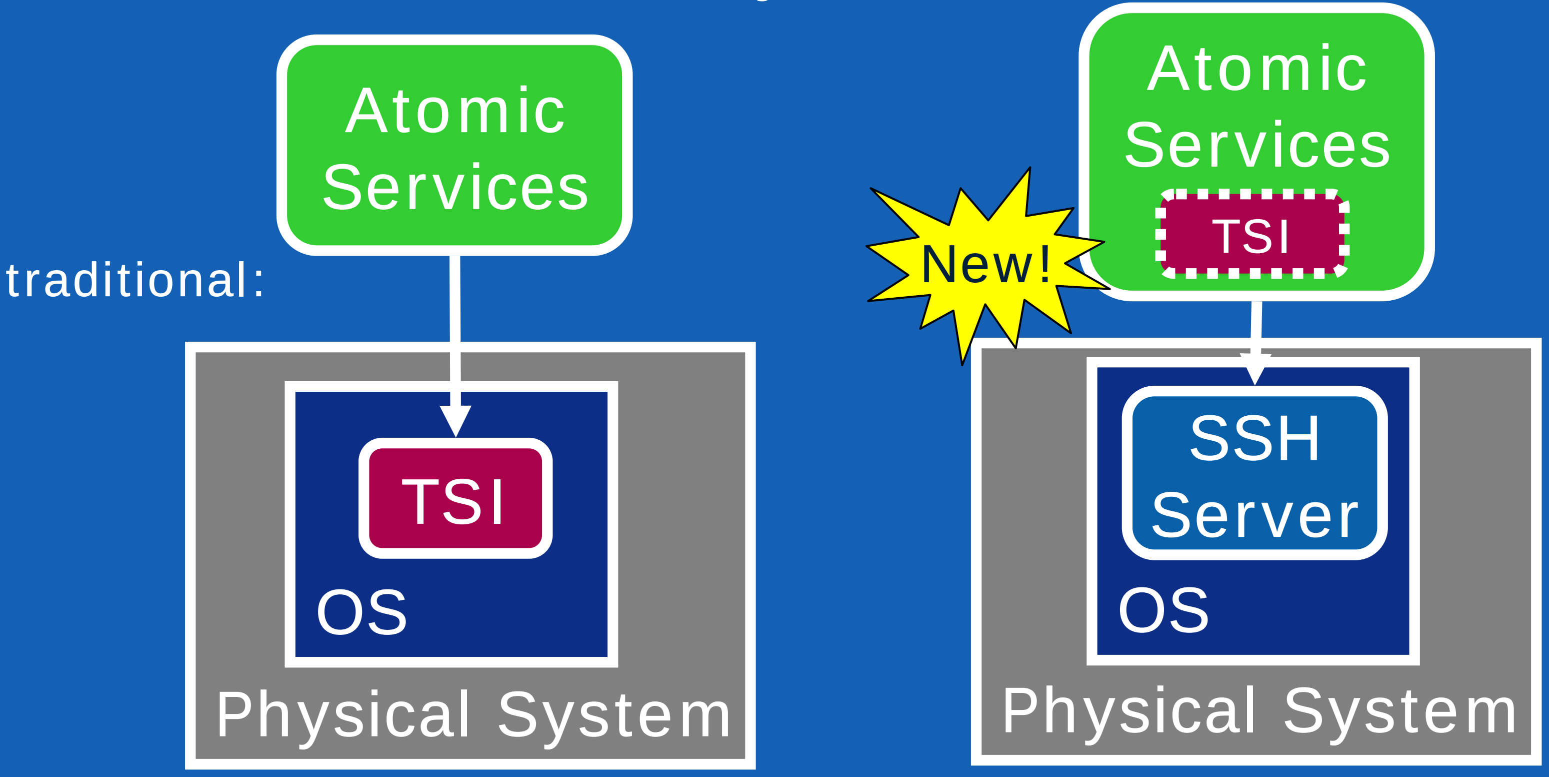
- **Random ByteIO**
  - Specification 1.0 (Oct 28, 2005) from OGF-WG
  - „plain“ flavor, chunks inlined in SOAP message
  - Advantages
    - reasonable performance
    - allows arbitrary access to file contents
    - usually no firewall limitations
- **GridFTP**
  - GridFTPFileTransferService allows initiating GridFTP transfers
  - good performance, but firewall limitations
- **scp**
  - good performance, no additional components needed

Software

Execution Agents

Target System Interface (TSI)

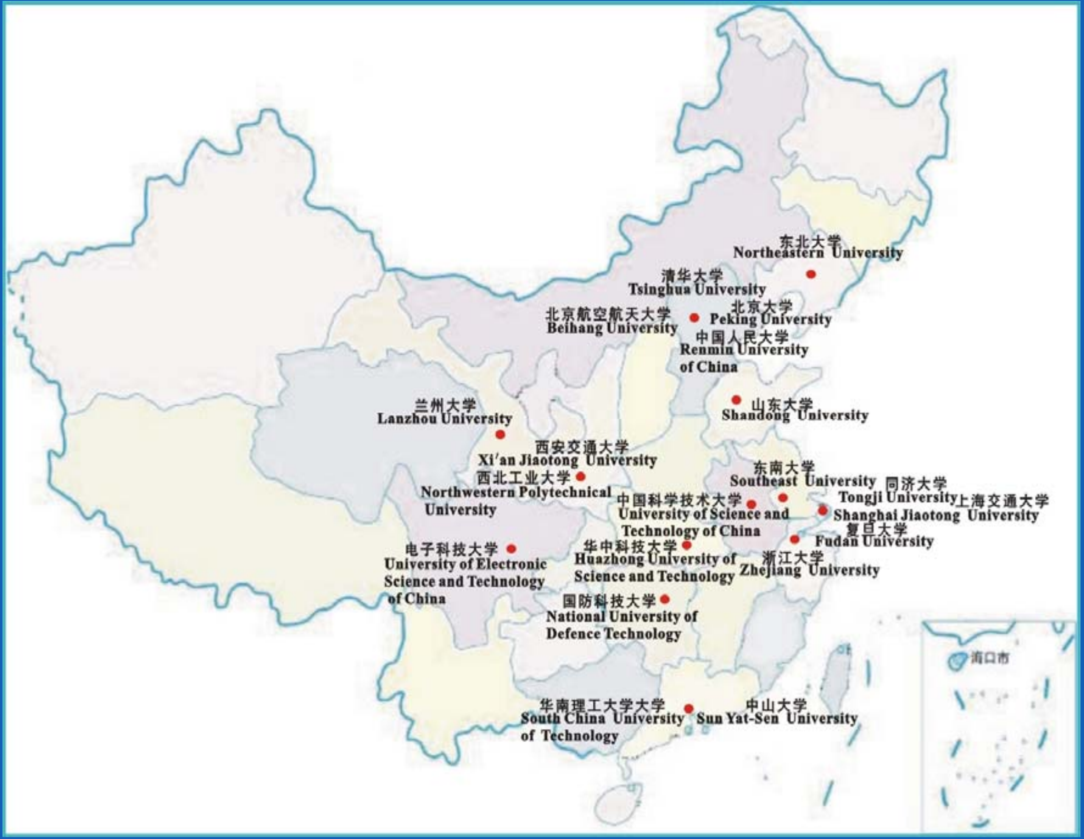
- Basic implementation in Java
- Native C implementation for performance
- SSH-version for easy access



GPE for ChinaGrid

Funded by Chinese Ministry of Education

- First phase from 2003 to 2005 connecting 20 key universities
- 15 Tflop/s computing power, 150TB storage
- Phase 2 connecting 100 universities has just started...



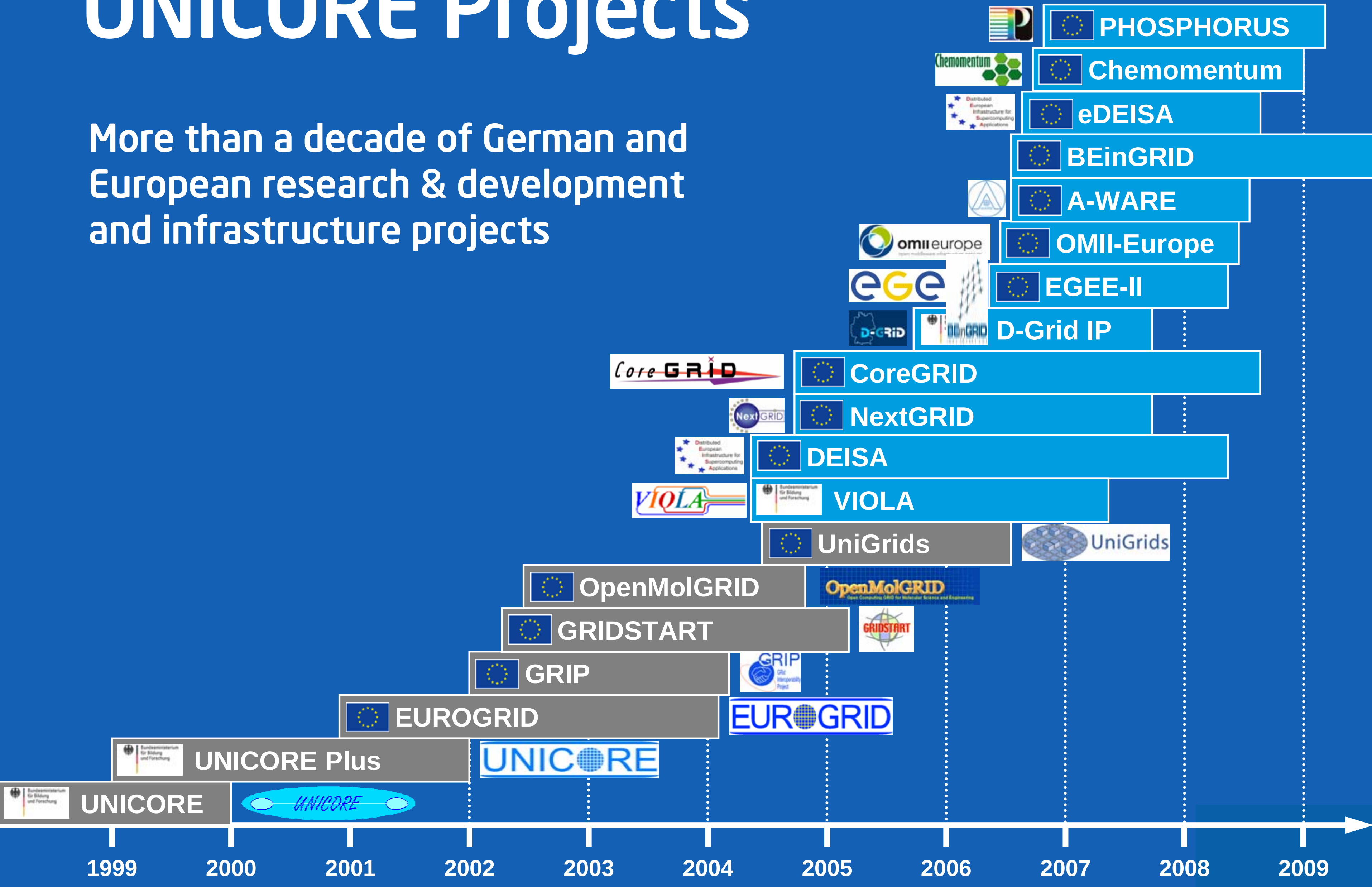
Cooperation with Intel 2005-2007

- Integrate GPE with CGSP (China Grid Support Platform)
- Allows using GPE clients and GridBeans in ChinaGrid
- Interoperability between China Grid, Unicore and Globus

Prototype for GPE1.4 integration available since August 2007

UNICORE Projects

More than a decade of German and European research & development and infrastructure projects



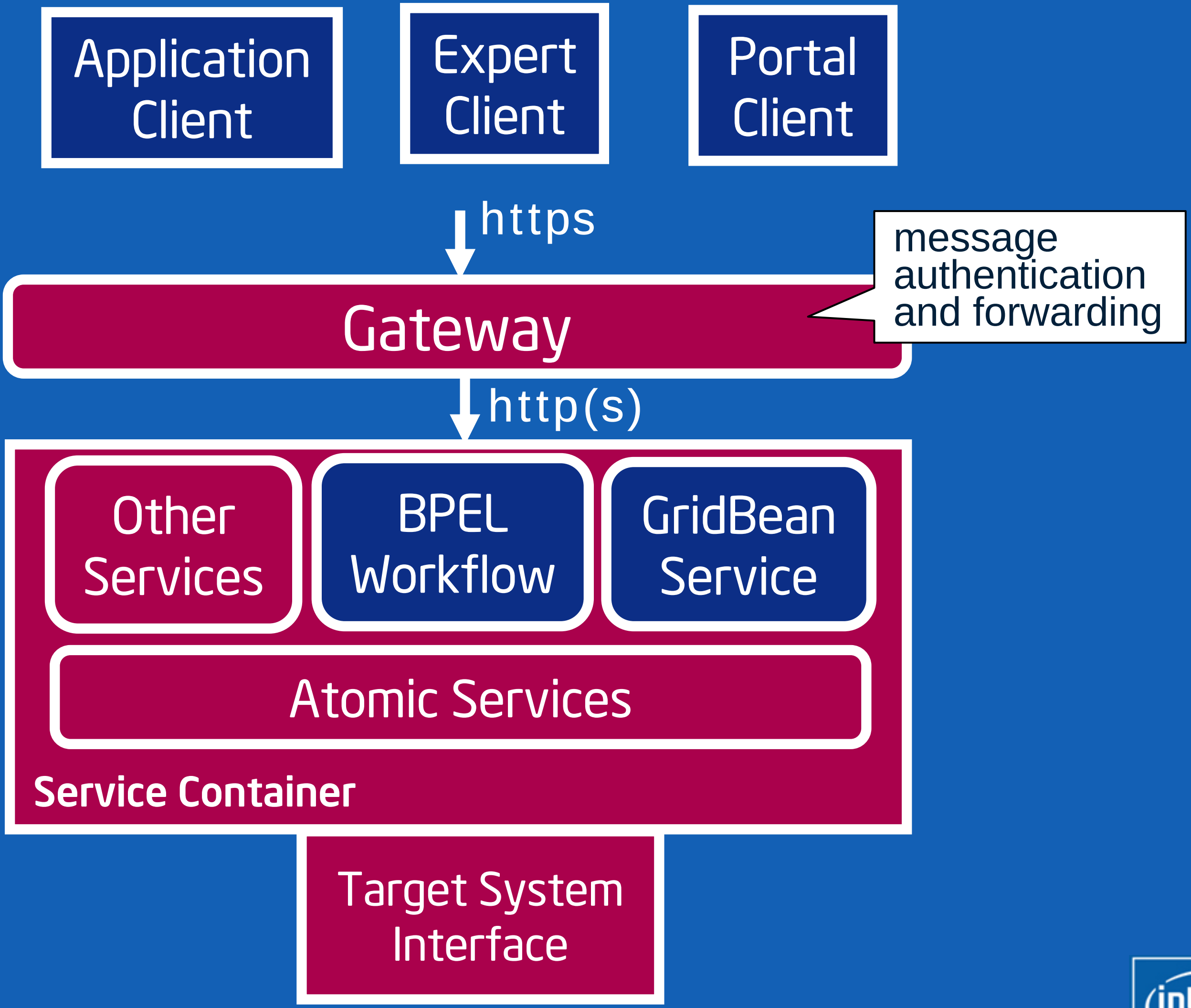
Source: Achim Streit, Research Center Juelich

UNICORE 6 Status

- **UNICORE 6 released August 14, 2007**
  - Includes GPE 1.4.3
- **Functionality**
  - Web service /WSRF core
  - Security (message signing, trust delegation)
  - Basic services (registry, jobs, files)
  - GPE Application client
  - GPE BPEL Workflow engine + Expert client
  - Command-line client and scripting tools

UNICORE 6 Components

■ =GPE



UNICORE 6 Service Container

- **WSRFlite**
- **State of the art web services tools:**
  - XFire, XmlBeans, Jetty
- **Specifications**
  - WSRF (version 1.2 final)
  - WS ServiceGroup
  - WS BaseNotification
  - WS-I „plain“ web services
- **Characteristics:**
  - High performance
  - Ease of configuration
  - Developer friendly, embeddable, flexible

Source: Bernd Schuller,
Research Center Juelich

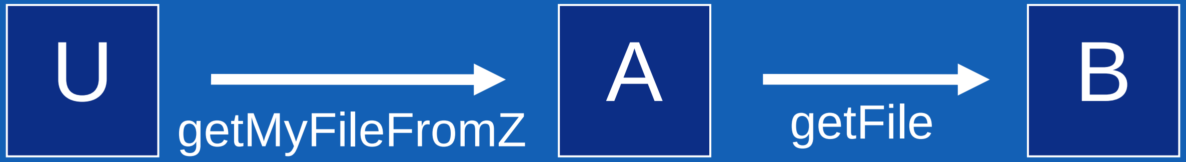
UNICORE 6 Security

- **Basic mechanism**
  - Users and servers are identified by X.509 certificates
  - Communication between components secured by client authenticated SSL/TLS
  - Messages contain additional security information in the SOAP header
- **Web service layer**
  - Extract security info from message
    - User: who is the originator of the message
    - Consignor: who sent the message
    - Trust delegation tokens, digital signature, …
  - Lookup security attributes in user data base (XUUDB)
    - Input: user's identity (certificate or DN)
    - Output: Unix login, role, projects, …
  - Check compliance to standard policies
    - Important operations require a digital signature

Source: Bernd Schuller,
Research Center Juelich

UNICORE 6 Trust Delegation

- **Common problem: user needs to delegate rights to a service**
  - example: file transfer

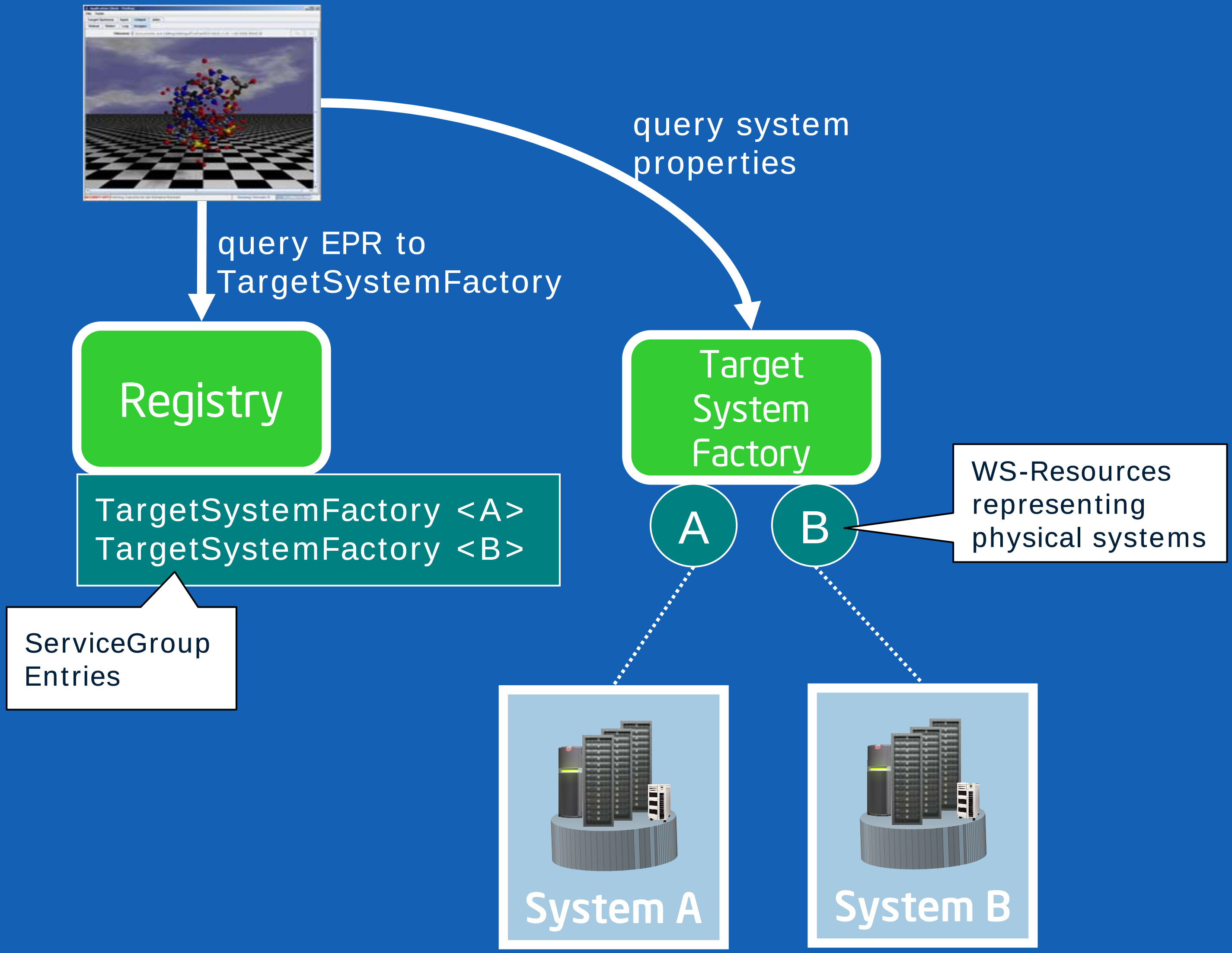


- **Clients can send a SAML trust delegation token**
  - „User U trusts Server A“
  - Digitally signed by U
- **Server A adds this token to his messages to Server B**
- **Server B checks validity and treats request „as if“ sent by User U.**

Source: Bernd Schuller,
Research Center Juelich

What's happening behind the scenes?

Registry as Entry Point



query system properties

query EPR to TargetSystemFactory

Registry

Target System Factory

TargetSystemFactory <A>
TargetSystemFactory <B>

ServiceGroup Entries

A

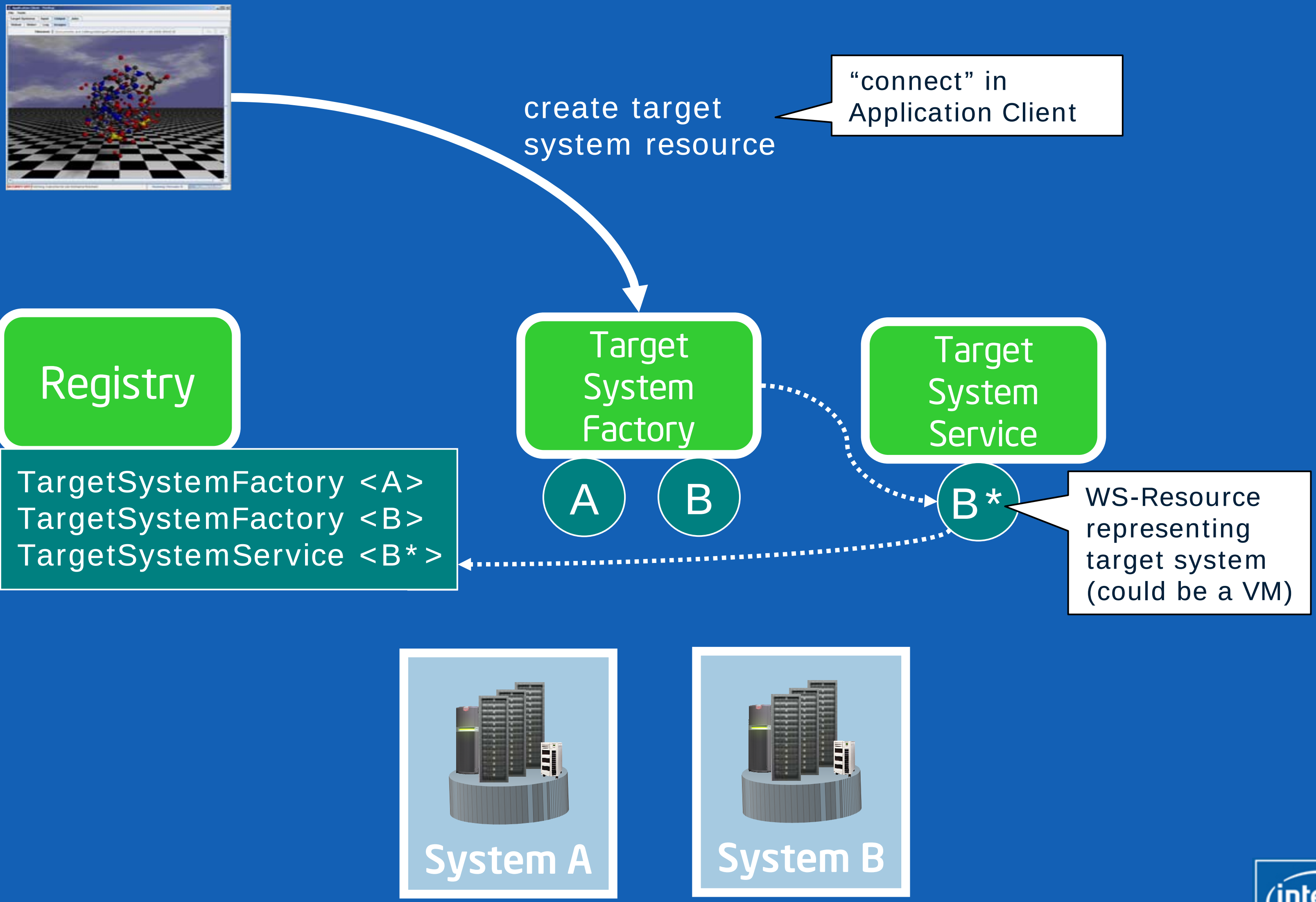
B

WS-Resources representing physical systems

System A

System B

Creating a Target System Resource



create target system resource

“connect” in Application Client

Registry

TargetSystemFactory <A>
TargetSystemFactory <B>
TargetSystemService <B*>

Target System Factory

A

B

Target System Service

B*

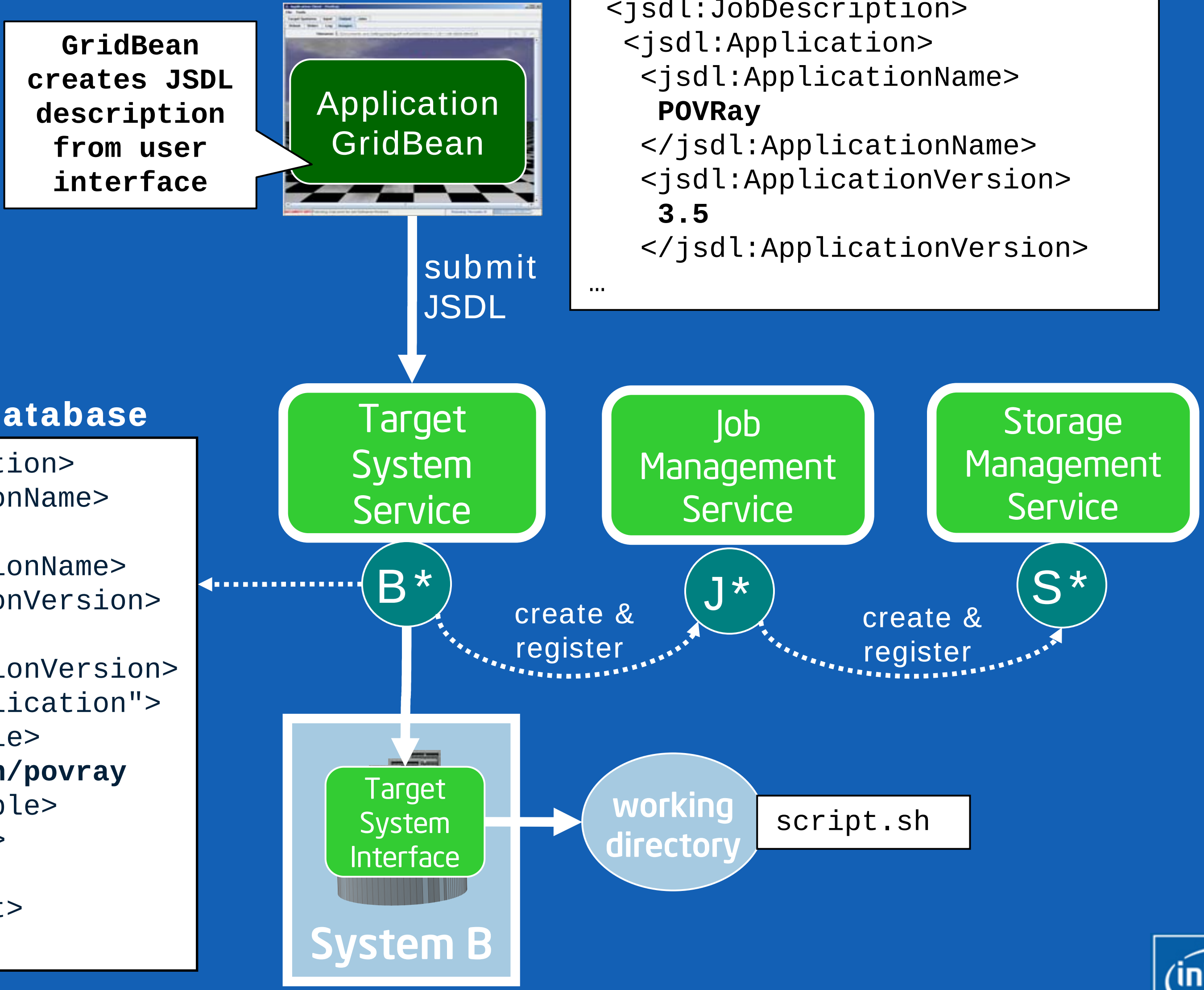
WS-Resource representing target system (could be a VM)

System A

System B

Submitting a Job

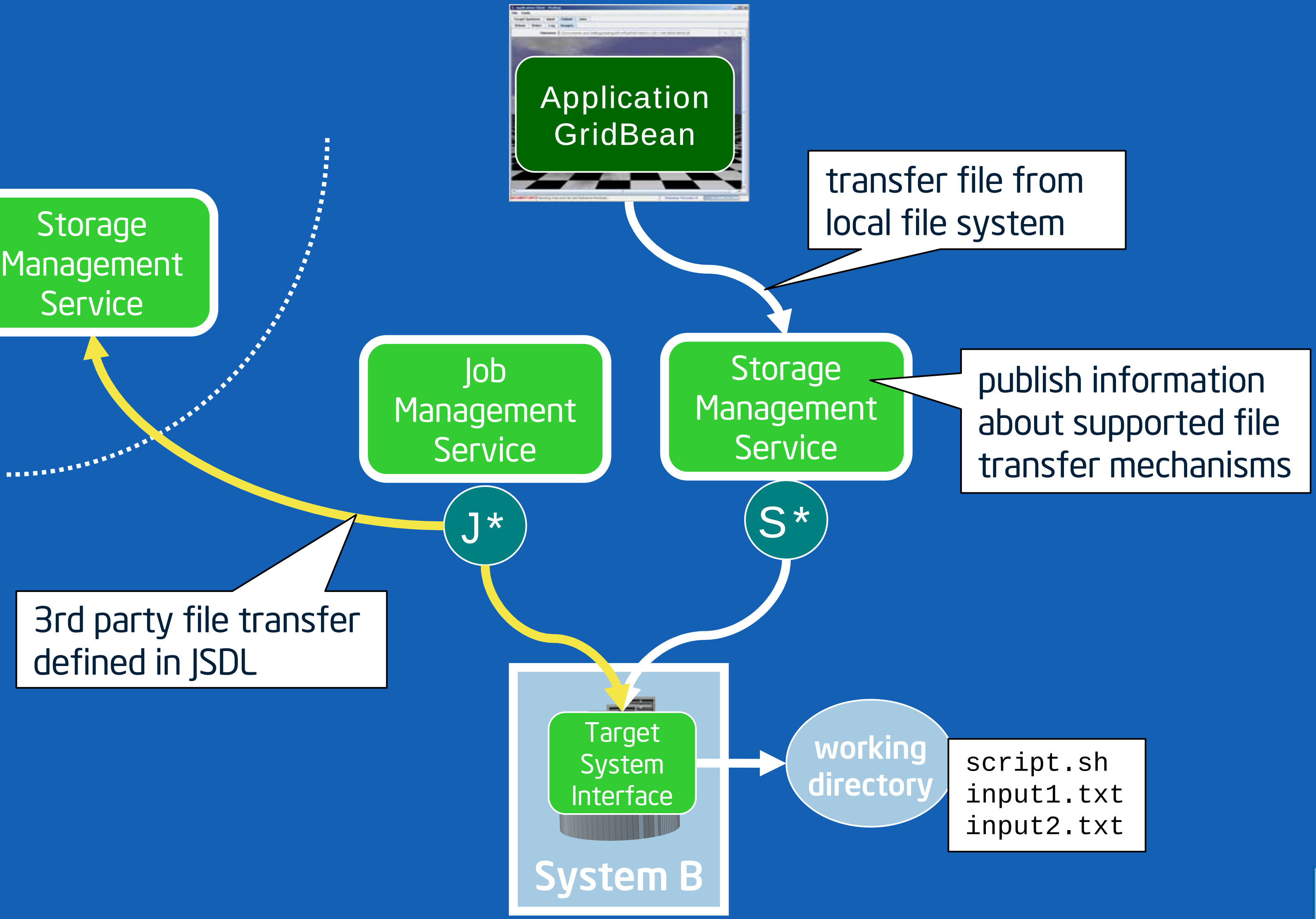
```
<jsdl:JobDefinition>
 <jsdl:JobDescription>
  <jsdl:Application>
   <jsdl:ApplicationName>
    POVRay
   </jsdl:ApplicationName>
   <jsdl:ApplicationVersion>
    3.5
   </jsdl:ApplicationVersion>
…
```



Incarnation Database

```
<idb:IDBApplication>
 <idb:ApplicationName>
  POVRay
 </idb:ApplicationName>
 <idb:ApplicationVersion>
  3.5
 </idb:ApplicationVersion>
 <jsdl:POSIXApplication">
 <jsdl:Executable>
  /usr/local/bin/povray
 </jsdl:Executable>
 <jsdl:Argument>
  +I$SOURCE?
 </jsdl:Argument>
…
```

Transferring Input Files



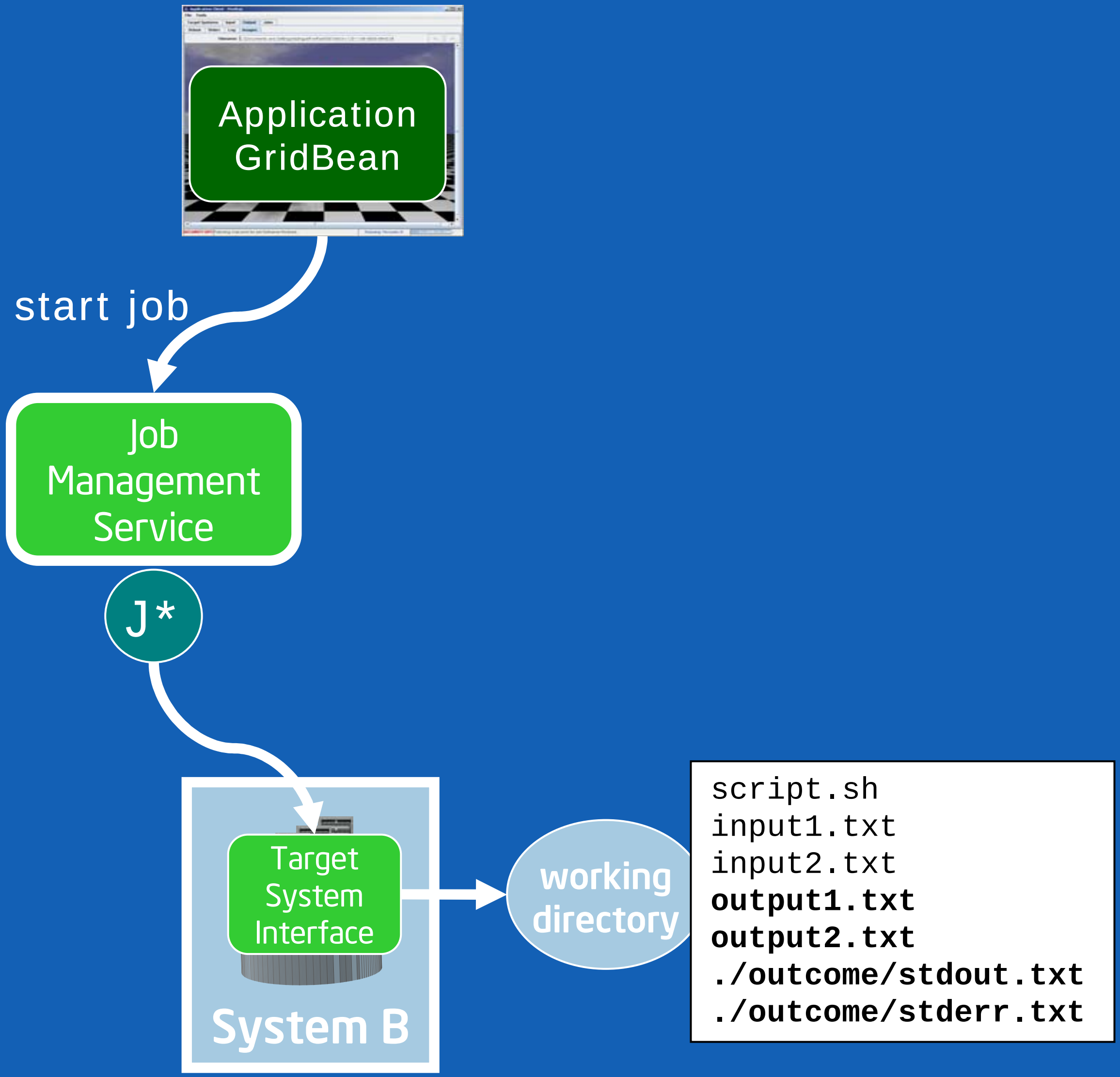
transfer file from local file system

publish information about supported file transfer mechanisms

3rd party file transfer defined in JSDL

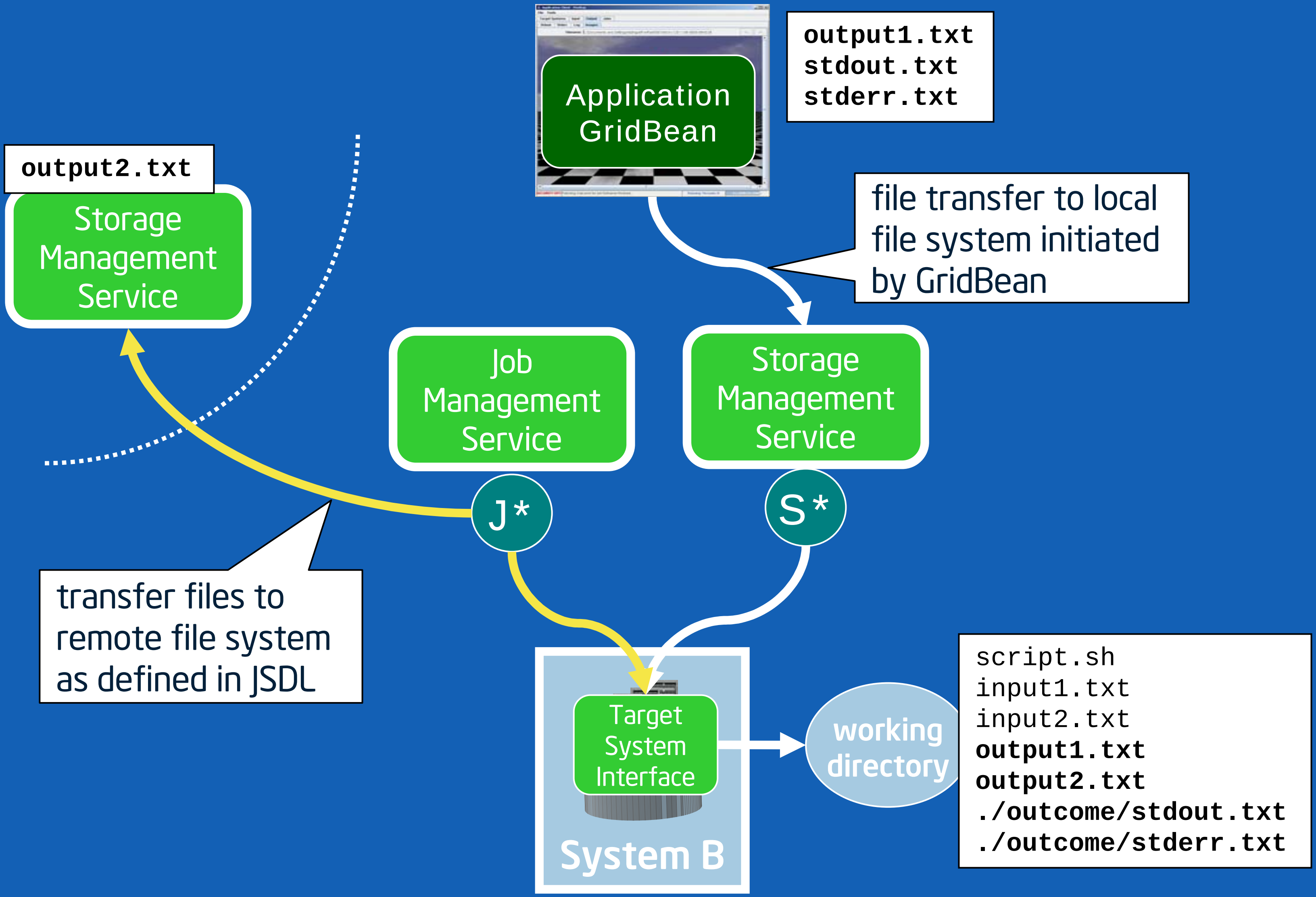
```
script.sh
input1.txt
input2.txt
```

Starting the Job



```
script.sh
input1.txt
input2.txt
output1.txt
output2.txt
./outcome/stdout.txt
./outcome/stderr.txt
```

Retrieving the Outcome



```
output1.txt
stdout.txt
stderr.txt
```

file transfer to local file system initiated by GridBean

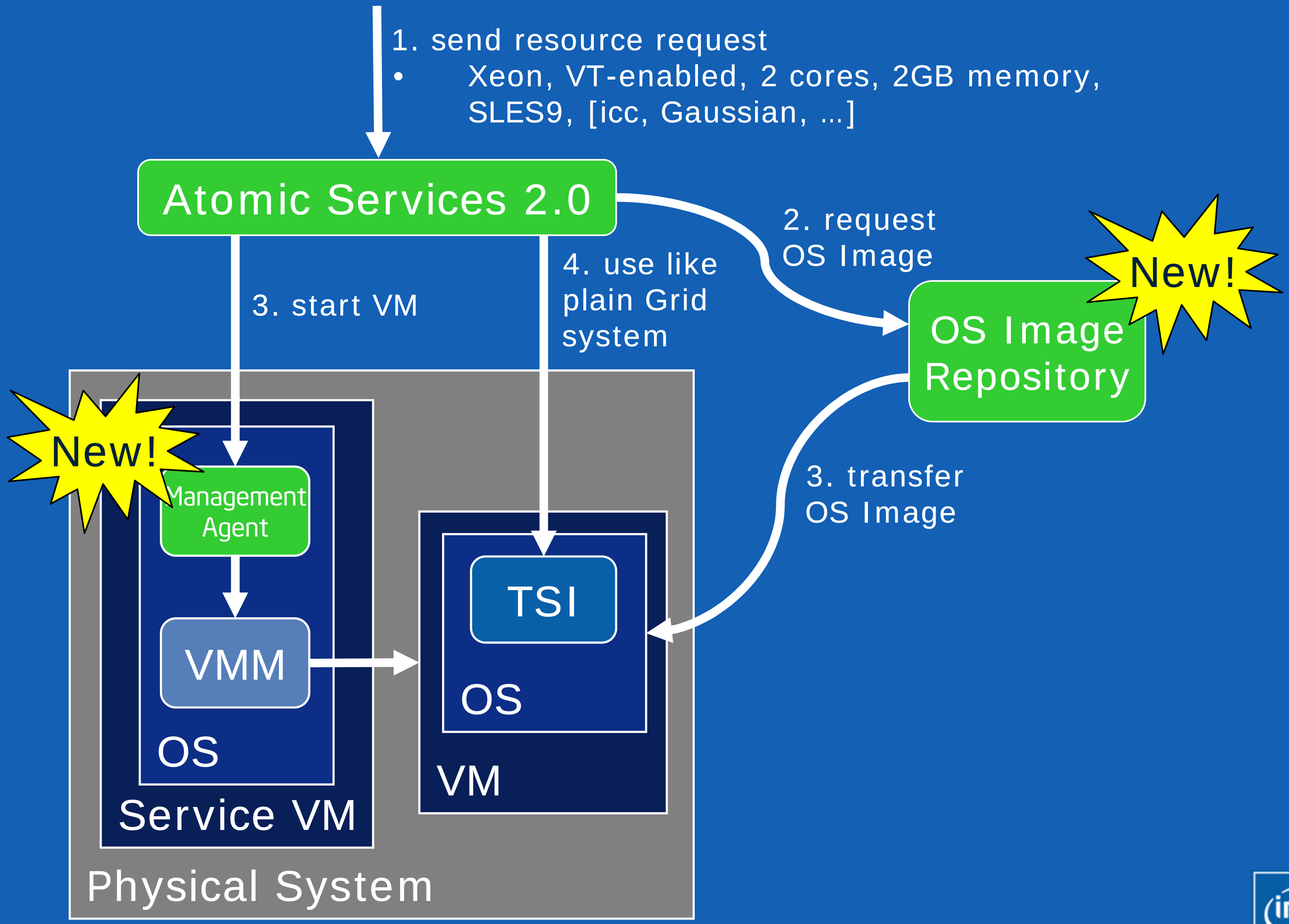
```
output2.txt
```

transfer files to remote file system as defined in JSDL

```
script.sh
input1.txt
input2.txt
output1.txt
output2.txt
./outcome/stdout.txt
./outcome/stderr.txt
```

GPE Outlook

Virtual Machine Provisioning in Action



Value of VM Usage vs. Skepticism Towards New Paradigm

Result from CERN OpenLab

Value

Flexibility

⇨Servers no longer tied to OS versions

Efficiency

⇨Share multi-core/multi CPU servers

Robustness

⇨Migrate processes in case of failure

Security

⇨Isolation of execution environments

Concerns

Security

- Give users root permissions in VMs?
- How to keep control over images being deployed?
- How to avoid network abuse?

Performance

- Compute performance
- Network I/O
- Image transfer/booting overhead

Integration

- Existing infrastructure components need to support new paradigm

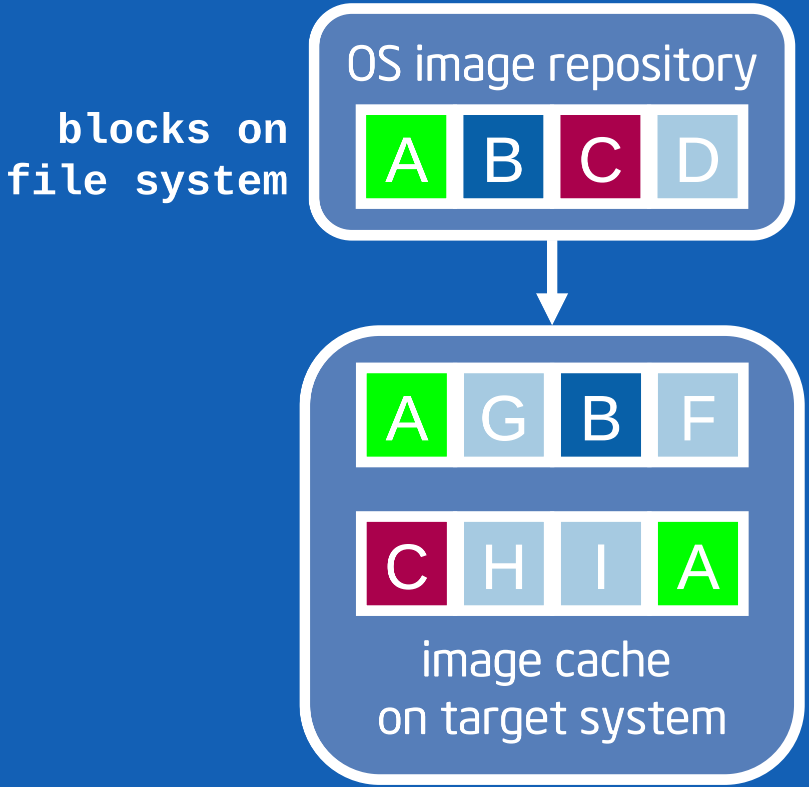
Software

Content-Based OS Image Transfers

Result from CERN OpenLab

Problem: Transfer of OS images from repository to physical system takes too long

Solution: Keep a local cache of OS image blocks on the physical system and transfer only those blocks that do not exist in the cache

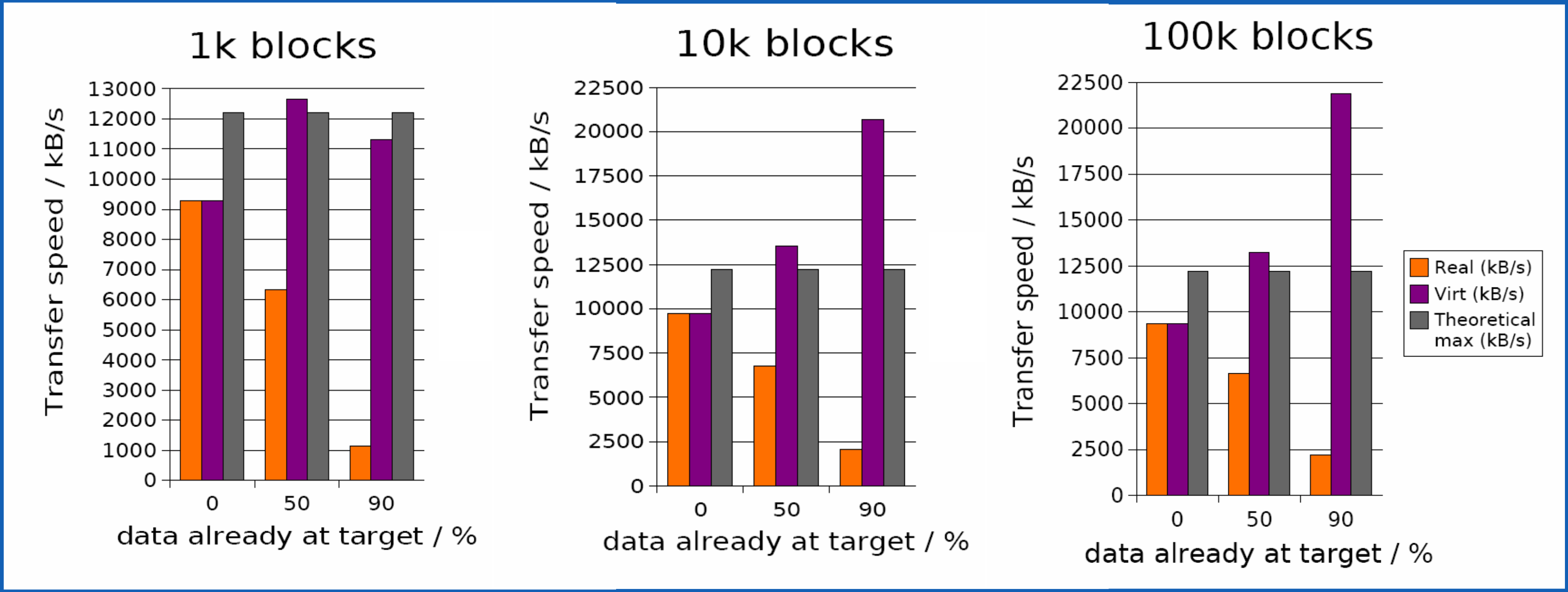


- **Steps**
  - send hash table of OS image to target system
  - compare hash table with table for cached images
  - transfer only missing blocks
- **Costs**
  - linear for hash table creation
  - logarithmic for table sorting
    - n(log n) with n number of blocks

Source: Havard Bjerke, CERN

OS Image Transfer: Results

Result from CERN OpenLab



90 percent common blocks realistic for CERN OS images

- De facto speed-up of factor 1.8

Worst-case:

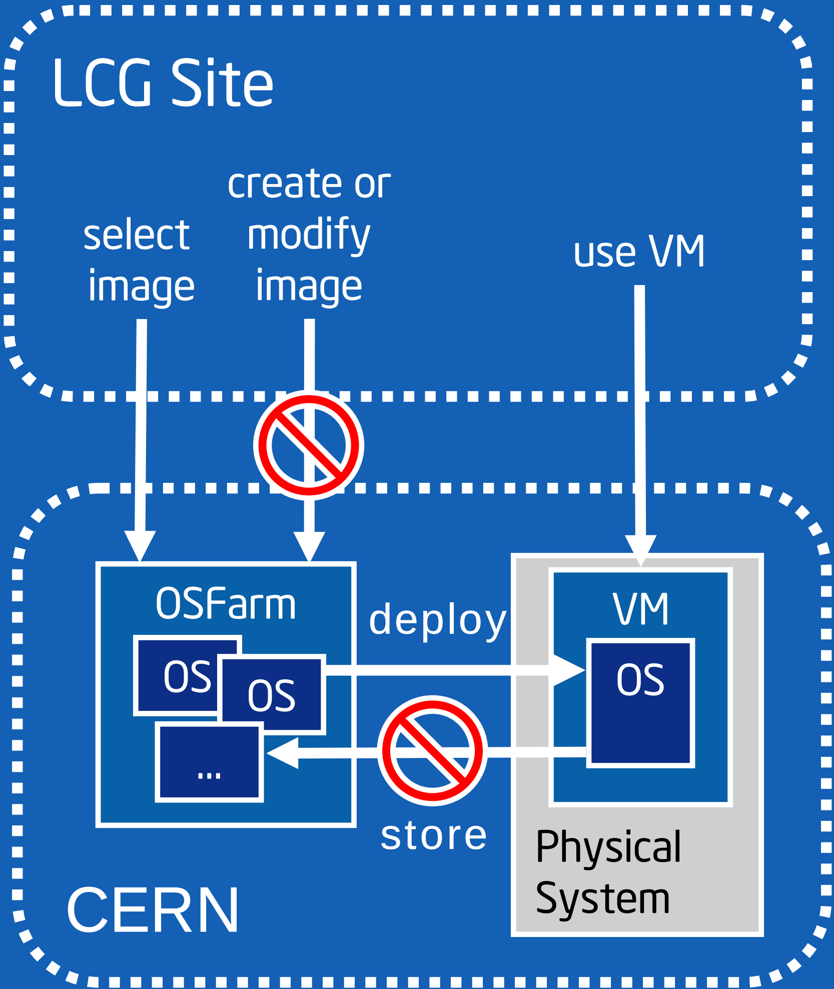
Scenario	Normal transfer	Content Based
lxbatch to lxbatch	5.3 GB	0.88 GB
SLC3 to SLC4	343 MB	185 MB
SLC4 to SLC3	762 MB	609 MB

Source: Havard Bjerke, CERN

Result from CERN OpenLab

Keeping control over VM images on the Grid

- No root access for users in VMs
- Only deploy a fixed set of images from the repository
  - write access only for site administrators
- Sign the images to verify integrity



Bleeding Edge GPE: Examples

- **GPE on Rails**
  - JRuby allows using the GPE API in Rails applications
  - Great framework for AJAX-enhanced Grid portals! (even works with Flex…)
- **Groovy and JRuby as workflow languages for end users**
  - Need a simple way to specify complex jobs and workflows
    - Graphical editors have limitations or get too complex
    - Simple scripting language desirable
  - GPE client API can be invoked from Groovy and JRuby!

GPE Plans for 2007/2008

Version GPE 1.5

- Release Date October 2007
- Implement requirements from Intel software testing
- To be included with UNICORE 6.1

Version GPE 2.0

- Beta Release December 2007
- Integrate new virtualization services
- Integrate Expert Client in Eclipse
- Enhanced APIs

Run proof of concepts at Intel customers

- Continue deployments in HPC community
  - CERN, European UNICORE community, China, Korea

Questions?

Hands-on Session…