ISTITUTO DI SCIENZA E TECNOLOGIE DELL'INFORMAZIONE "A. FAEDO"

Building Digital Libraries on Grid-enabled technologies: challenges, experiences, and results

Tutorial at ECDL 2007
September, 16th 2007

Donatella Castelli (CNR-ISTI, Italy)
George Kakaletris (Univ. of Athens, Greece)
Pasquale Pagano (CNR-ISTI, Italy)
Heiko Schuldt (Univ. of Basel, Switzerland)

UNI BASEL

Agenda

14:00 – 14:30 **The Digital Library Infrastructure paradigm**

14:30 – 14:45 **Grid computing**

14:45 – 15:30 **Infrastructure management**

15:30 – 16:00 ***Coffee break***

16:00 – 17:00 **Content management, Process management and Search**

17:00 – 17:30 **Demos**

17.30 – 17:40 **Conclusions**

- Slides will be available at
  - http://diligent-training.isti.cnr.it/

The Digital Library Infrastructure paradigm

Donatella Castelli
ISTI-CNR

Overview

- Requirements (of new generation Digital Libraries)
- Resource sharing (a means to satisfy these requirements)
- Infrastructures (a sustainable solution to sharing)

DELOS: Grand 10-Year Vision

"The potential exists for digital libraries to become the **universal knowledge repositories and communication conduits for the future**, a common vehicle by which everyone will **access, discuss, evaluate, and enhance information of all forms**"

Requirement: Universal Knowledge Repositories

- DL creation and maintenance must "*cheap*" since many of the organizations that demand DLs are small, distributed, and dynamic; they use the DL to support temporary activities such as courses, exhibitions, projects, etc.

Requirement: Information in all forms

- Multimedia documents (e.g. images, audio-videos, 3D-objects)
- Data (e.g. sensor data, experimental data), products (e.g. outcomes of specific data processing)
- Compound information objects
- On-demand information objects
- …..

On-demand information objects

- a fixed text
- a pollution map
- a table summarizing data
  from millions of observed satellite measures
- a graph reporting an analytical trend of certain information extracted from a great amount of observed data

International Report on
Mediterranean Sea Chlorophyll Distribution during year 2003

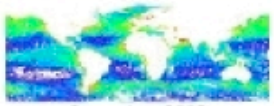
1. Scientific and Societal Concerns

Any scheme to monitor the ocean biota and their environment must strive to address the major scientific and societal concerns of the day pertaining to marine life. This section summarises some major concerns that emerged during discussions at the meeting. Many other concerns could have been included, but space precludes a complete listing of concerns.

1.1. Biodiversity and Conservation

Marine biodiversity is not easy to assess and is generally poorly known. There are many complicating factors, including a three-dimensional, fluid, mobile environment, its vastness, and its challenging depths. Away from shore, primary producers and primary grazers are usually small, drifting forms that undergo spatial variability and seasonal changes.

The larger invertebrate grazers have a range of life history stages, often with planktonic and benthic phases. Many large animals are migratory. Ocean habitats can be linked by the dispersal of planktonic larvae, and in this way, the systems can be interconnected even at a distance.

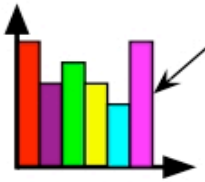


Jan – Apr 2003

Finally, the higher-order diversity of life is much greater in the oceans than in terrestrial systems—there are 13 unique phyla in the oceans and only one on land. Marine biodiversity is essentially the evolutionary history of life. In general, long-term environmental stability seems to increase biodiversity and, conversely, global climate change can be expected to decrease it.

	X1	X2	X3	X4	X5	X6	X7	X8	X9
Y1	12	13	15	26	11	34	45	45	54
Y2	32	12	46	67	21	22	44	12	44
Y3	23	33	56	77	32	44	12	55	33
Y4	44	34	12	55	34	45	12	22	44

Measures of yyy



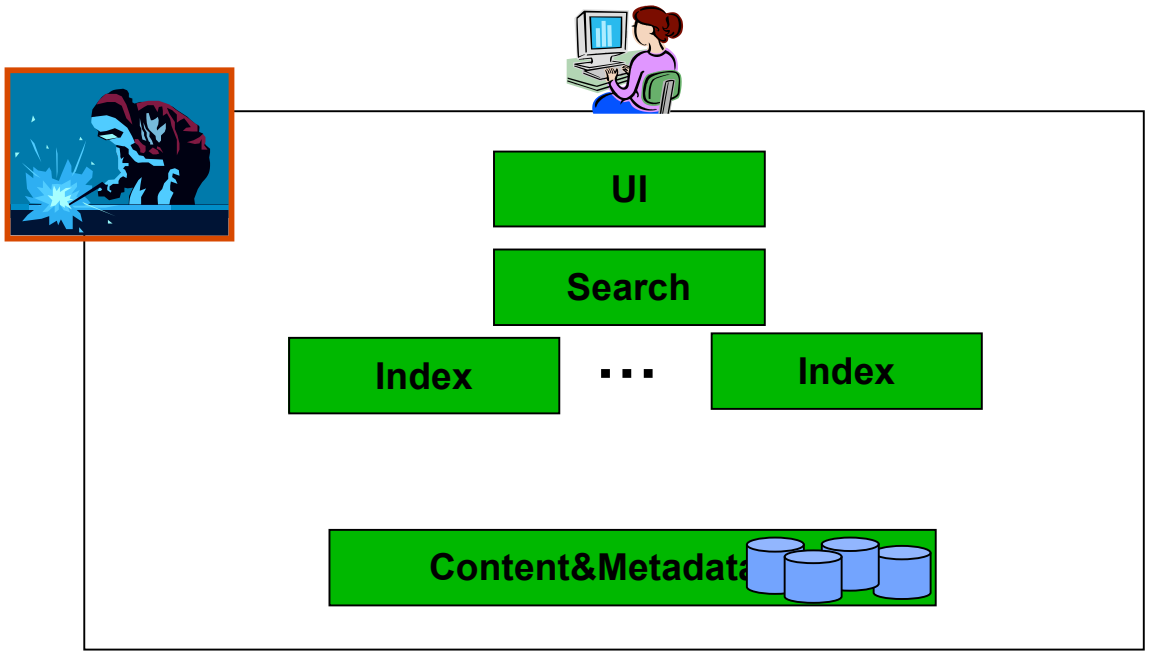
Values of xxx

Automatically updated with the most recent data

Supporting on-demand information objects

- Access to many different, large, heterogeneous information sources
- Use of specialized services
- Large processing capabilities

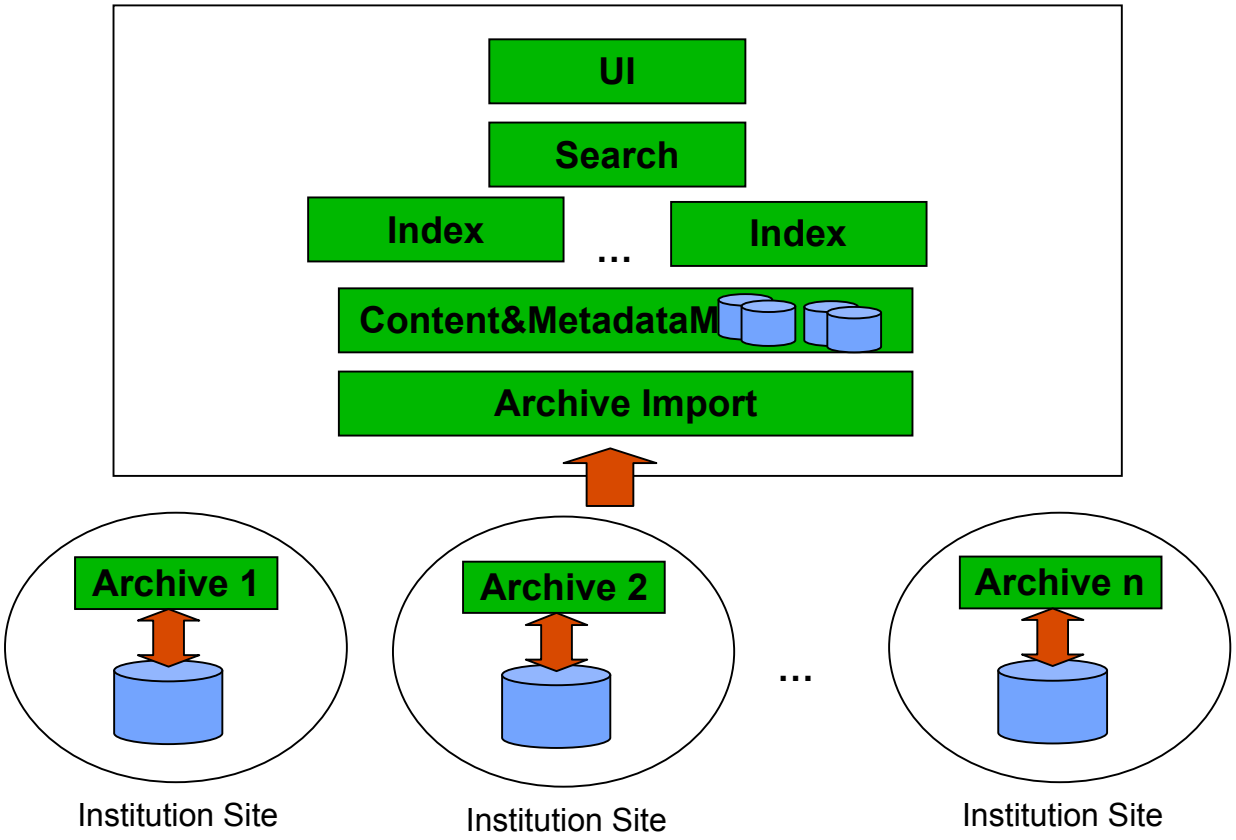
DLs built from scratch



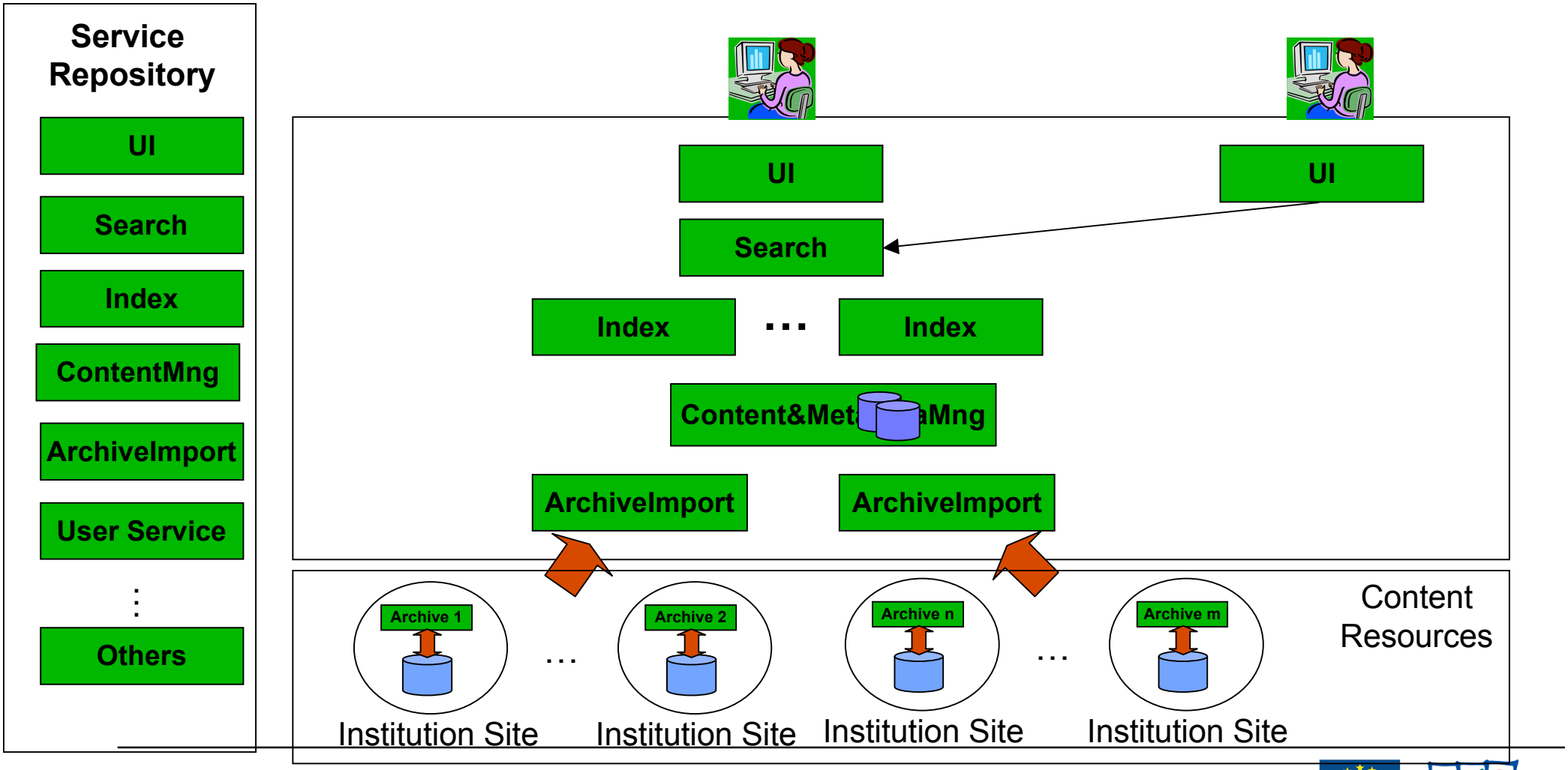
Lack of sustainability

- The construction and management of a DLS requires high investments and specialized personnel, content production is very expensive, multimedia and data handling requires high computational resources
- Years are spent in designing and setting up a DL
- The systems lack interoperability and the services provided are difficult to reuse

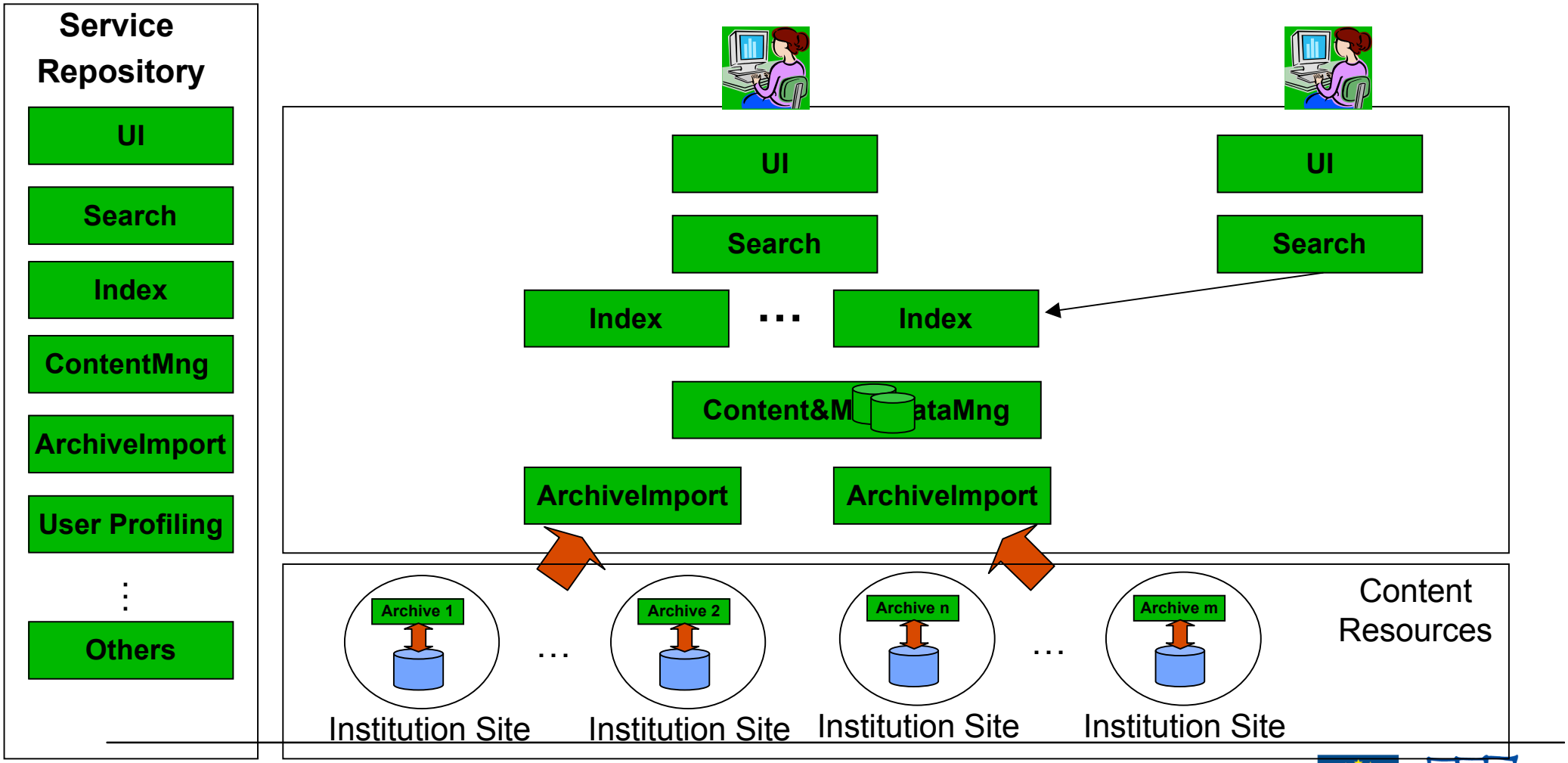
Sharing of content



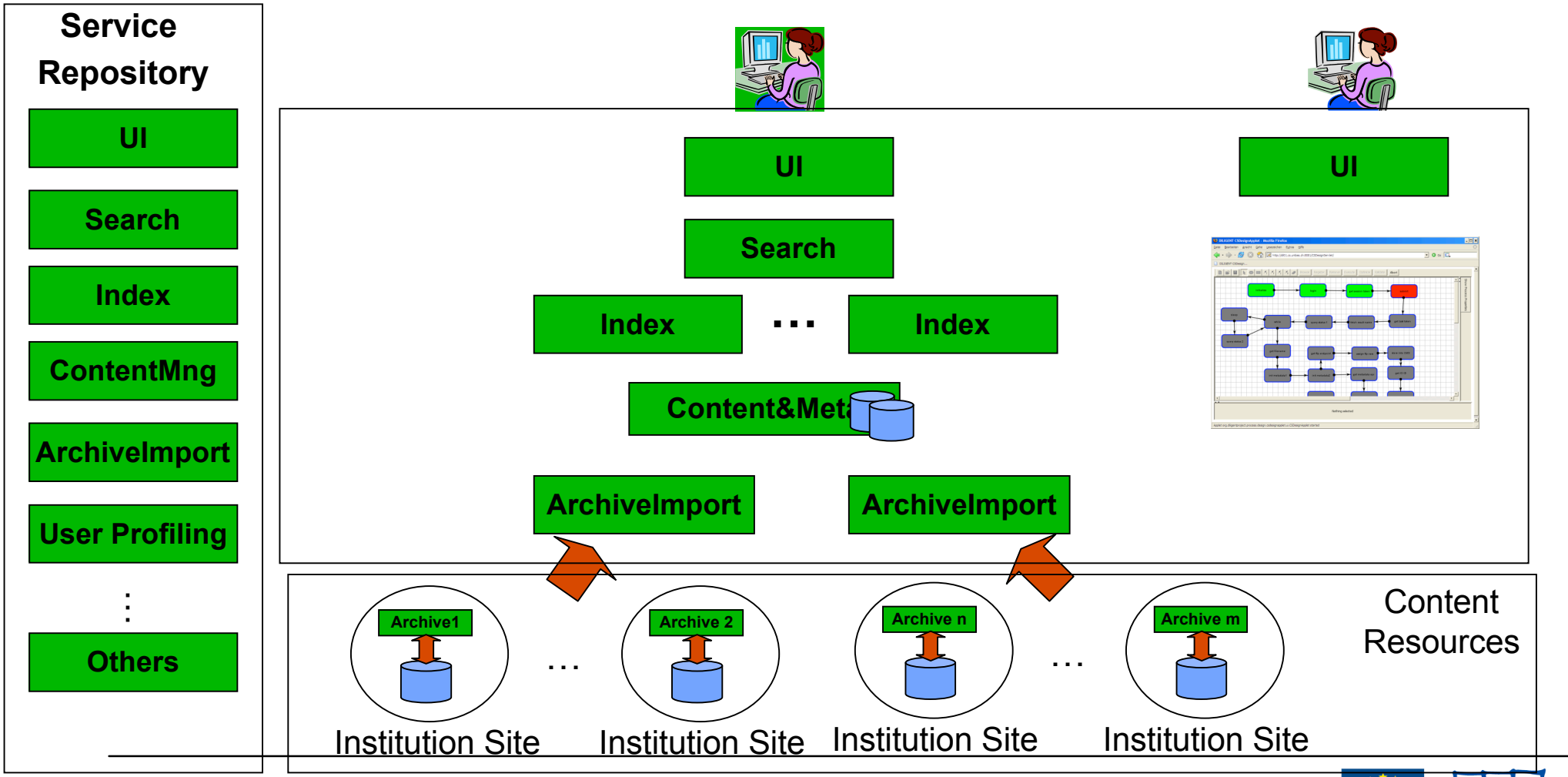
Sharing of other resources: services



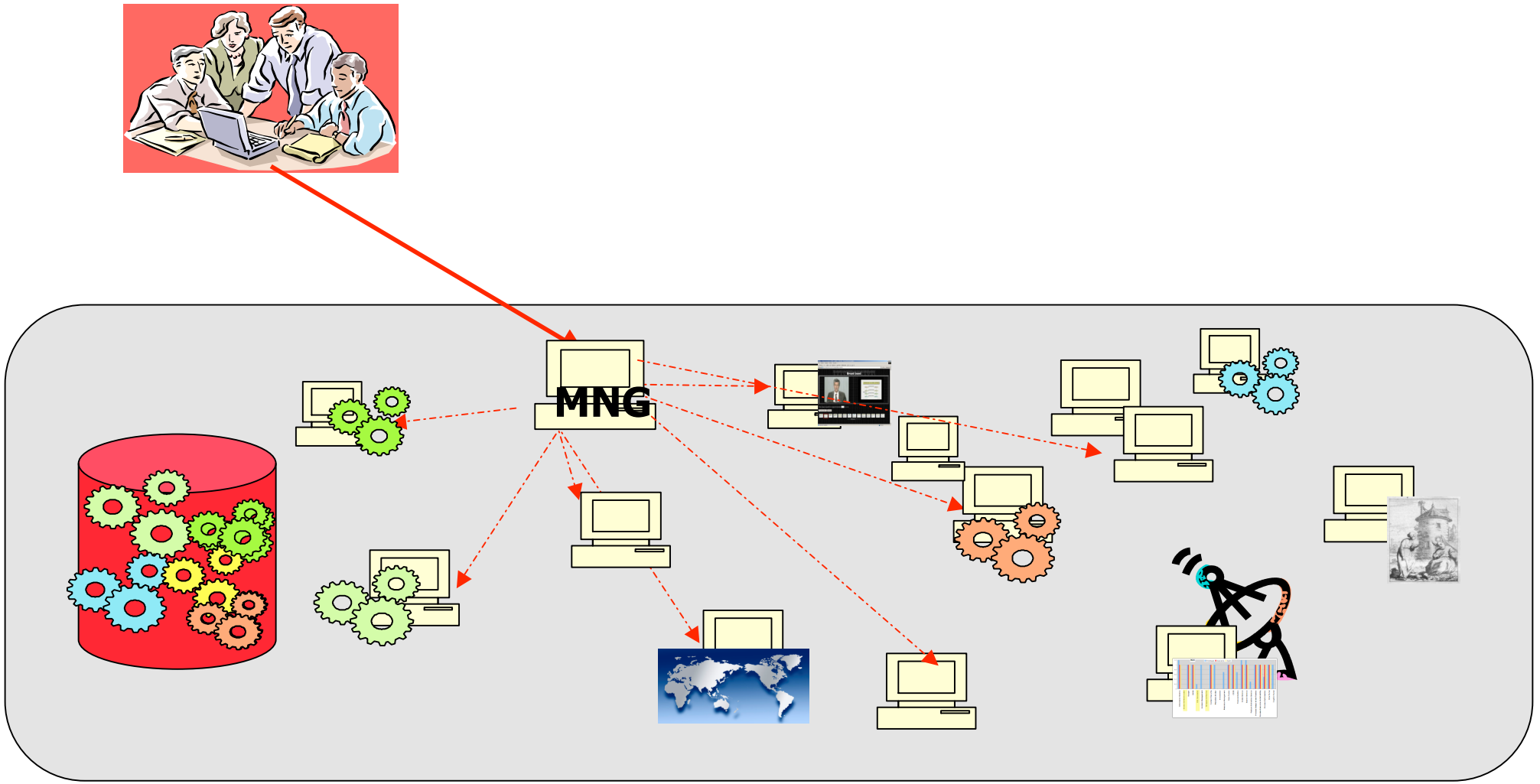
Sharing of other resources: services (cont.)



Sharing of other resources: composition of services (cont.)



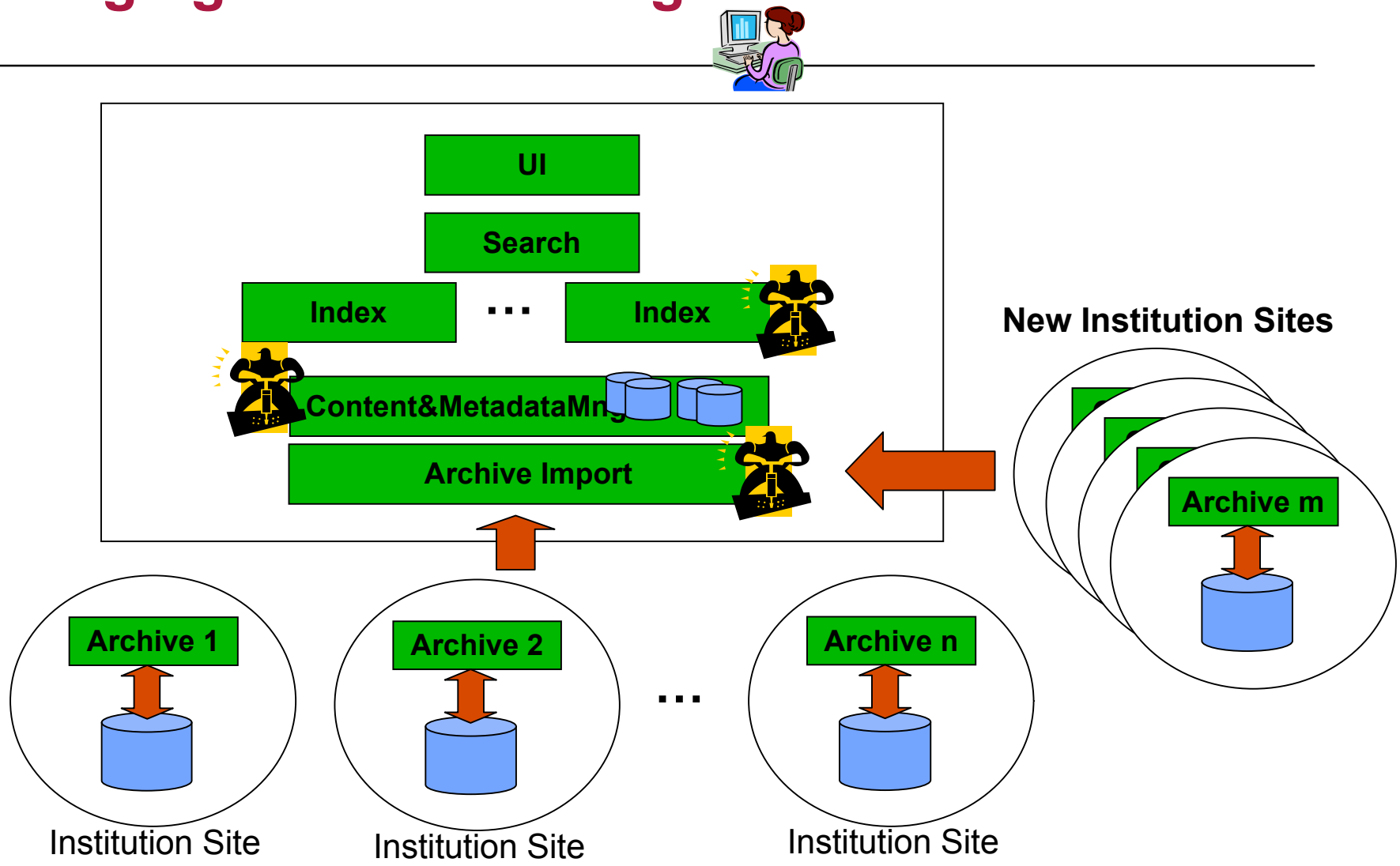
Sharing of other resources: computing and storage



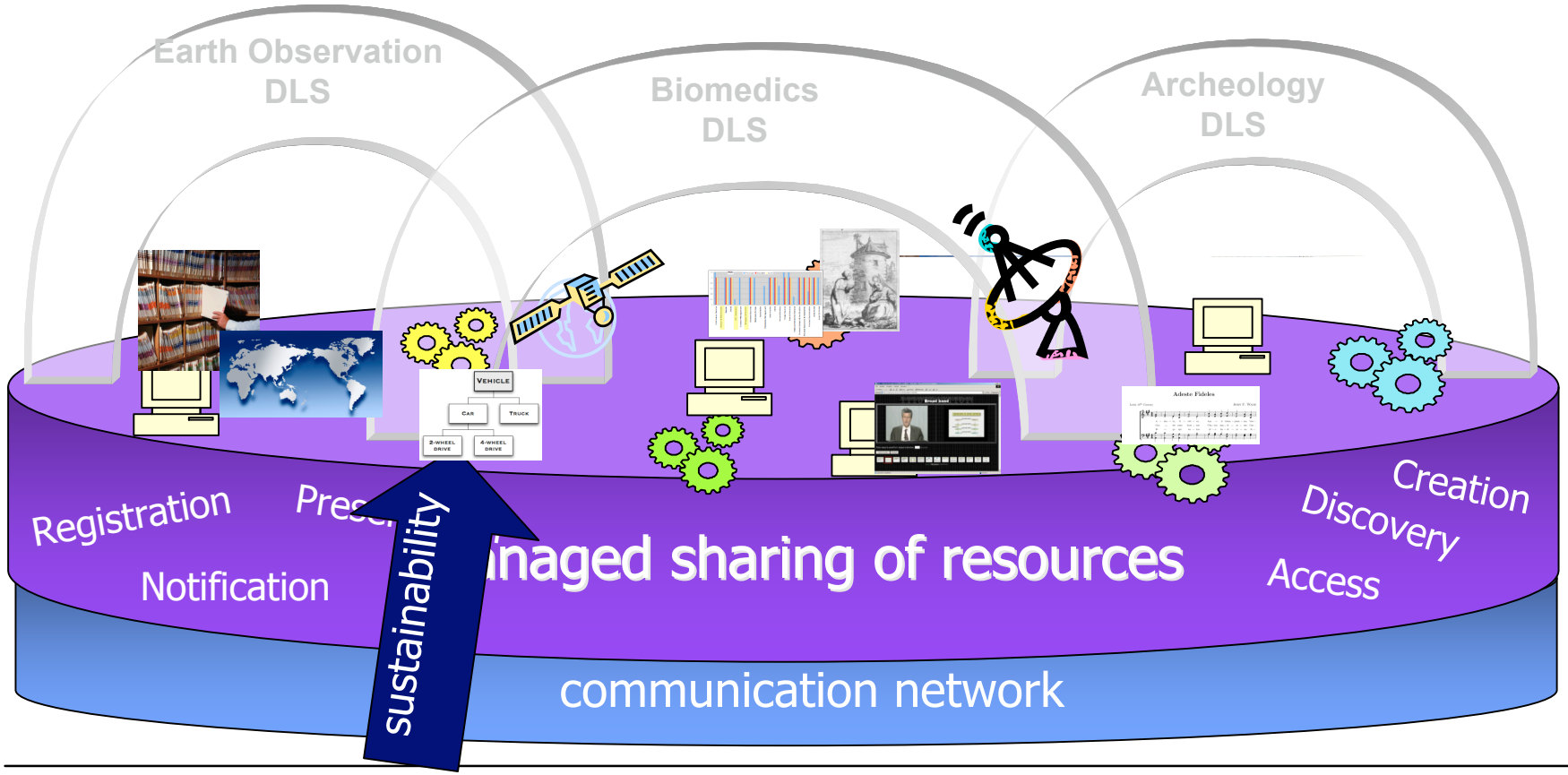
New capabilities become sustainable by many

- 3D and multimedia objects
- “Products” generated by processing large amount of data
- On-demand information objects
- Complex feature extraction
- …..

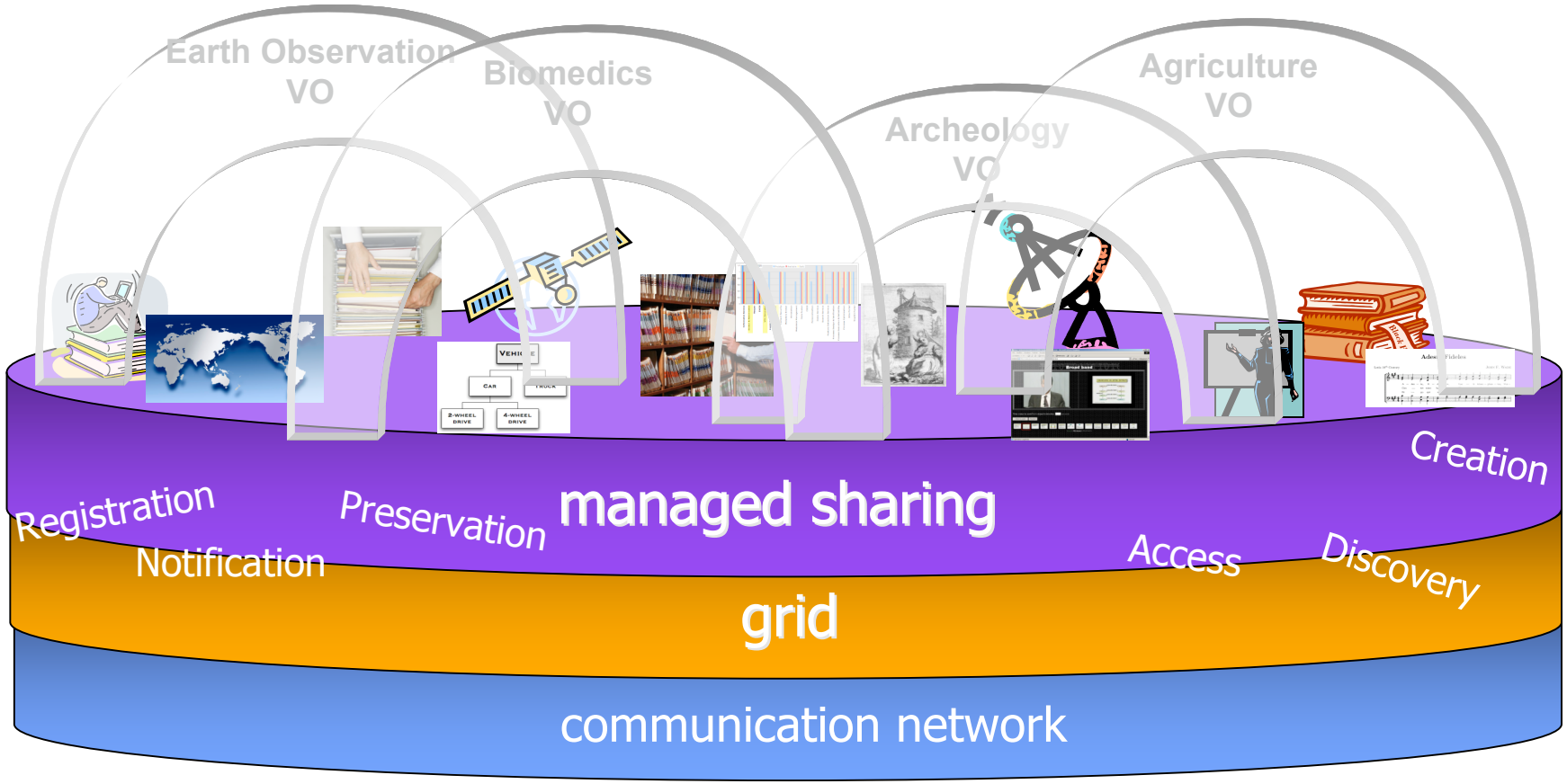
Managing content sharing



Infrastructures: outsourcing the resource sharing management



Infrastructure: exploiting grid solutions



New Challenges

- Infrastructure management
- Design DL services which exploit the sharing capabilities at the best

DILIGENT: A DIgital Library Infrastructure on Grid-ENabled Technology

Digital Library Infrastructure on Grid-Enabled Technology that allows members of dynamic virtual organizations to create on-demand transient virtual digital libraries based on shared resources, i.e. processing and storage capabilities, multi-type content and applications

PARTNERS

- ERCIM (FR)
- CNR-ISTI (IT)
- Univ. of Athens (GR)
- Univ. of Basel (SW)
- Engineering (IT)
- CERN (IT)
- Univ. of Strathclyde (UK)
- European Space Agency (IT)
- FAST (SW)
- RAI (IT)
- Scuola Normale Superiore (IT)
- 4D-Soft (HU)

Diligent

Grid computing

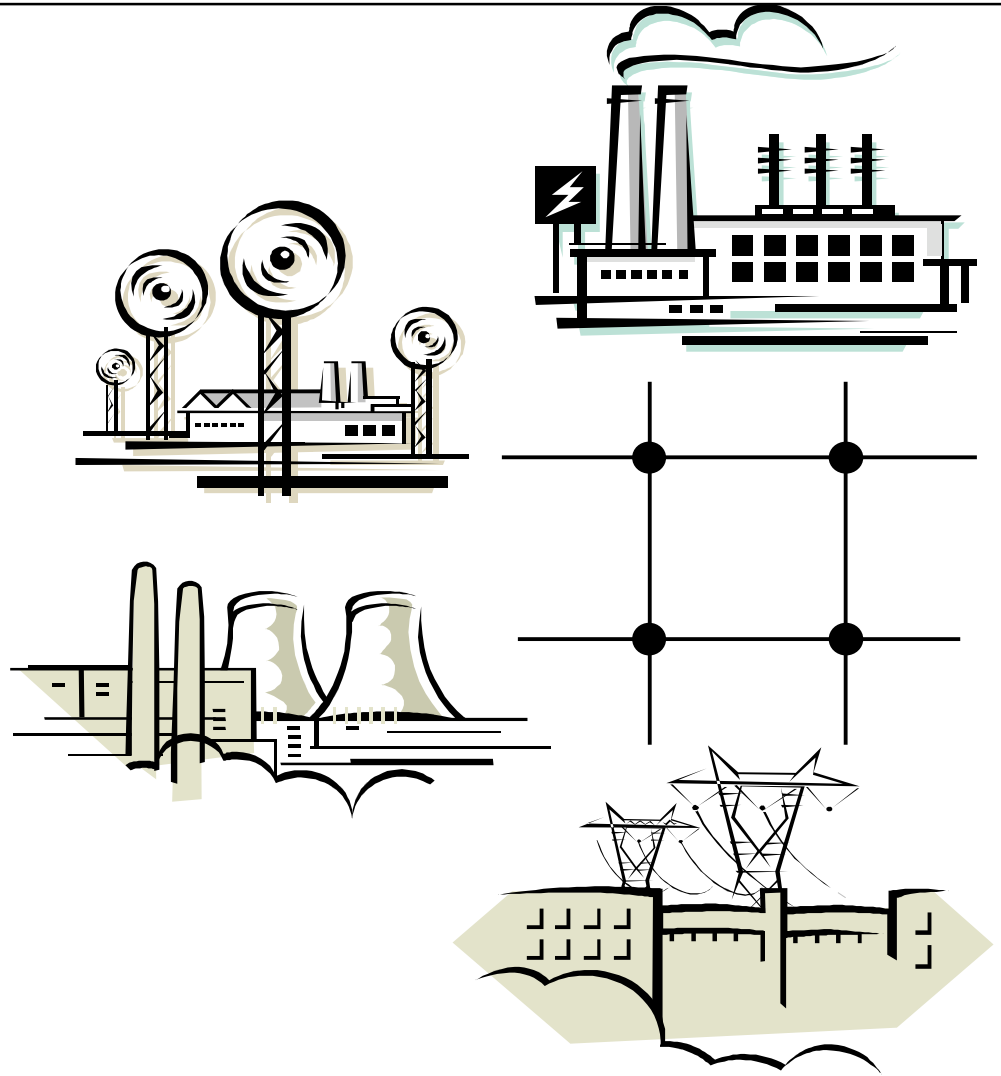
Pasquale Pagano
CNR-ISTI

Information Society Technologies

Overview

- What is GRID computing?
- Why is it different?
- Why do it?
- Grid Peculiarities

Grid Infrastructures: Basic Idea …



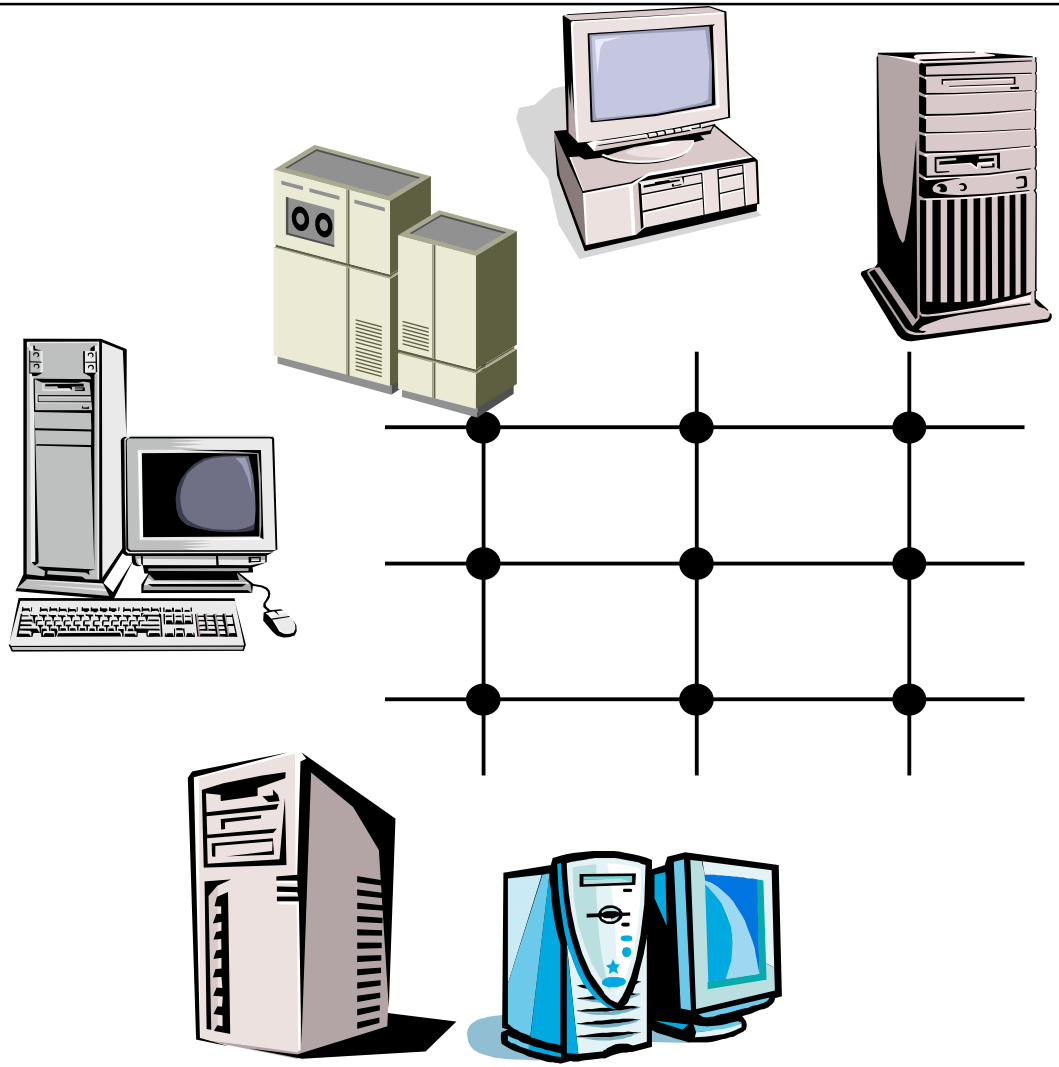
Basic Idea:

- Resources are available in a network without limitation

Simple access to resources; users do not need to be aware of their origin/location

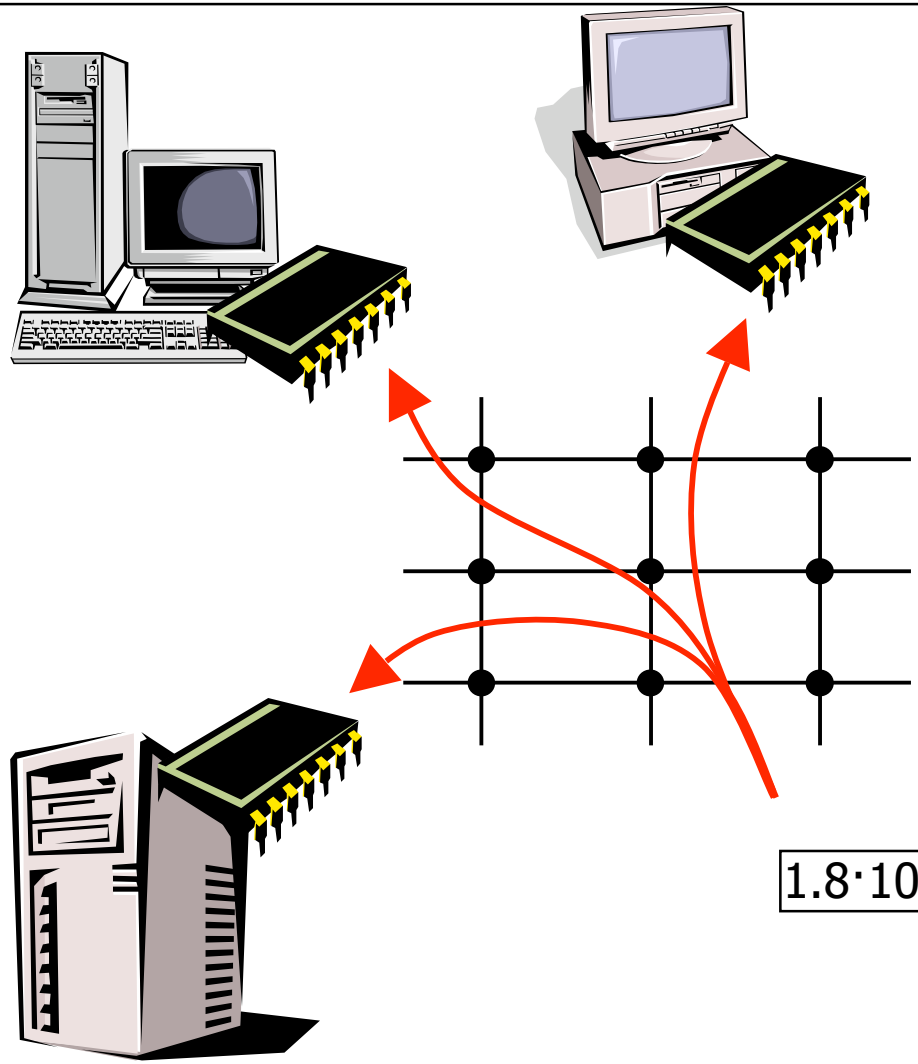
- Example: power line

… Grid Infrastructures: Basic Idea



Analogously, also computer and computing resources of a distributed system shall be exploited

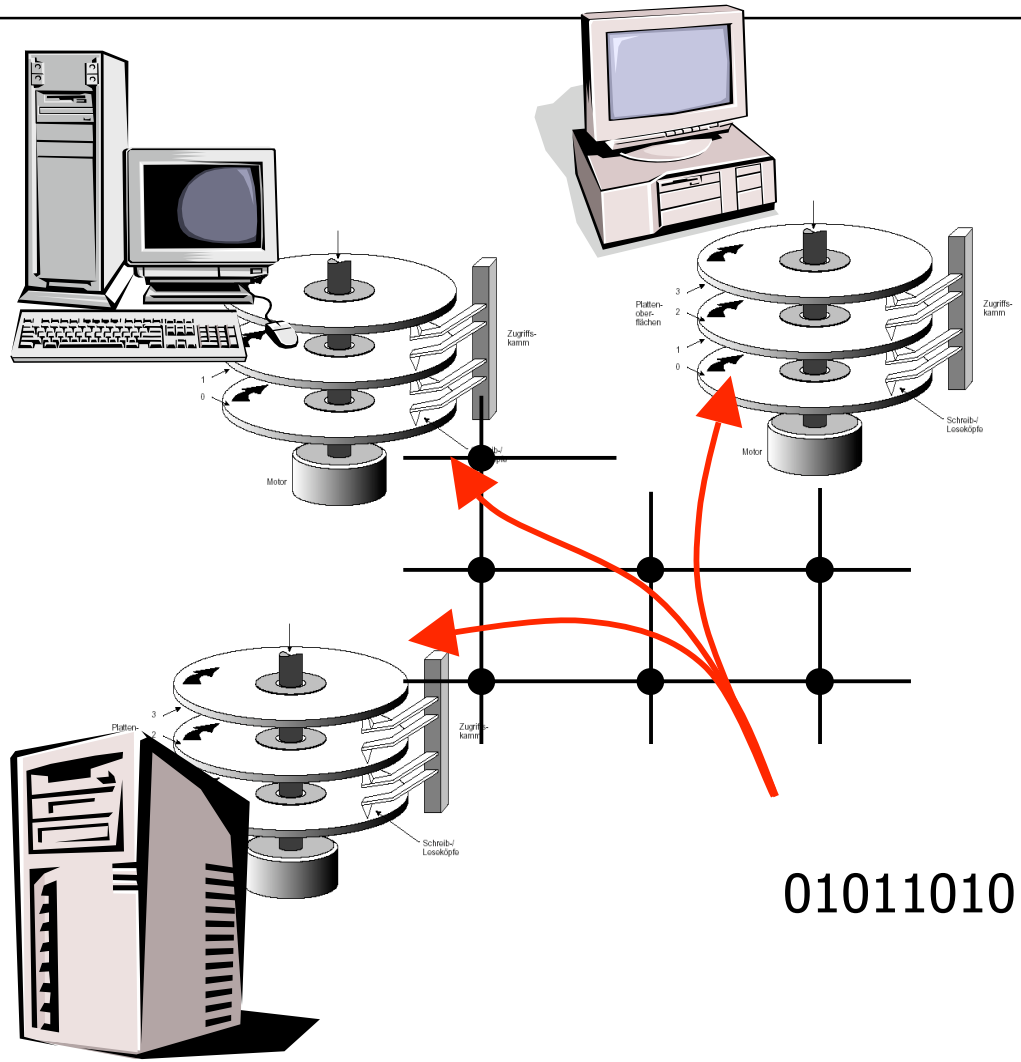
Computational Grid



- Resource which is shared in the Grid: CPU
- Large, complex computationally intensive problems are split in smaller ones and are processed in parallel in the Grid

$1.8 \cdot 10^{342} * 1.35 \cdot 10^{520} *$ $9.23 \cdot 10^{911} / 8.51 \cdot 10^{100} * ..$

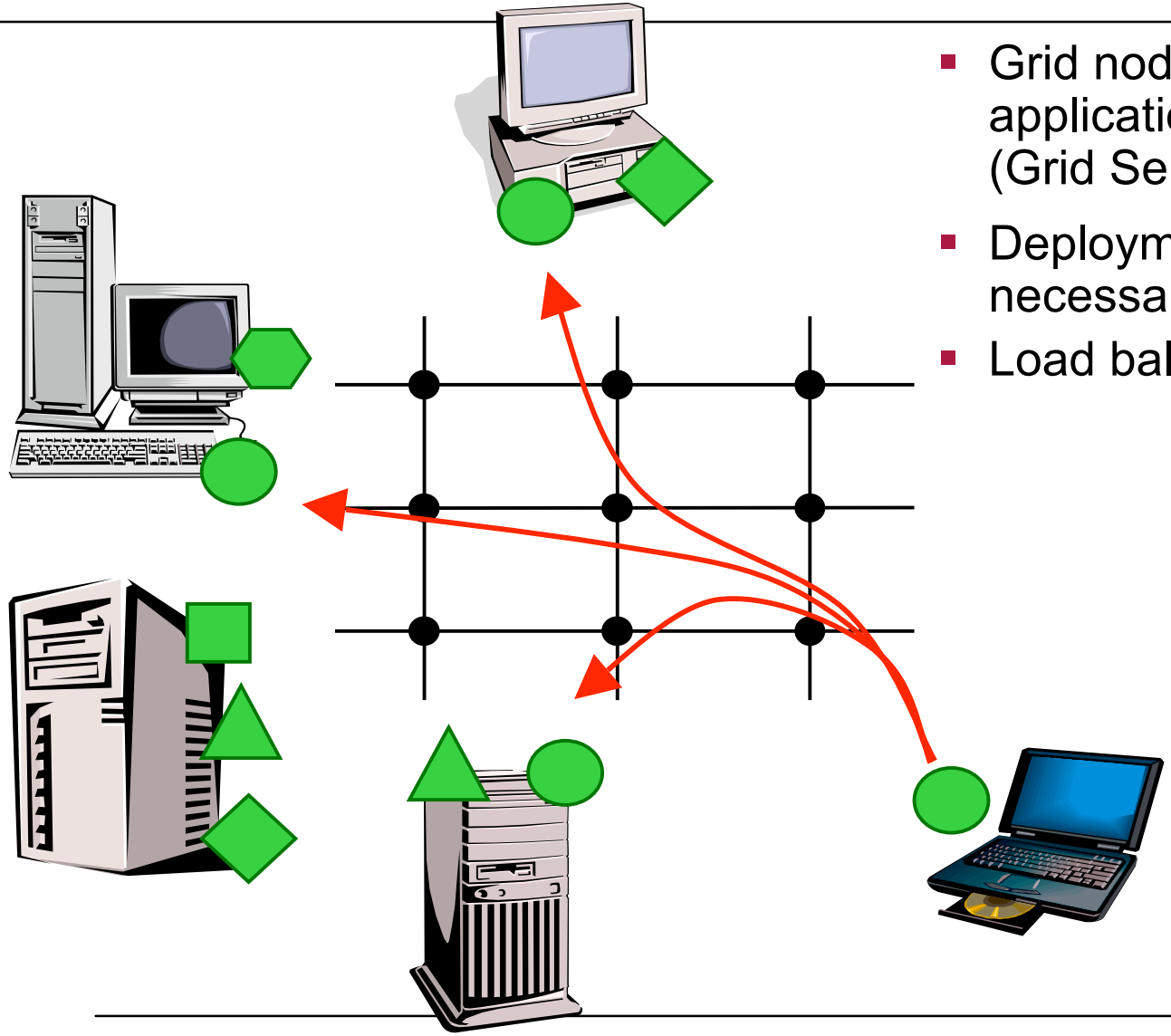
Data Grid



- Grid nodes (computers) provide storage
- The data Grid shall act as a large-scale, distributed database
- New data can be distributed among available storage nodes in the Grid

0101101010010101001011101011011011

Service Grid



- Grid nodes provide dedicated application logic as services (Grid Services)
- Deployment of new services if necessary, on less loaded hosts
- Load balancing in the Grid

What is GRID computing?

[dreamer's vision]

Whereas the Web is a service for sharing information over the Internet, the Grid is a **service for sharing computer power and data storage** capacity over the Internet. The Grid goes **well beyond simple communication** between computers, and aims ultimately to turn the global network of computers into **one vast computational resource**.

[scientist's vision]

Grid computing provides **flexible, secure, coordinated resource sharing** mechanisms among dynamic collections of individuals, institutions, and resources (***virtual organizations***).

GRID computing is about

- Resource sharing
- Secure access
- Resource use
- Wide area network
- Virtual organization

- Open Standards

Grid computing is about Resource Sharing

Resources are:

- computers, storage, data, remote software, sensors, networks, …
- owned by many different organizations
  - exists within different administrative domains,
  - run different software,
  - different security and access control policies.

Resource sharing is not about getting something for nothing, but is a situation where everyone concerned sees the advantage of sharing.

Grid computing is about Secure Access

Resource Sharing is always conditional: issues of trust, policy, negotiation, payment, …

Grid deals with:

- **Access policy** - resource providers and users must define clearly and carefully what is shared, who is allowed to share, and the conditions under which sharing occurs;
- **Authentication** - you need a mechanism for establishing the identity of a user or resource;
- **Authorization** - you need a mechanism for determining whether an operation is consistent with the defined sharing relationships.

Grid computing is about Resource Use

Efficient use of resources. No matter how many resources you have, there will always be times when there is a queue of people waiting to use them.

On the Grid, the **information about the different activities being submitted** are accessible, and the Grid MW is able to calculate the **optimal allocation of resources**.

Grid activities are beyond client-server communication and includes **coordinated problem solving**: distributed data storage and analysis, computation, collaboration, …

Grid computing is about Wide Area Network

The **performance** of wide area networks has been **doubling every nine months** or so over the last few years.

- WANs operate at 155 Mbps when in 1985 the US supercomputer centers were connected at 56 Kbps

Is it enough?

Ultra-low latency so there is no delay for applications when working distributed on the Grid

Compensation for any failure that occurs on the Grid during a calculation, be it a transmission error or a PC crash

Grid computing is about Virtual Organization

Dynamic, multi-institutional aggregation of resource providers and consumers.

Community of different organizations:

- overlays on classic organization structures
- implies different system administrators, users, institutional goals, and often socio-political constraints

Large or small, static or dynamic

Is the Grid vision new?

"We will perhaps see the spread of 'computer utilities', which, like present electric and telephone utilities, will service individual homes and offices across the country."

Len Kleinrock, 1967

Is the Grid the first attempt?

Condor (University of Wisconsin)

- Software system that creates a High-Throughput Computing (HTC) environment
- Effective utilization of computing power (workstations, servers, clusters)
- Specialized workload management system for compute-intensive jobs
- Started already in 1988

Why is this Different?

Lack of central control

- Where activities run
- When they run

Shared resources

- conflict, evolve

Communication and coordination

- Cross-domain, cross-operating system, cross-resource type, …

So why do it?

- Computations that need to be done with a human acceptable time limit
- Data that cannot fit on one site
- Data produced by multiple sites
- Peaks of computations that are limited to specific events
- Applications that need to be run bigger, faster, …

Grid computing peculiarities

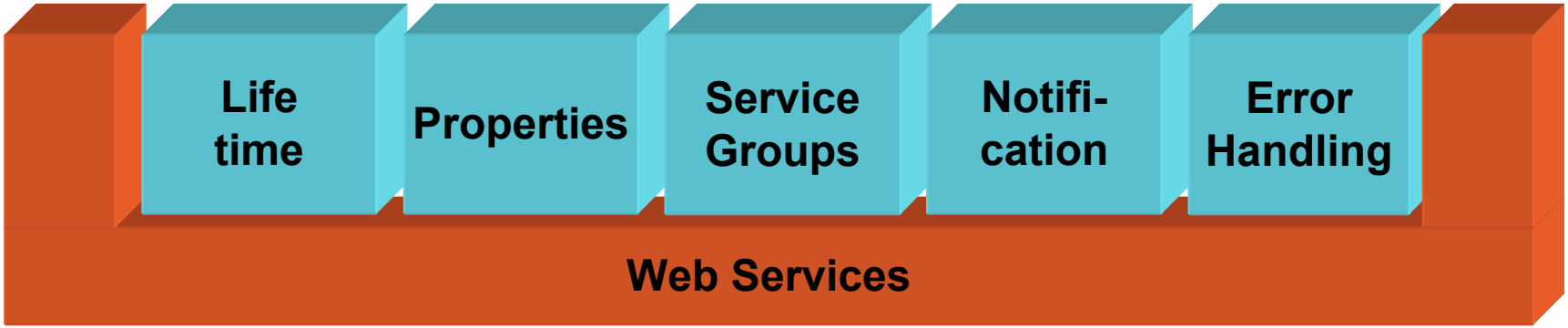
Distributed management

- Of physical resources
- Of software services
- Of communities and their policies

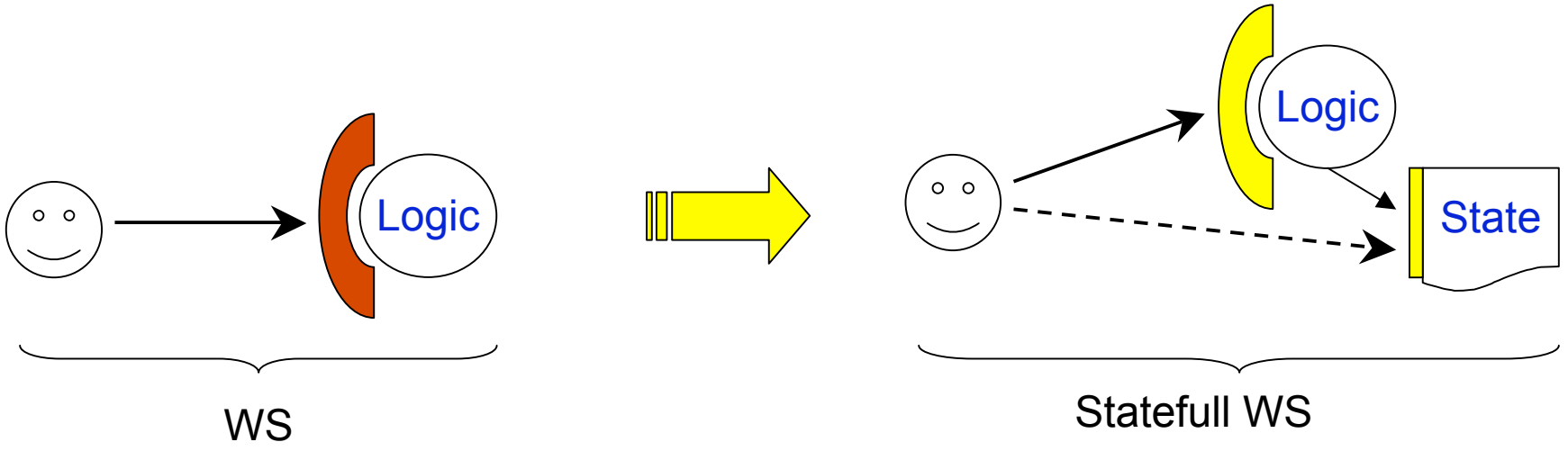
Unified approach

- Build on Web Service framework
  - Common management abstractions & interfaces
- Use WSRF, WS-Notification to represent/access state
  - Common state management

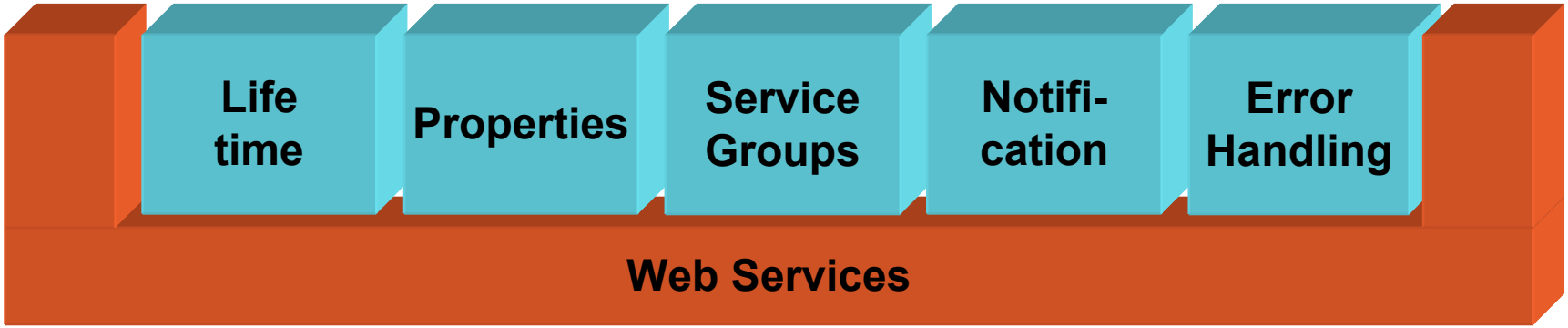
Web Service Resource Framework (WSRF)



Unified way to model and interact with stateful web services



Web Service Resource Framework (WSRF)



Unified way to model and interact with stateful web services

Lifetime (WS-ResourceLifetime)

- Factory based dynamic creation of services
- Instances are created with a limited lifetime
- Prevent services from consuming resource indefinitely (“Garbage Collection”)

Properties (WS-ResourceProperties)

- Defines type and values of a resource state

Service groups (WS-ServiceGroups)

- Collection of grid services (e.g. all resources of a cluster)
- E.g. to distribute an action to a set of services

Notification (WS-Notification)

- Notification about state changes
- Applies traditional publish/subscribe paradigm

Error Handling (WS-BaseFaults)

- Defines base handling of communication errors

Advantages and Disadvantages

Advantages

- Clear Description of Resources and Interfaces
- Dynamic sharing of resources
- On-demand services exploitation
- Cross-organizations trusted environment
- Widely accepted Web Service standards

Disadvantages

- Reference implementations are still in development
- Several complementing specifications are in development
- Complex middleware requires maintenance and administration overhead

Where do these slides come from?

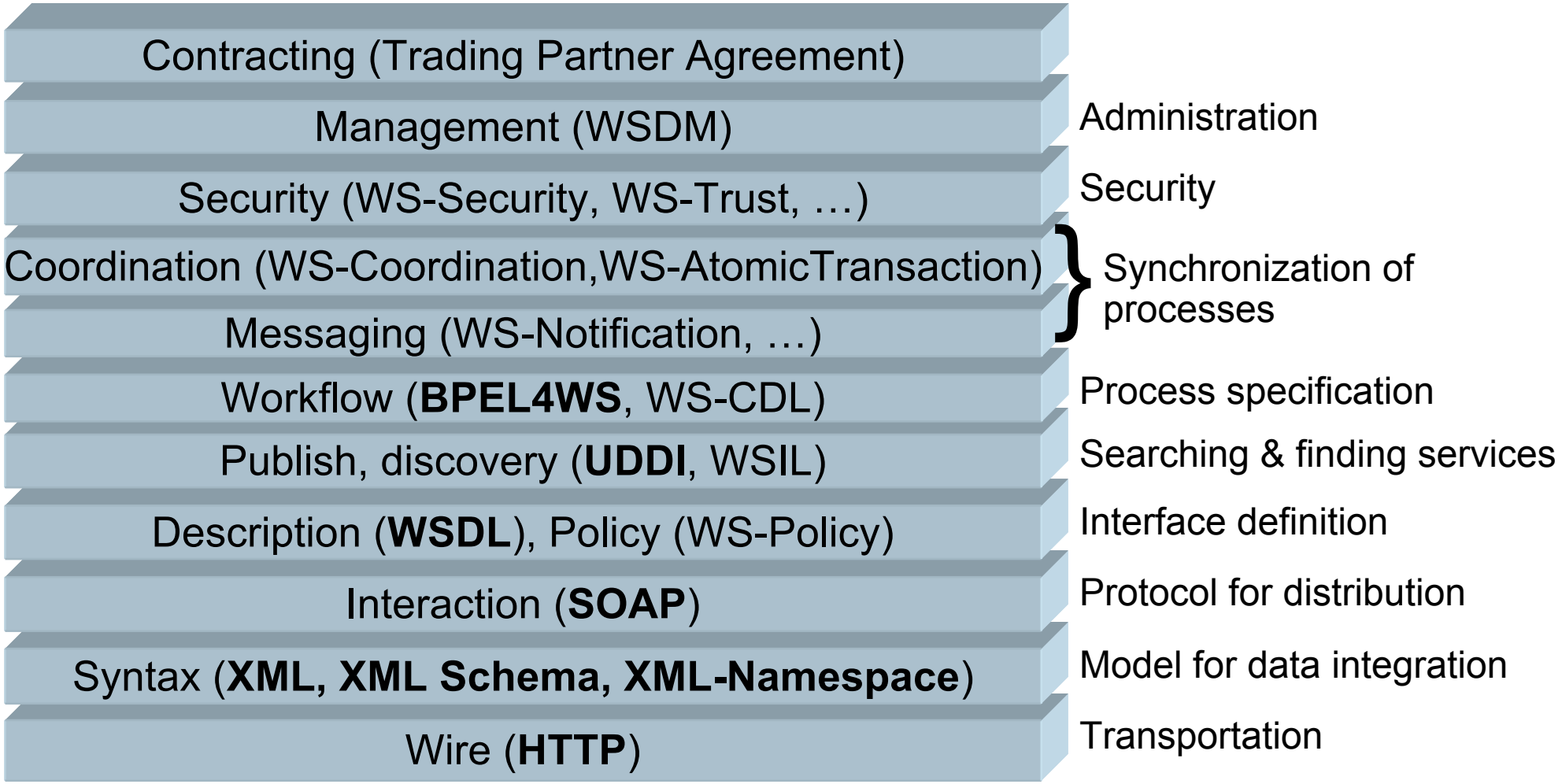
These slides elaborates ideas taken from:

- Globus alliance: www.globus.org
- Cern lab: gridcafe.web.cern.ch/
- Diligent project: www.diligentproject.org
- Dave Snelling talk at EGEE 2006 conference: http://egee-technical.web.cern.ch/egee-technical/conferences/EGEE06

Further readings

- F. Berman, G. Fox, A. Hey (Eds.). **Grid Computing – Making the Global Infrastructure a Reality**. John Wiley & Sons, 2003, ISBN: 0-470-85319-0.
- I. Foster, C. Kesselman (Eds.). **The Grid: Blueprint for a New Computing Infrastructure**. Morgan Kaufmann Publishers, 2003. ISBN: 1-55860-933-4
- I. Foster, C. Kesselman, J. Nick, S. Tuecke. **The Physiology of the Grid: An Open Grid Services Architecture for Distributed Systems Integration**. Open Grid Service Infrastructure WG, Global Grid Forum, June 22, 2002.
- I. Foster, C. Kesselman, S. Tuecke. **The Anatomy of the Grid: Enabling Scalable Virtual Organizations**. International J. Supercomputer Applications, 15(3), 2001.
- I. Foster. **What is the Grid? A Three Point Checklist**. GRIDToday, July 20, 2002.
- I. Foster and A. Iamnitchi**. On Death, Taxes, and the Convergence of Peer-to-Peer and Grid Computing**. IPTPS '03

Web Services Stack



Relevant Standards Overview

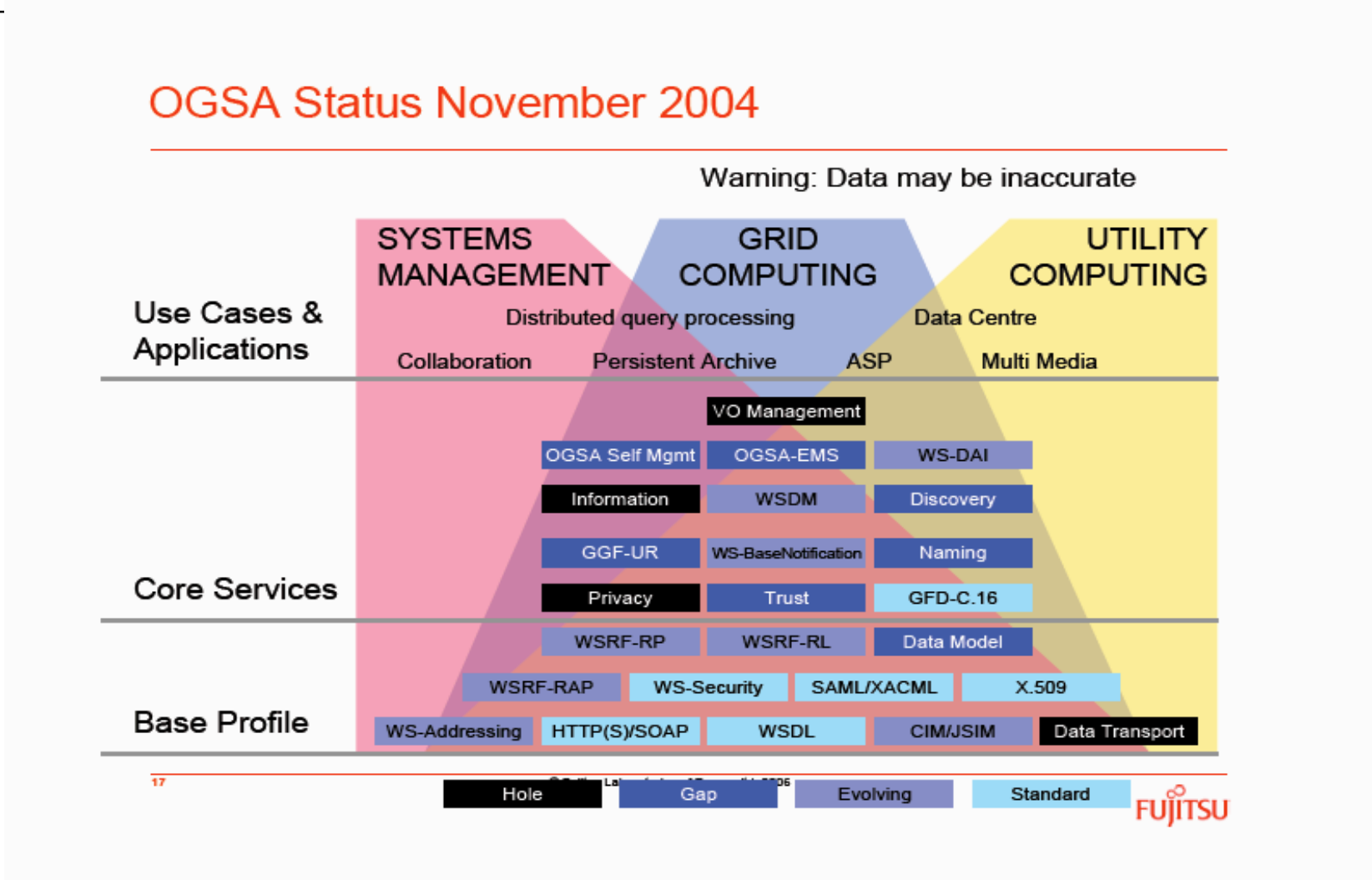
XML	A universal model for data exchange and data integration
XML Schema	Defines the schema of a XML document, makes syntactical restrictions, defines structural patterns, defines data types
XML Namespace	Provide means to avoid naming conflicts in XML documents
SOAP	Simple Object Access Protocol - A universal communication protocol
WSDL	WS Description Language - Description of the WS interfaces, parameters, etc.
WS-Policy	General-purpose model to describe the policies of a Web service
UDDI	Universal Description, Discovery and Integration A standard for publication and discovery of information
BPEL4WS	Business Process Execution Language Specification of workflow for the composition of services
WS-Notification	Defines the publish/subscribe pattern for message oriented systems
WS-Coordination	Coordination of distributed actions. Includes transaction management.
WS-Security	Secure SOAP messages
WS-Trust	Management of trust relationships
WSDM	Distributed Management of Web Services

Complementing specifications

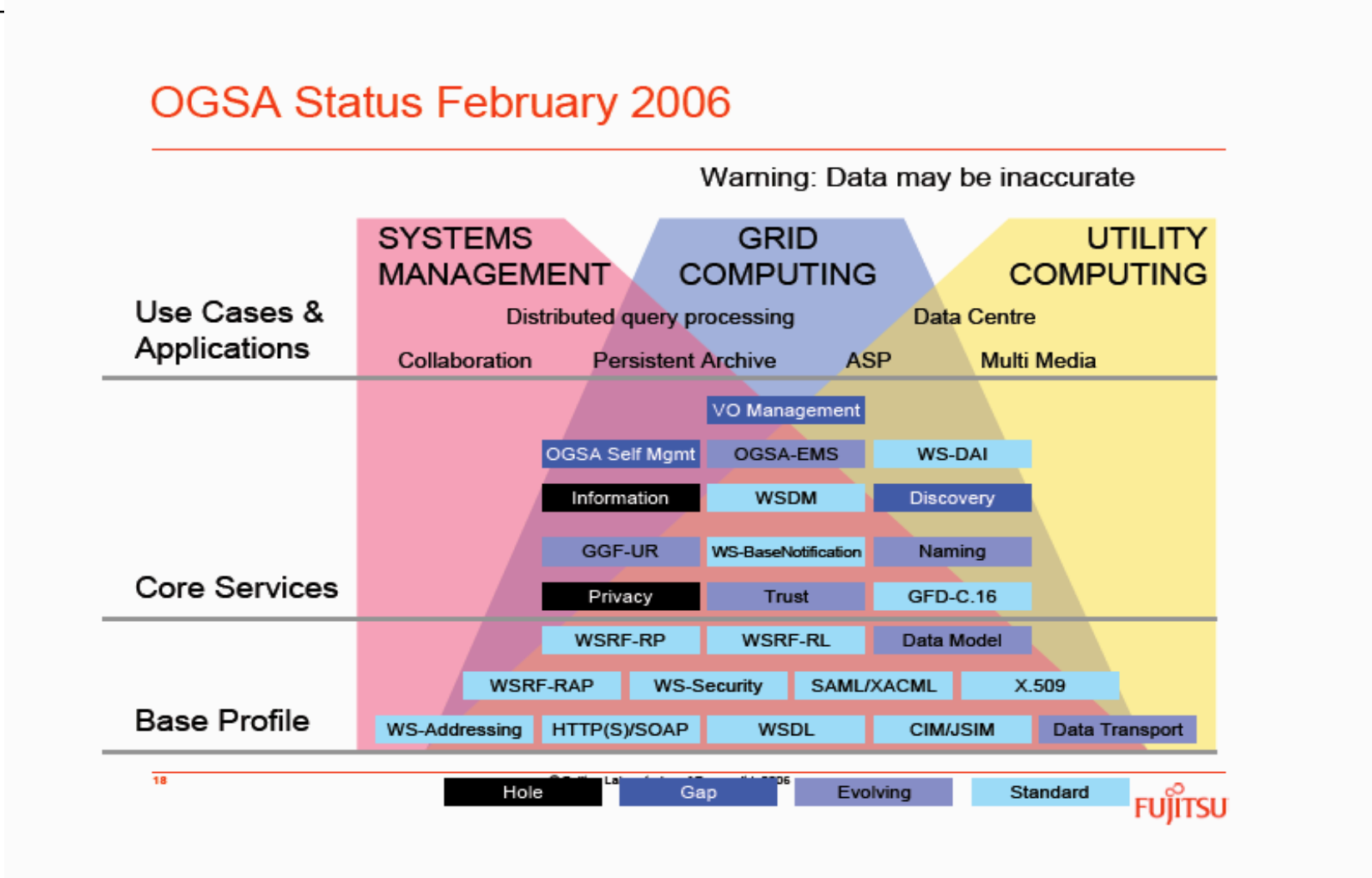
Concept	OGSI	WSRF	WS-M	RW-RT	OGSA-*
Resource		√	√	√	√
Properties	√	√	√	√	√
Notification	√	√	√		√
Lifecycle	√	√	√	√	√
Composition	√	√	√	√	√
Faults	√	√			√
Collections	√	√	√		√
Naming	√				√

Orange = Historical, Blue = Evolving, Green = Standard

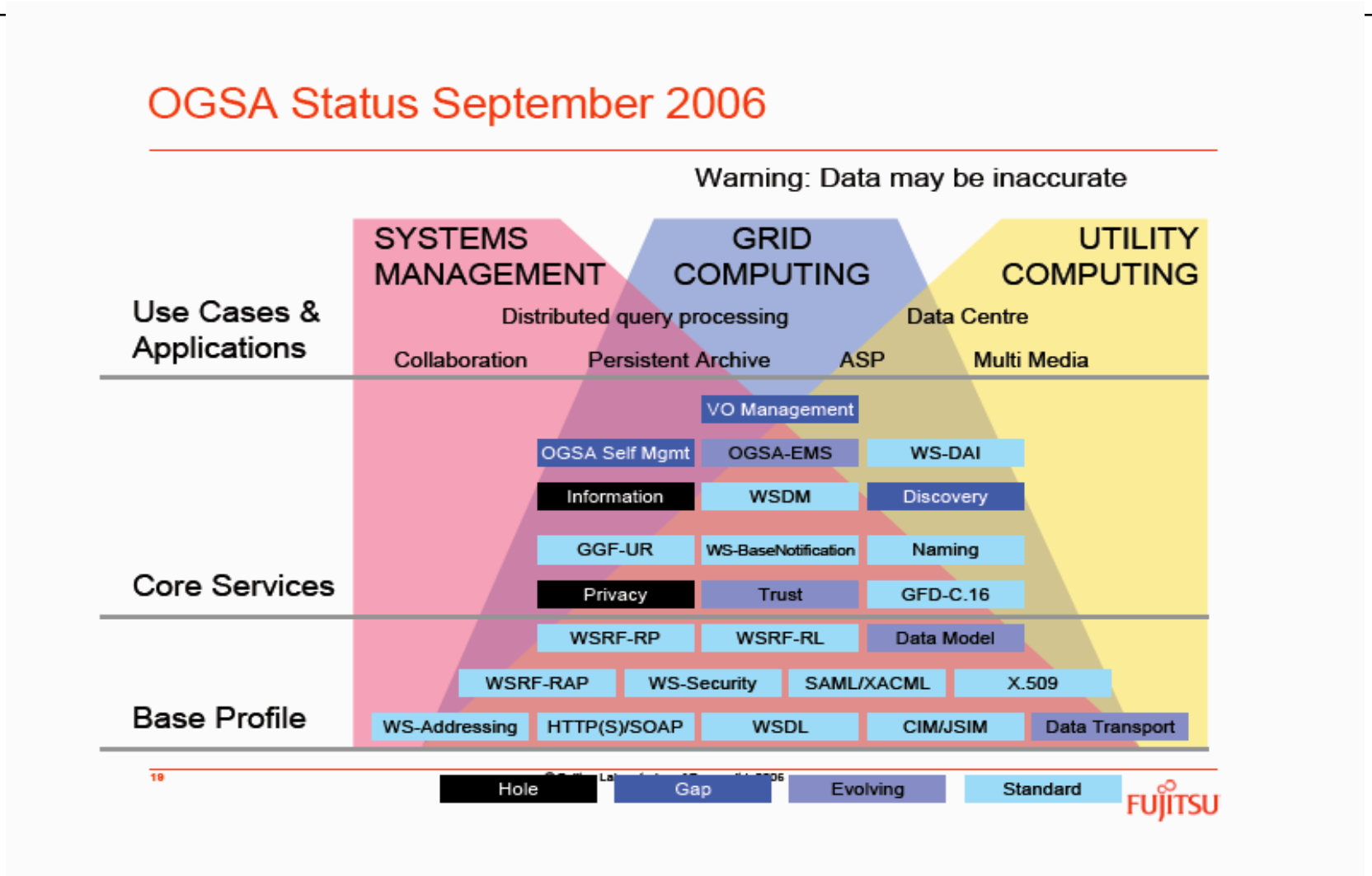
Specifications Evolution



Specifications Evolution



Specifications Evolution



Questions

Diligent

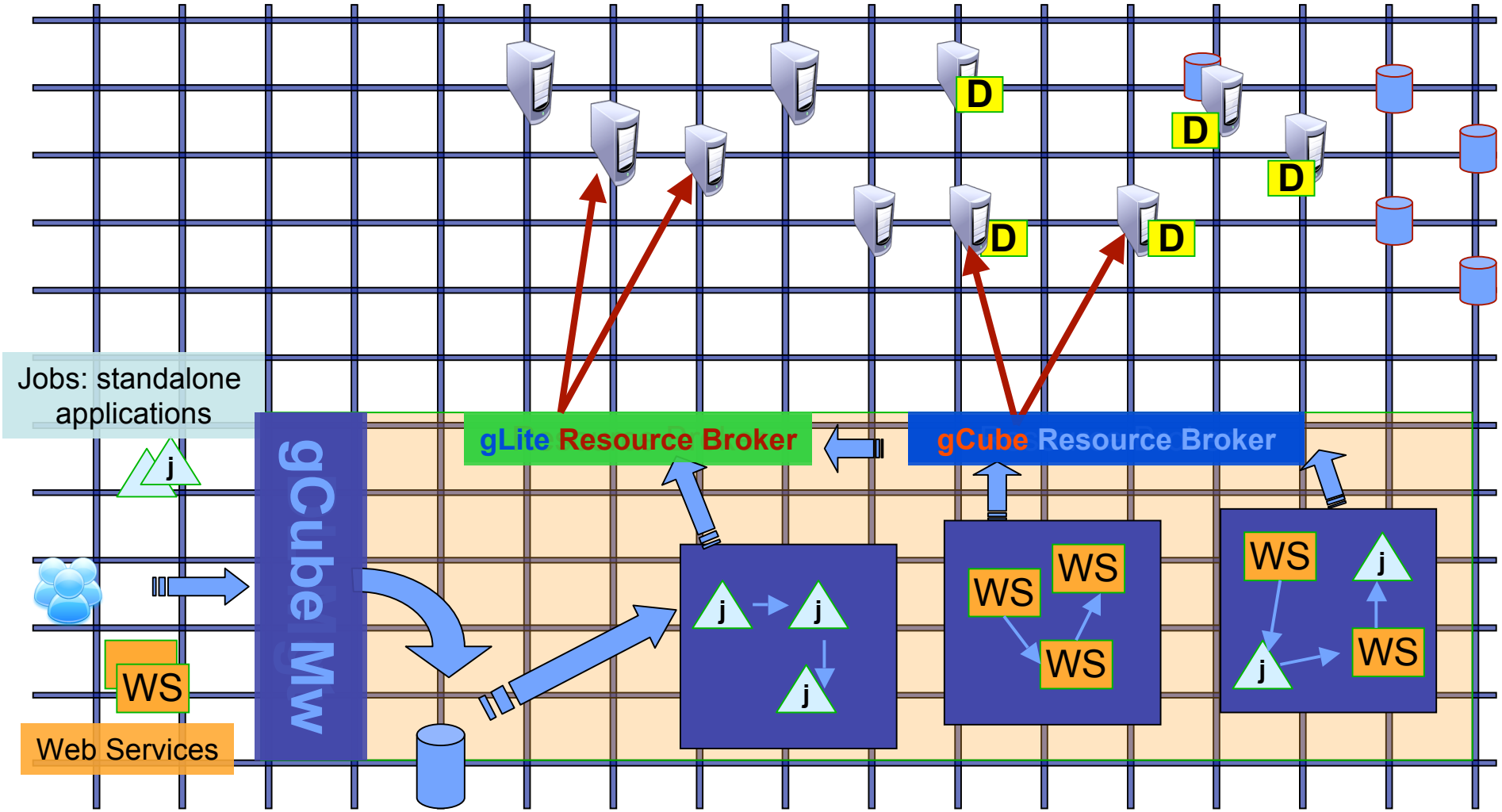
Infrastructure Management

Pasquale Pagano
CNR-ISTI

Infrastructure Management Goals

- *management of large number of distributed VOs;*
- *access to and handling of distributed multi-focused data and services;*
- *on-demand efficient processing of huge amounts of data;*
- *orchestration of user defined services, with defined QoS (e.g. scalability, reliability);*
- *knowledge preservation*
  - *storage of derived data as well as dependencies*
  - *traceability of the operations performed*

Infrastructure Management on the Grid



Infrastructure Management on the Grid

Cont.

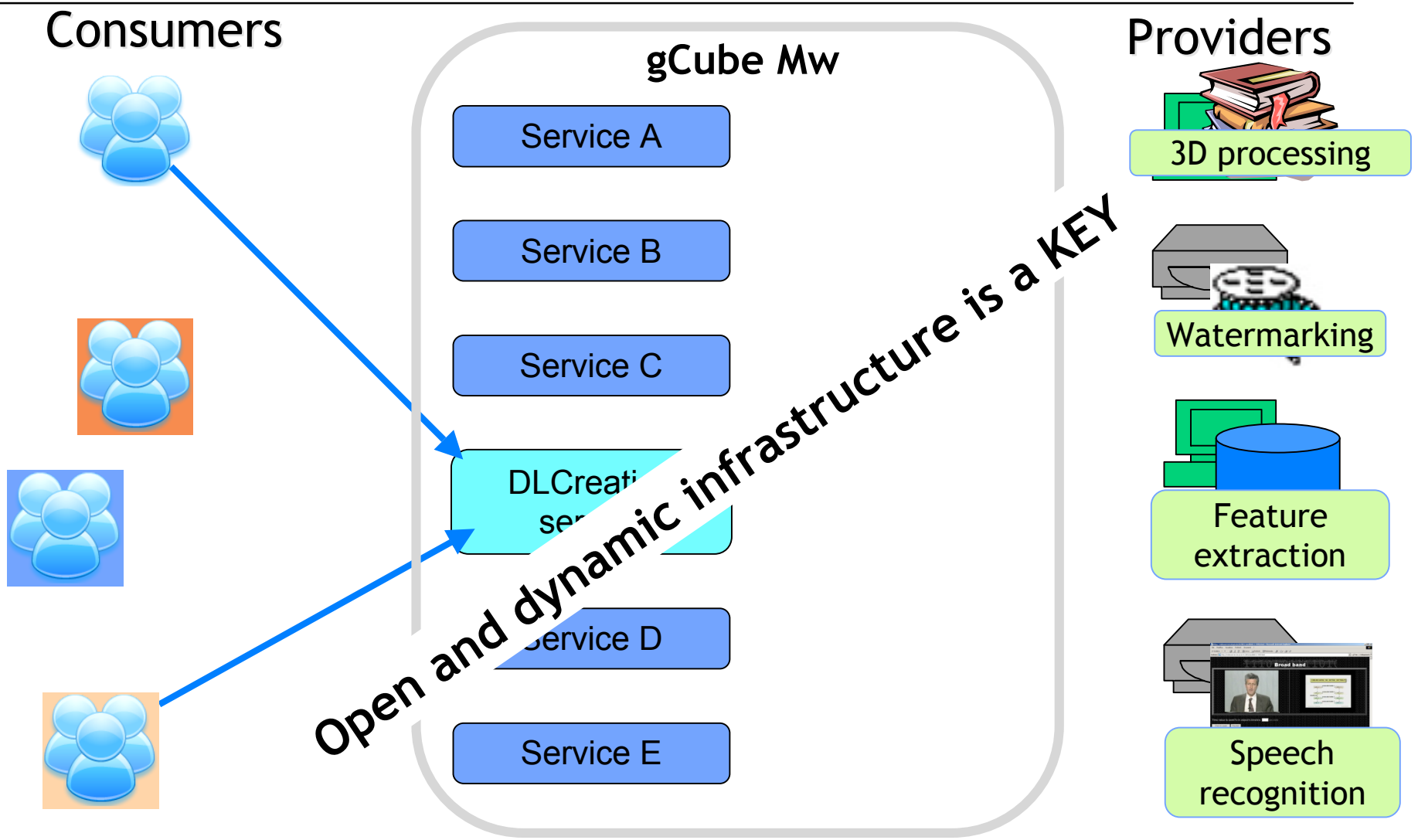
Infrastructures for distributed DL applications on the Grid are designed as a set of cooperating resources.

The infrastructure is composed by:

- the set of grid resources (computing, storage)
- the set of DL services (content and storage management, search, index, …)
- the set of processes defined to manage such resources
- the set of collections created to bring together the content to manage
- the set of enabling services (core services: Information System, Resource Orchestrator, …)

Infrastructure Management on the Grid

Cont.



Grid Infrastructures for the DL communities

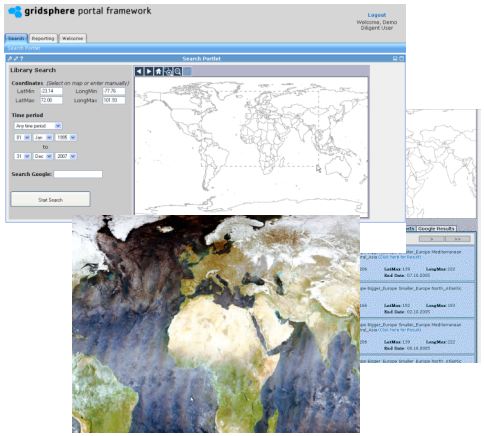
- Grid empowered infrastructures dramatically changes the **DL development model** used by distributed and dynamic organisations and communities

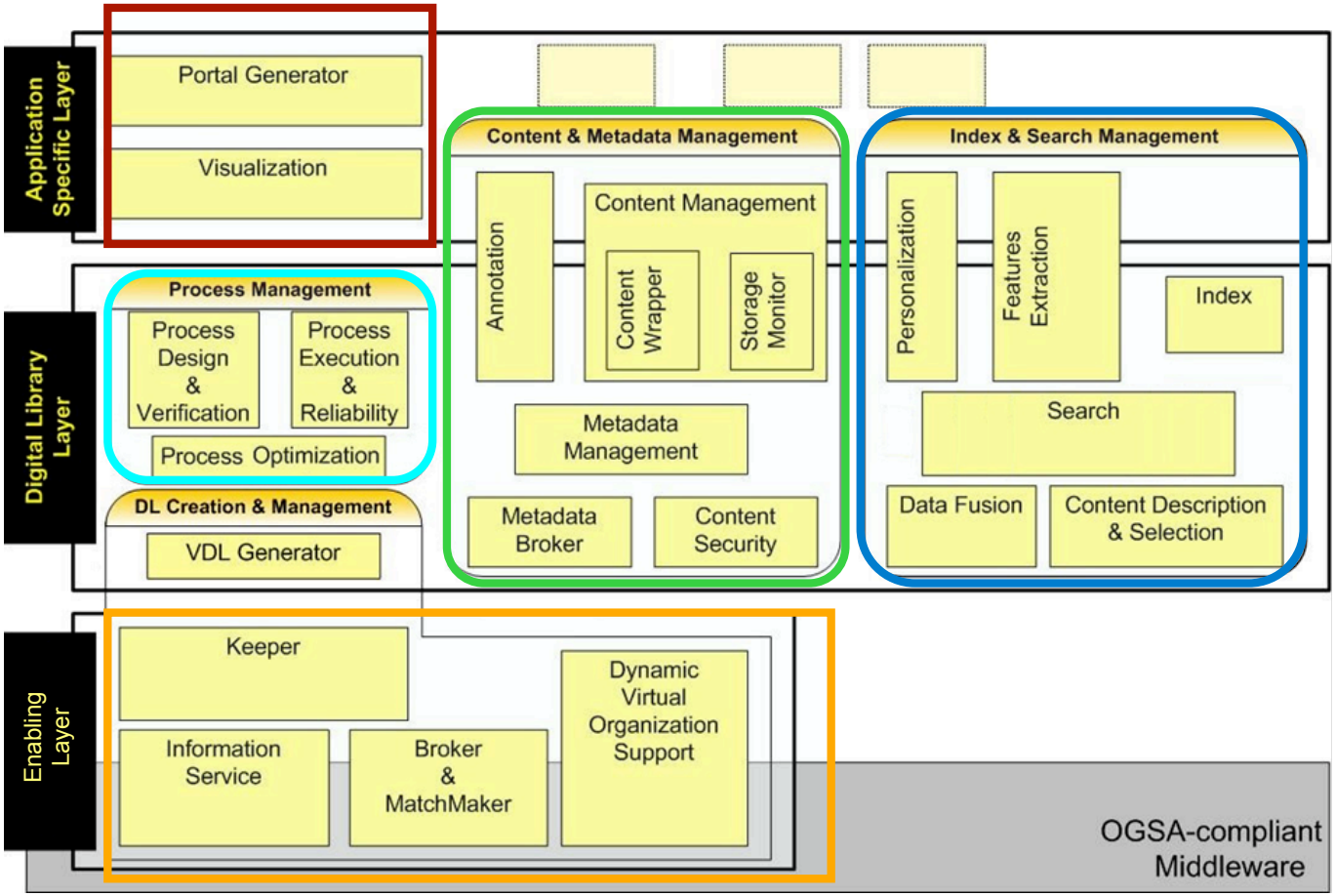
- Using Grid empowered infrastructures, the organisations and communities are able to setup their own environment:

  - When and for the time they need it
  - Exploiting existing Grid-based services
  - Accessing to and handling of distributed multi-focused data and services
  - Orchestrating user defined services, with defined QoS (wrt. scalability, reliability)
  - Profiting from a shared storage and computational set of resources
  - Sharing data and services in a collaborative and efficient way

gCube Architecture

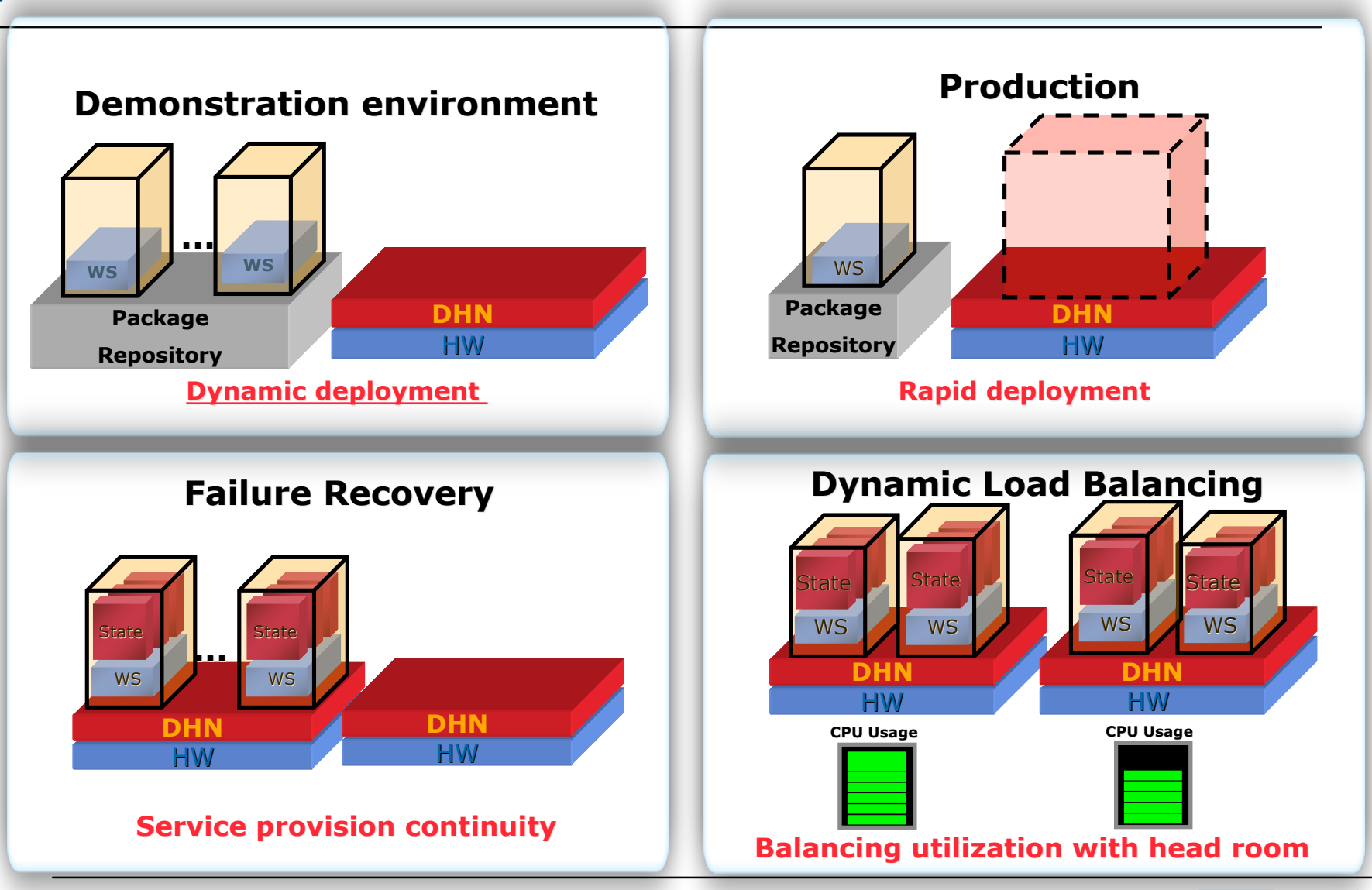
gCube Architecture





gCube: Enabling Layer

gCube about Resource Management



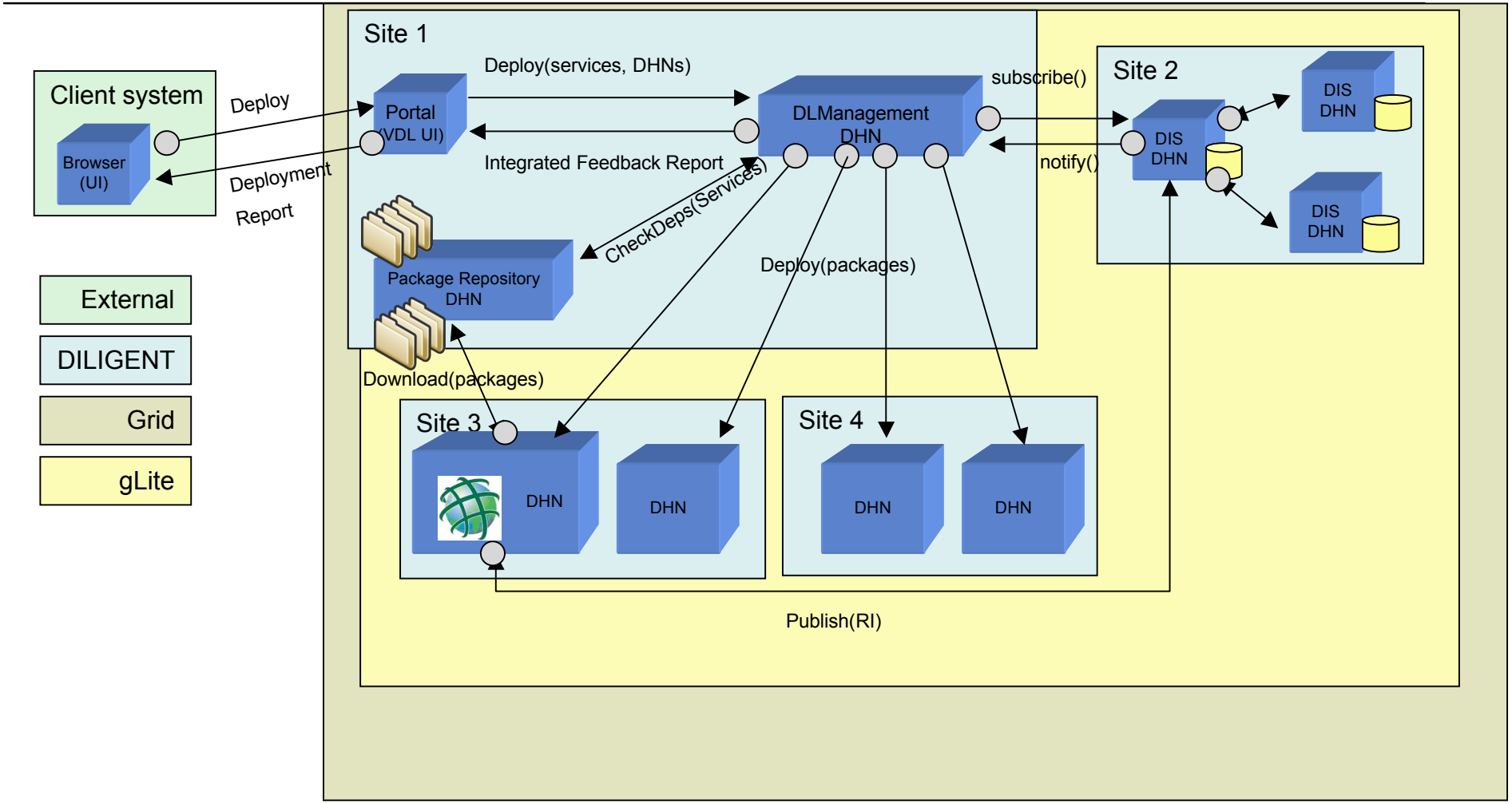
gCube about Resource Management

gCube provides resources management capabilities

- by enabling:
  - Remote deployment
  - Environment configuration
  - Lifetime management
  - Service provision continuity
  - Usage normalisation

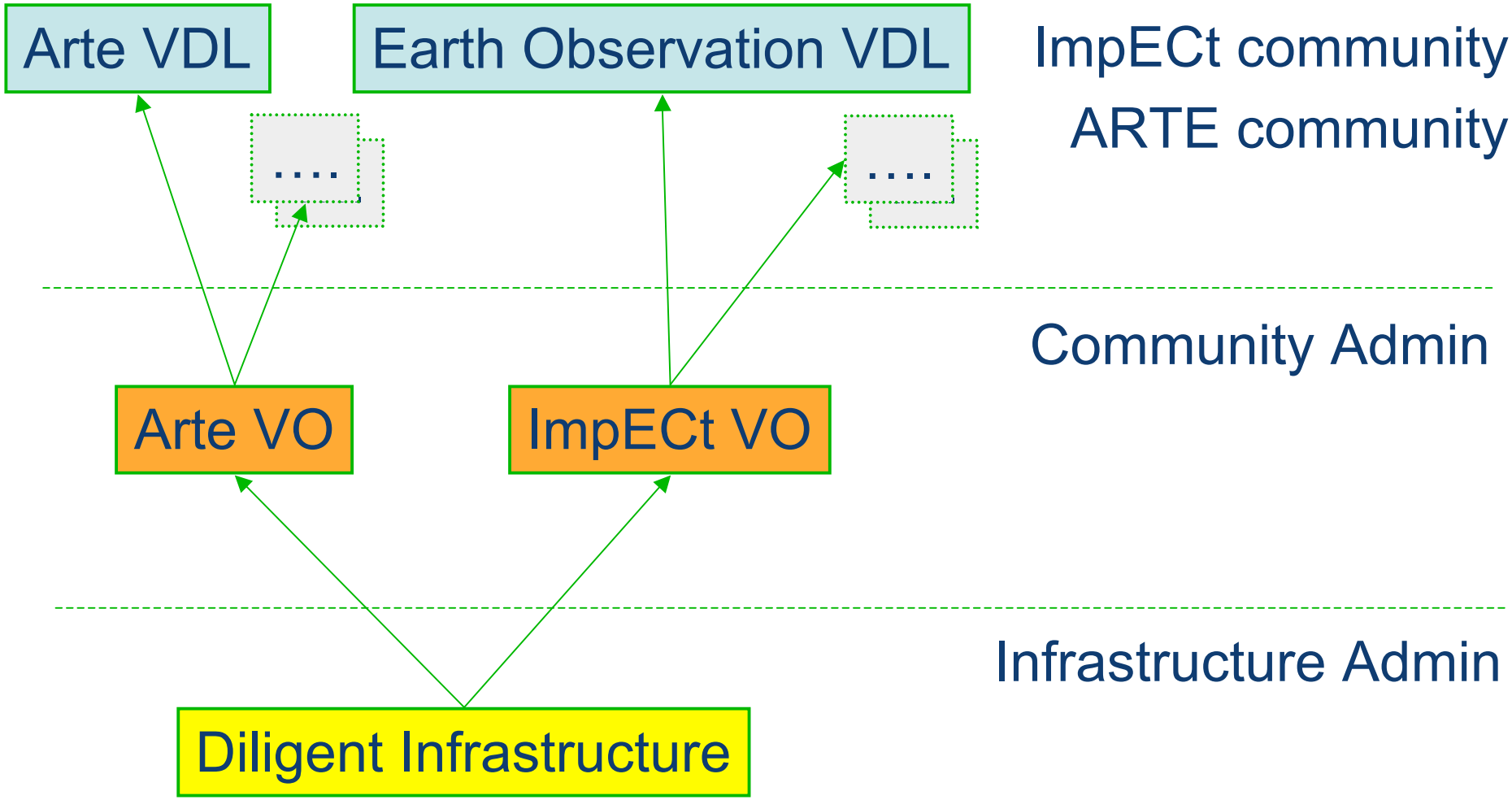
- and transparently managing:
  - Failure recovery
  - Dynamic load balancing

gCube about Resource Management

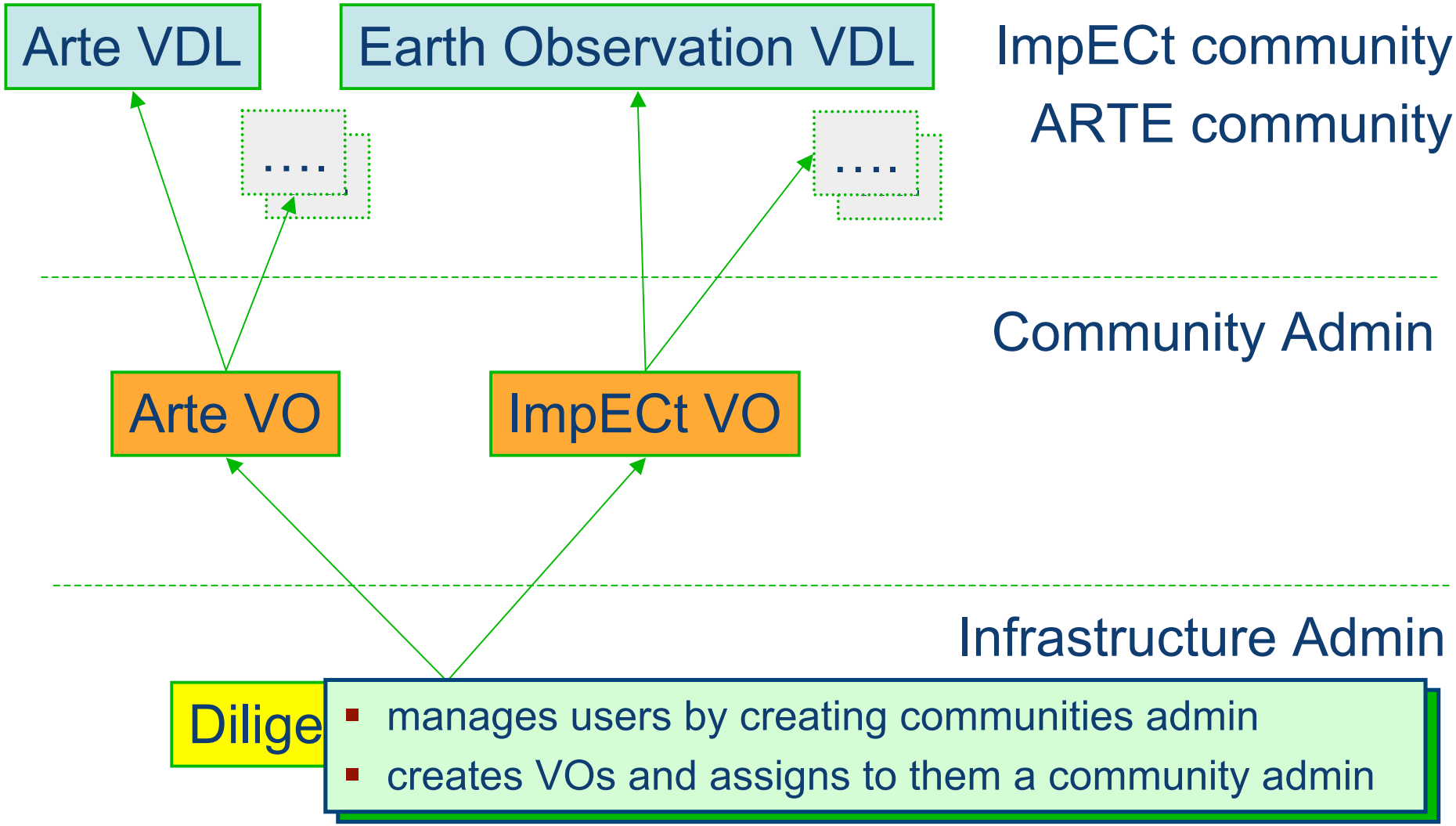


Demo

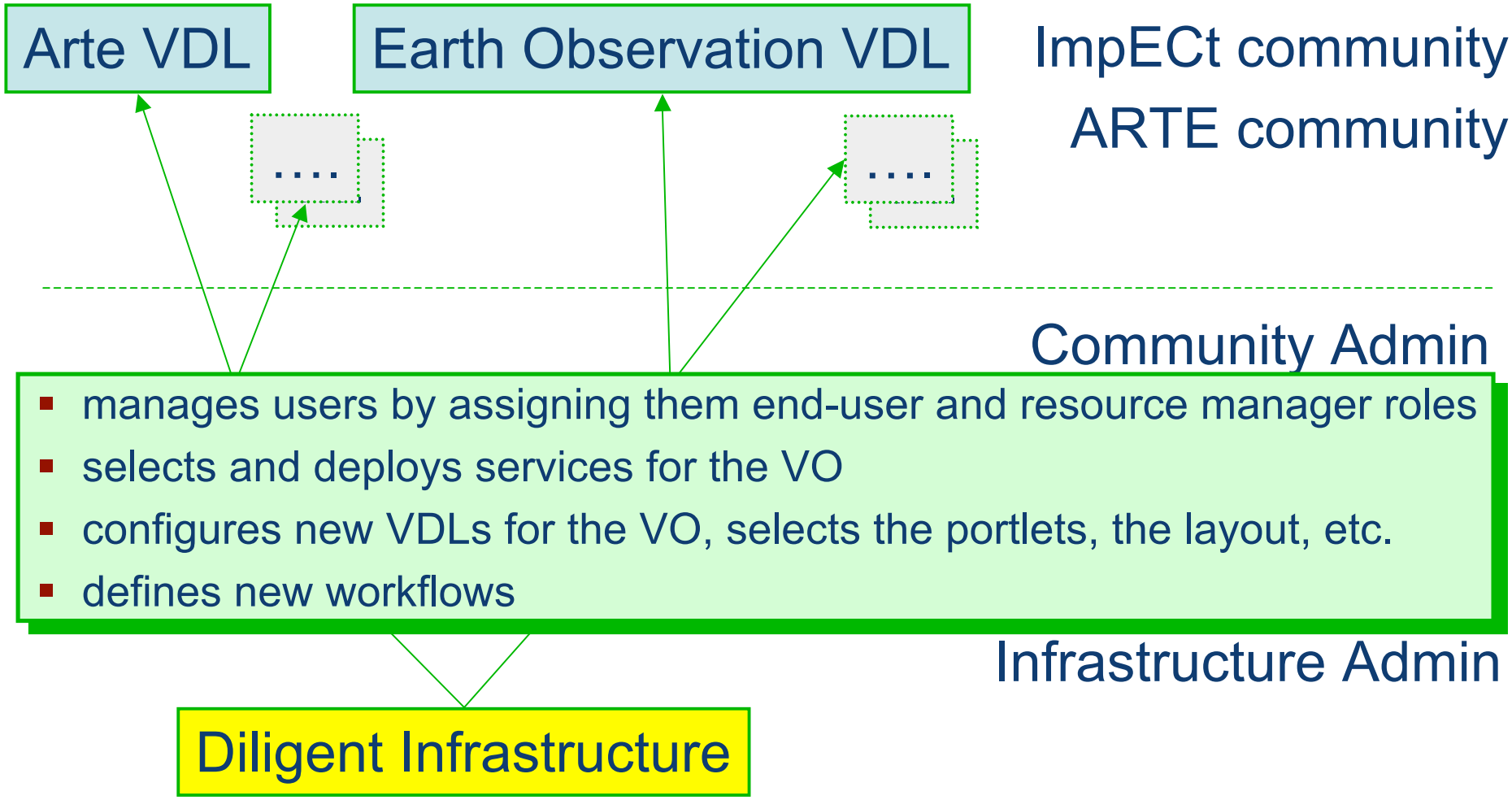
User Roles


Arte VDL
Earth Observation VDL
....
....
ImpECt community
ARTE community
Community Admin
Arte VO
ImpECt VO
Infrastructure Admin
Diligent Infrastructure

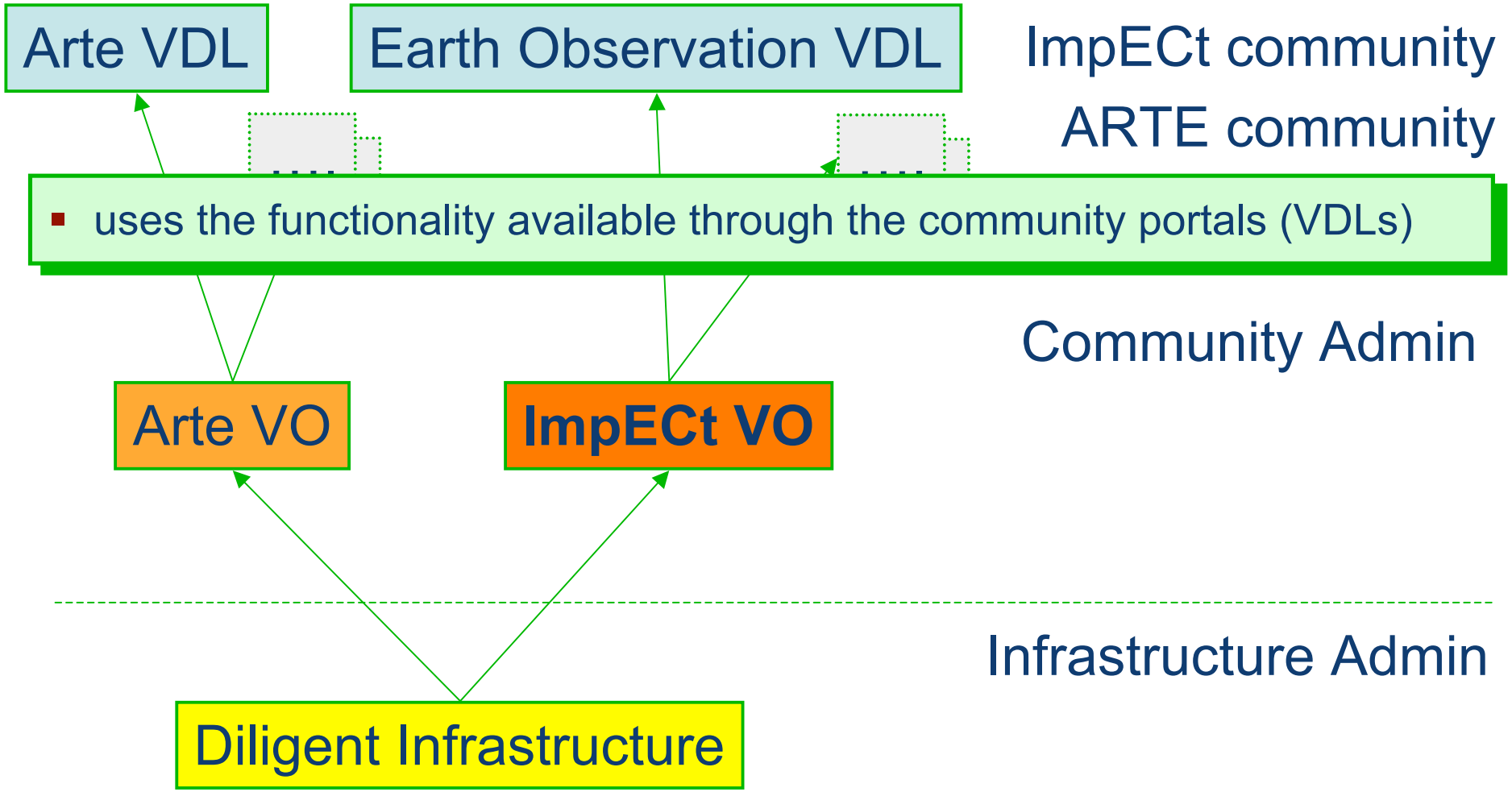
User Roles


Arte VDL
Earth Observation VDL
ImpECt community
ARTE community
....
....
Community Admin
Arte VO
ImpECt VO
Infrastructure Admin
Dilige
manages users by creating communities admin
creates VOs and assigns to them a community admin

User Roles


Arte VDL
Earth Observation VDL
ImpECt community
ARTE community
....
....
Community Admin
manages users by assigning them end-user and resource manager roles
selects and deploys services for the VO
configures new VDLs for the VO, selects the portlets, the layout, etc.
defines new workflows
Infrastructure Admin
Diligent Infrastructure

User Roles


Arte VDL
Earth Observation VDL
ImpECt community
ARTE community
uses the functionality available through the community portals (VDLs)
Community Admin
Arte VO
ImpECt VO
Infrastructure Admin
Diligent Infrastructure

Conclusion

- To promote knowledge exchange
- To increase applications interoperability

➲ to allow communities to access, aggregate and manipulate data and services from a wide variety of independently information sources and resource providers

Links

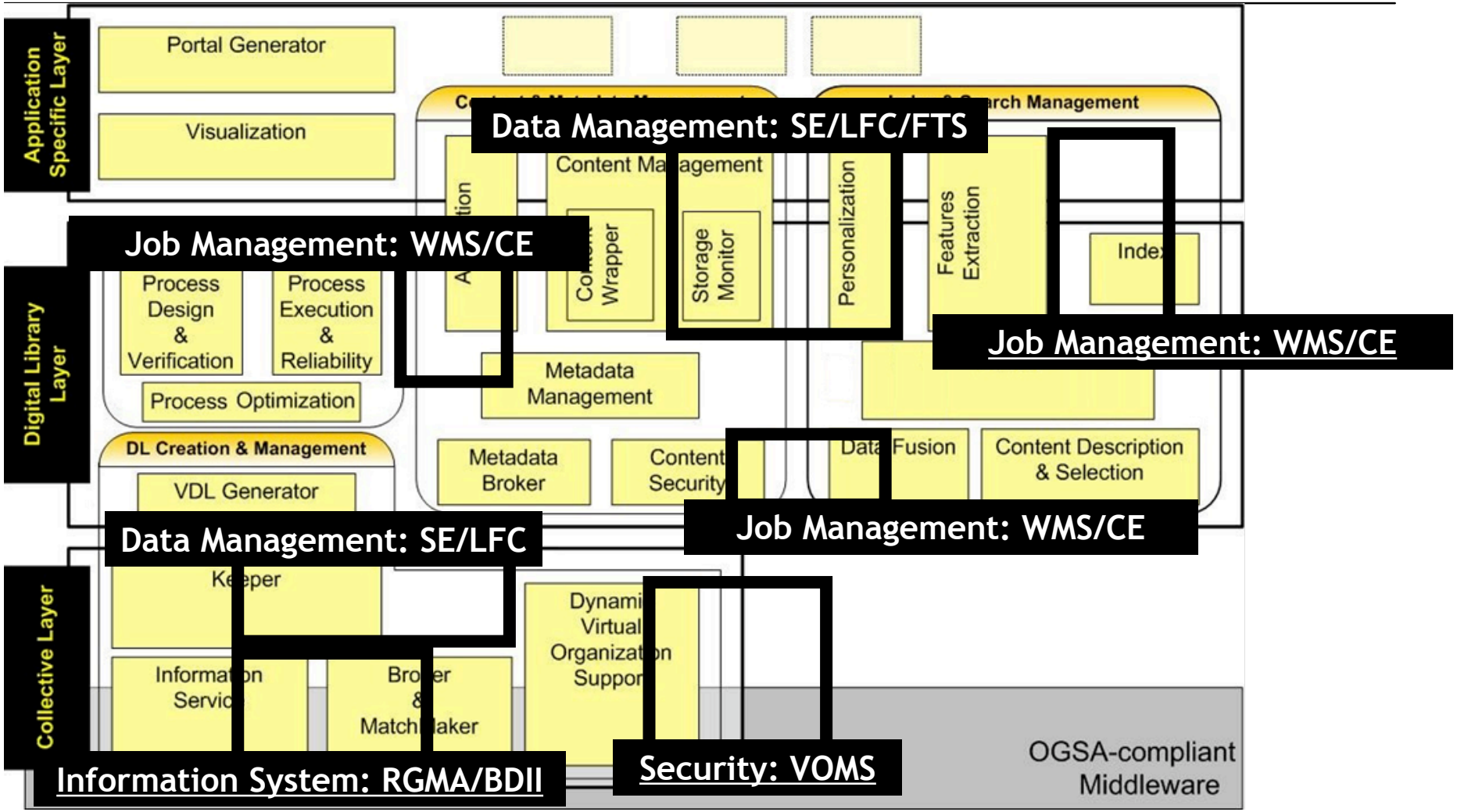
Project Web Site: www.diligentproject.org

Info: info@diligentproject.org

Software Web Site: www.gcube-system.org

Additional slides

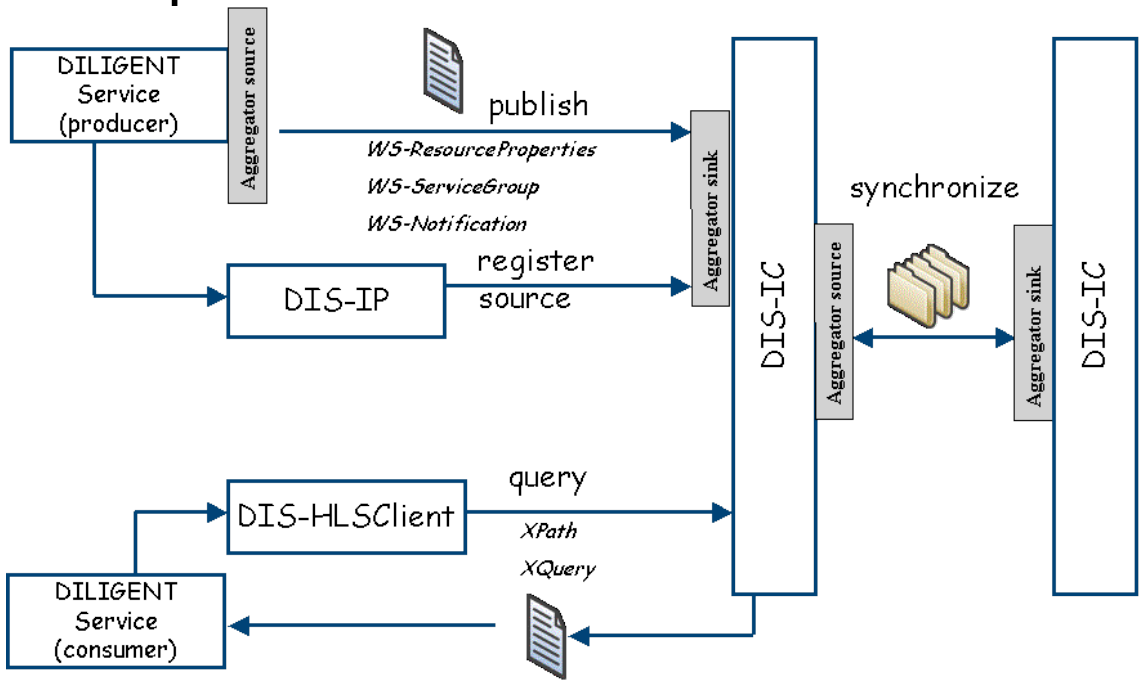
gCube WSs – gLite grid services interactions



gCube Information System

The gCube Information System (DIS) service is responsible for the aggregation, storage and monitoring of information about gCube resources. This information is provided through query and subscription interfaces.

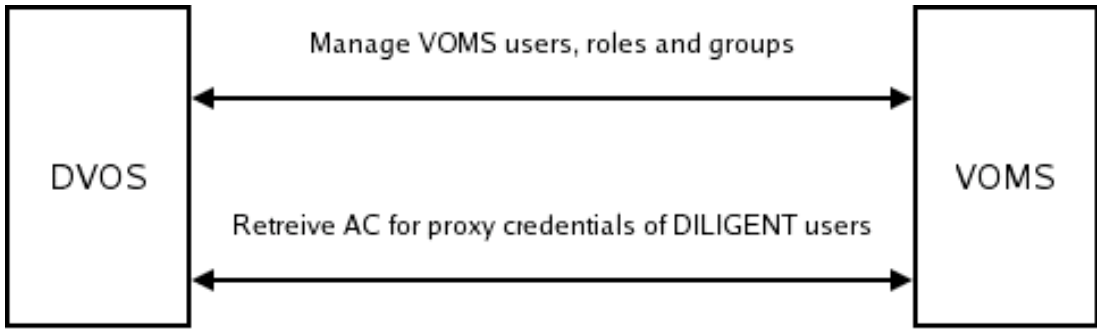
The DIS exploits gLite by aggregating information about gLite resources (SEs, CEs) from BDII.



Dynamic Virtual Organization Support

The Dynamic Virtual Organization Support (DVOS) service provides a robust and flexible security framework based on an advanced authentication and authorization model. It also offers notification support and dynamic aggregation of resources and users/groups

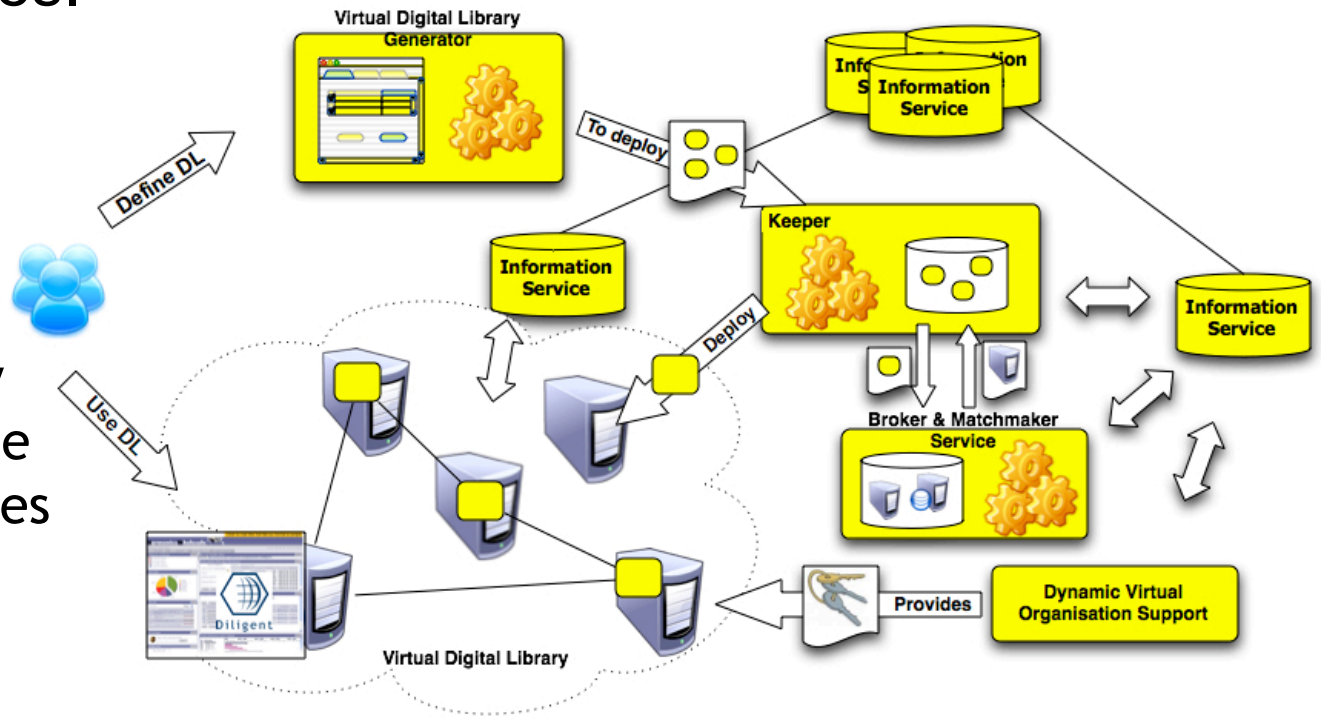
The DVOS exploits gLite by relying in VOMS for its AuthZ service



Keeper

The Keeper service is responsible for the creation of DLs. This includes the management of the gCube packages, the selection of the appropriate gCube resources and the dynamic deployment, monitoring and re-allocation of gCube services.

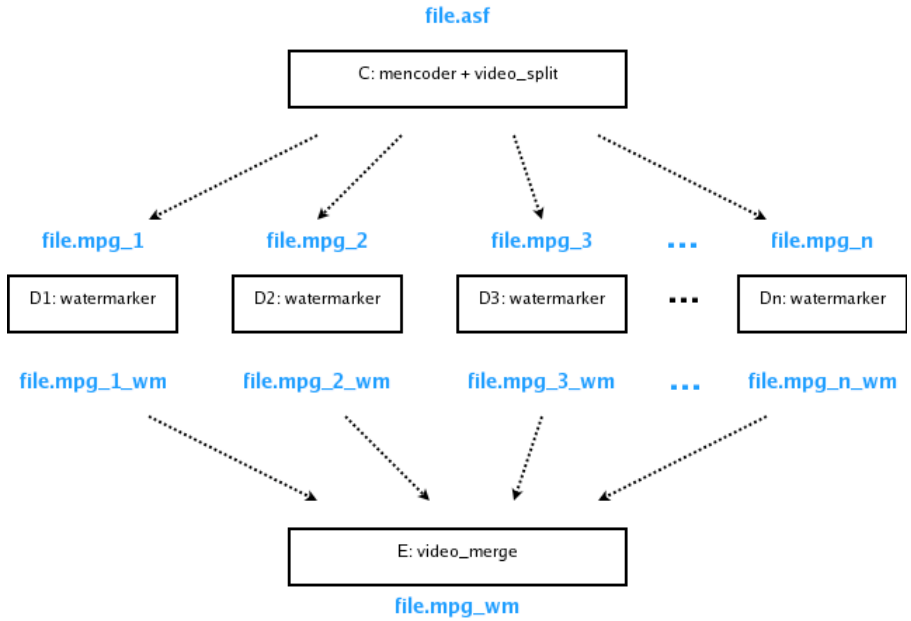
The Keeper exploits gLite by storing the gCube software packages in gLite SE.



Content Security

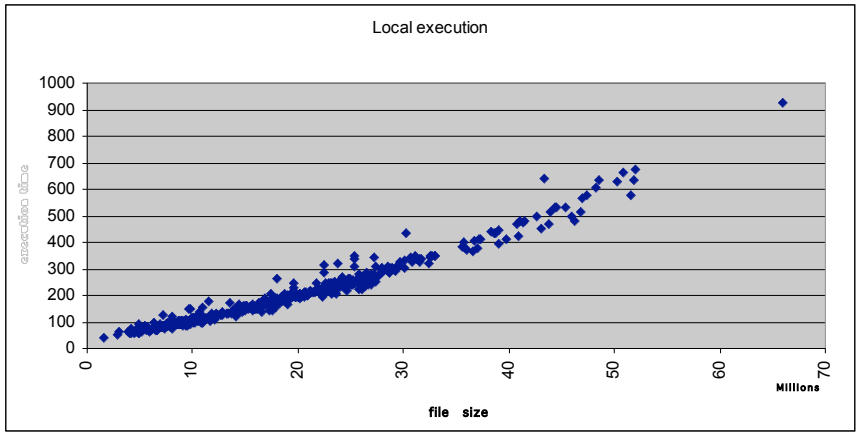
The Content Security service deals with the specialties of protecting multimedia content providing authenticity, integrity and confidentiality to the gCube data through the execution of grid-enabled watermarking algorithms for different media types.

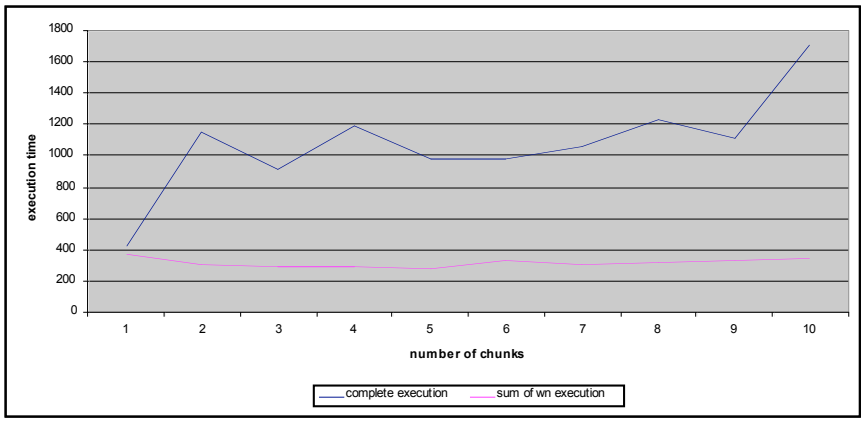
- The Content Security exploits gLite by executing a gridified version of the watermarking application in gLite CEs.



Content Security

- Analysis of remote execution - Strategy
  - Conversion from asf to mpeg format
  - Splitting to n independent chunks
  - Gridified parallel watermarking of every chunk
  - Merging of marked chunk to complete video
- Local Execution
  - Dual + HT Intel(R) Xeon(TM) CPU 2.80GHz
  - Linear execution of watermark embedding in video collection
  - Completion time: 28.9 hours
- Grid Execution
  - Distribution of watermark embedding submission to 39 worker nodes
  - 535 successful, 86 resubmissions
  - Completion time: 2.6 hours





Diligent

gCube: Content Management

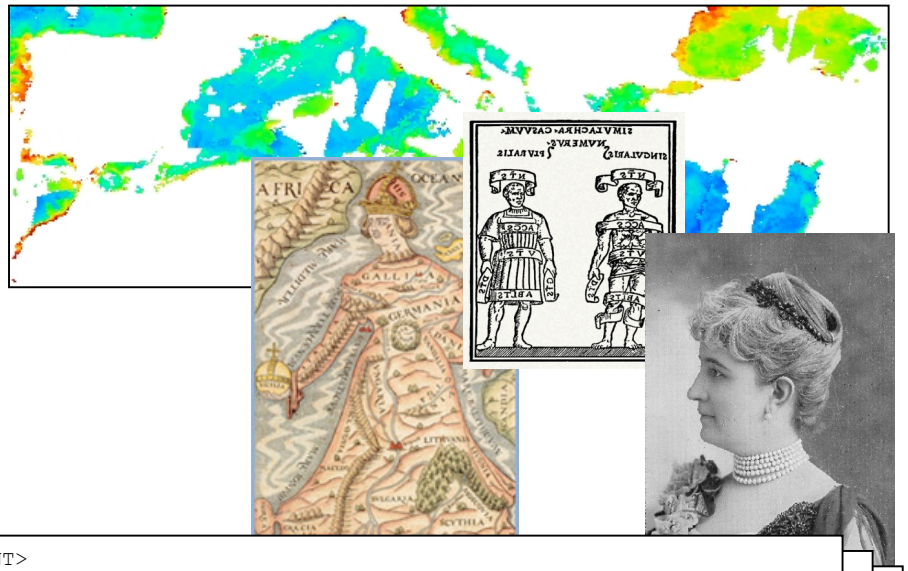
Heiko Schuldt
UNIBAS

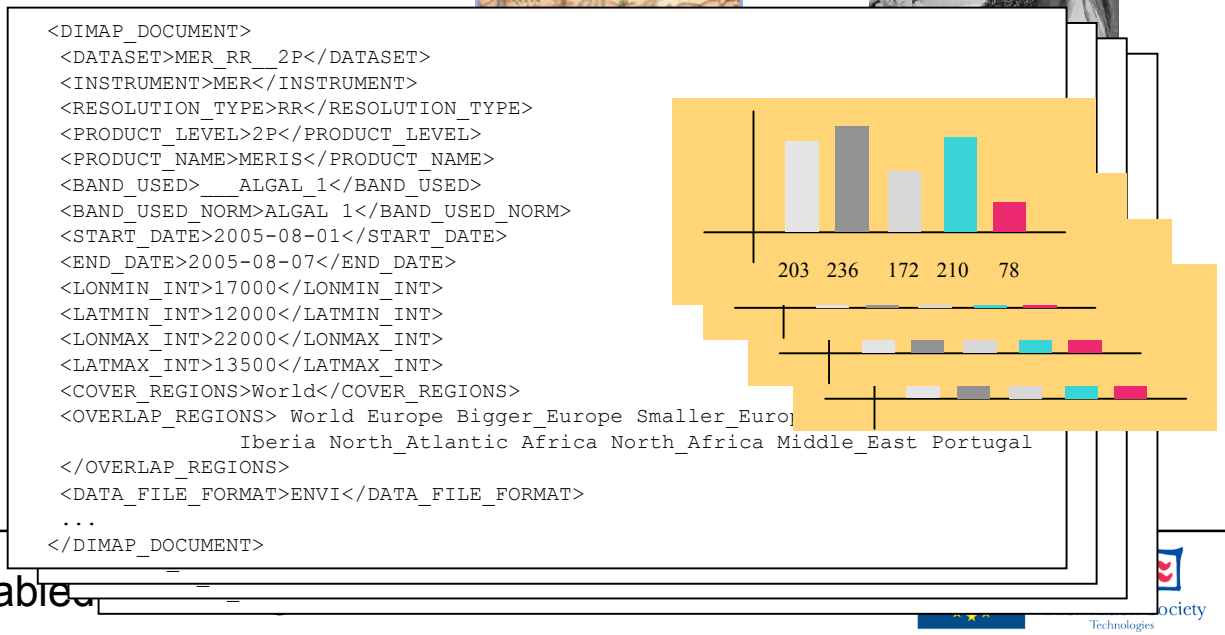
Information Society Technologies

Content Management: Challenges

- Store large volumes of digital content in the Grid

- But there is much more:
  - Metadata for each object
  - Automatically extracted features per object (e.g., color histo-grams for images)
  - Storage properties (e.g., size, etc.)
  - … and all that highly interconnected





```
<DIMAP_DOCUMENT>
 <DATASET>MER_RR__2P</DATASET>
 <INSTRUMENT>MER</INSTRUMENT>
 <RESOLUTION_TYPE>RR</RESOLUTION_TYPE>
 <PRODUCT_LEVEL>2P</PRODUCT_LEVEL>
 <PRODUCT_NAME>MERIS</PRODUCT_NAME>
 <BAND_USED>___ALGAL_1</BAND_USED>
 <BAND_USED_NORM>ALGAL 1</BAND_USED_NORM>
 <START_DATE>2005-08-01</START_DATE>
 <END_DATE>2005-08-07</END_DATE>
 <LONMIN_INT>17000</LONMIN_INT>
 <LATMIN_INT>12000</LATMIN_INT>
 <LONMAX_INT>22000</LONMAX_INT>
 <LATMAX_INT>13500</LATMAX_INT>
 <COVER_REGIONS>World</COVER_REGIONS>
 <OVERLAP_REGIONS> World Europe Bigger_Europe Smaller_Euro
                   Iberia North_Atlantic Africa North_Africa Middle_East Portugal
 </OVERLAP_REGIONS>
 <DATA_FILE_FORMAT>ENVI</DATA_FILE_FORMAT>
 ...
</DIMAP_DOCUMENT>
```

Content Management: Introduction

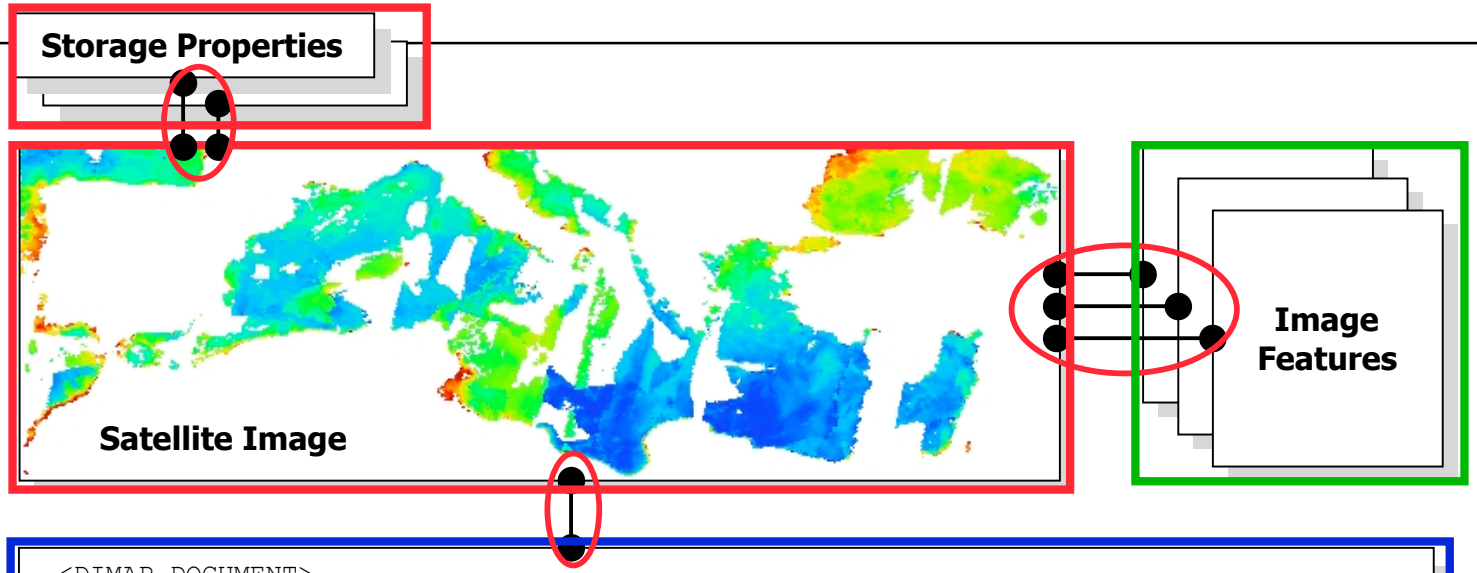
- Content and Storage Management in gCube provides means for
  - Persistently storing, and
  - Physically structuring

  any content in the Grid-enabled Digital Library
- Current Grid technology focuses mostly on performance for batch processing. But: a DL also requires
  - more than file-system-like functionalities to manage content, e.g.,
    - Content must be interrelated in multiple ways
    - and enriched with various application specific attributes
  - real-time performance while searching and browsing the content
- All content management-related functionality is provided as service

Sample Document (Earth Observation)

Diligent

Content & Storage Management



Feature Extraction

Metadata as XML Document

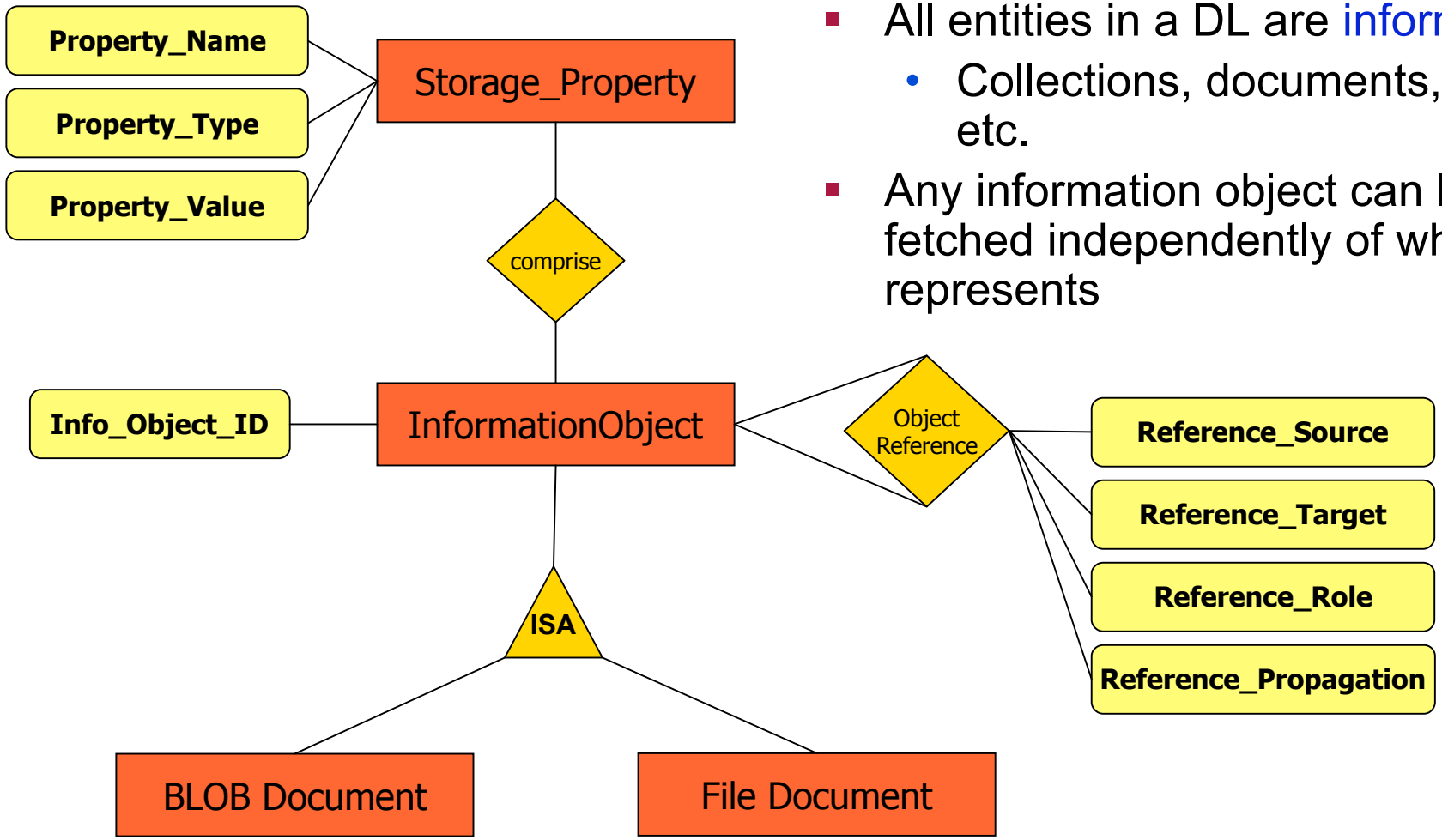
```
<DIMAP_DOCUMENT>
 <DATASET>MER_RR__2P</DATASET>
 <INSTRUMENT>MER</INSTRUMENT>
 <RESOLUTION_TYPE>RR</RESOLUTION_TYPE>
 <PRODUCT_LEVEL>2P</PRODUCT_LEVEL>
 <PRODUCT_NAME>MERIS</PRODUCT_NAME>
 <BAND_USED>___ALGAL_1</BAND_USED>
 <BAND_USED_NORM>ALGAL 1</BAND_USED_NORM>
 <START_DATE>2005-08-01</START_DATE>
 <END_DATE>2005-08-07</END_DATE>
 <LONMIN_INT>17000</LONMIN_INT>
 <LATMIN_INT>12000</LATMIN_INT>
 <LONMAX_INT>22000</LONMAX_INT>
 <LATMAX_INT>13500</LATMAX_INT>
 <COVER_REGIONS>World</COVER_REGIONS>
 <OVERLAP_REGIONS> World Europe Bigger_Europe Smaller_Europe Mediterranean
            Iberia North_Atlantic Africa North_Africa Middle_East Portugal
 </OVERLAP_REGIONS>
 <DATA_FILE_FORMAT>ENVI</DATA_FILE_FORMAT>
 ...
</DIMAP_DOCUMENT>
```

Metadata Management

Building D

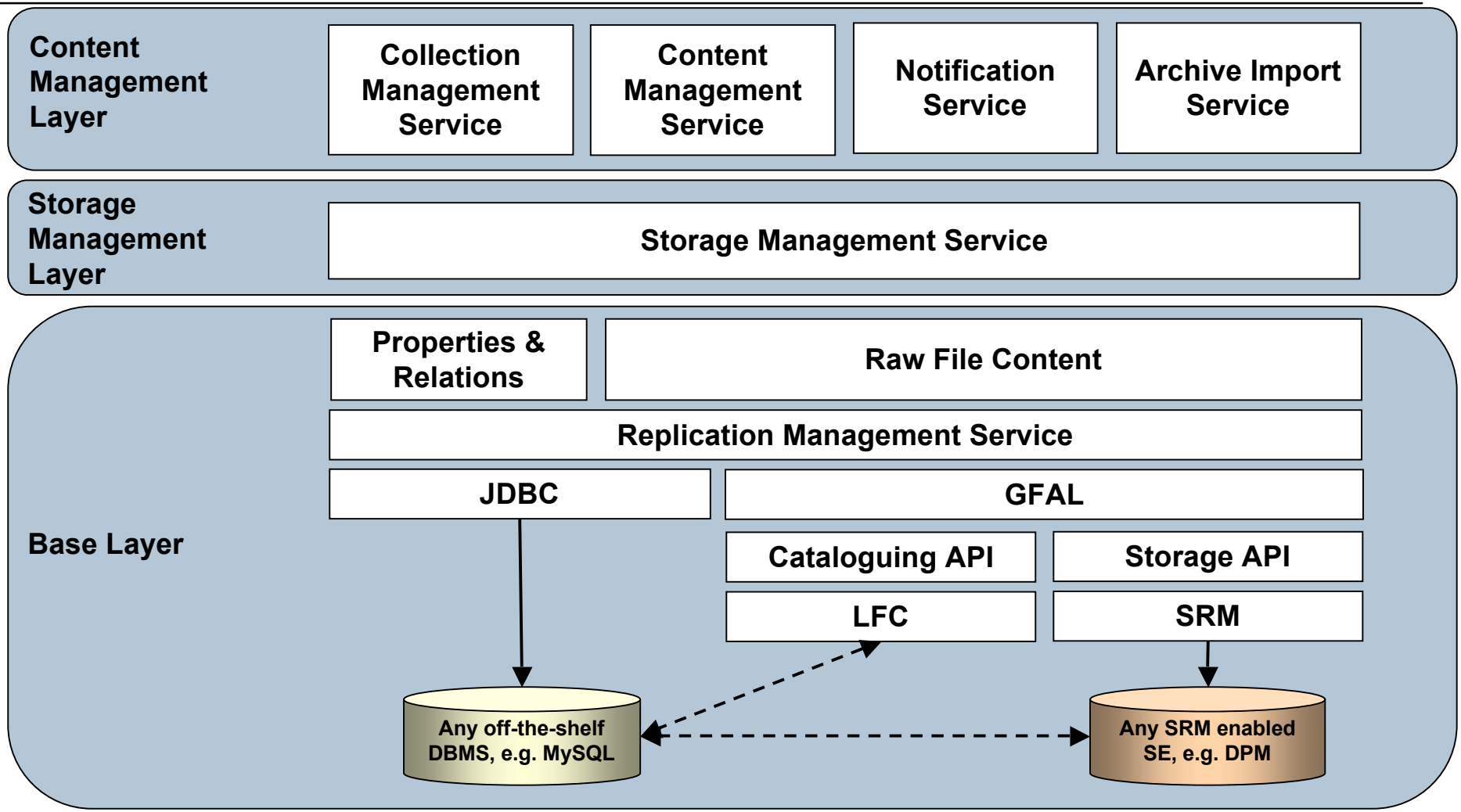
Information Society Technologies

Document Model



- All entities in a DL are information objects
  - Collections, documents, metadata etc.
- Any information object can be stored or fetched independently of what it represents

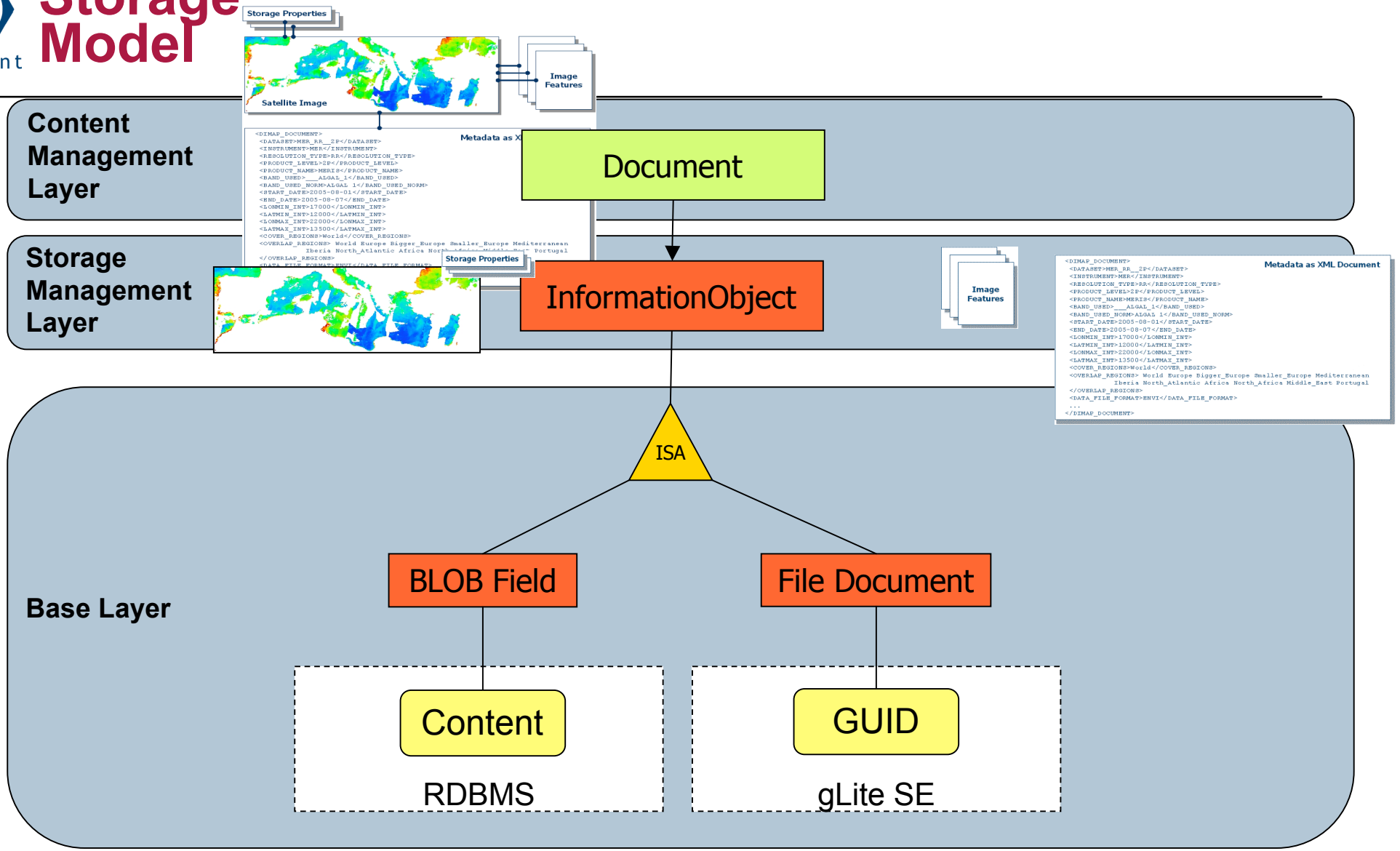
Layered Data Management Architecture



Grid (gLite) Storage

- LFC: **L**CG **F**ile **C**atalog
  (LCG: **L**HC **C**omputing **G**rid / LHC: **L**arge **H**adron **C**ollider)
  - Centralized catalog for storing locations of files stored in the grid
  - Complete catalog can be replicated
- SRM: **S**torage **R**esource **M**anager. Interface to
  - Copy a file on a storage element
  - Gather information about a file stored into a storage element (SE)
  - Remove a file from a SRM storage
  - Retrieve information about a SRM managed storage.
- GFAL: **G**rid **F**ile **A**ccess **L**ibrary
  - provides calls for catalog interaction, storage management and file access and can be very handy when an application requires access to some part
- DPM: **D**isk **P**ool **M**anager
  - APIs for accessing local storage

Storage Model

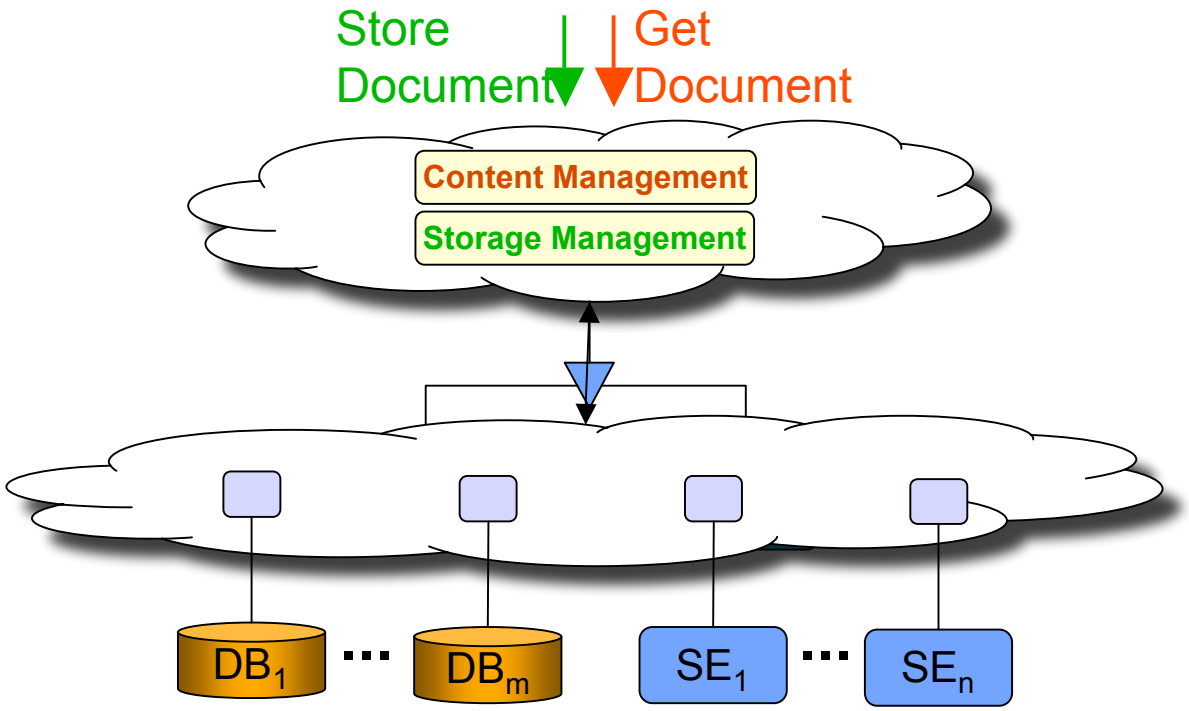


Where to Store What?

- An information object can be stored either in the relational database management system (RDBMS) or in the Storage Element (SE)
  - The RDBMS is a natural choice to store
    - properties of documents and
    - relations between documents
  - Decision about storage (database or Grid storage) for raw content
    can be based on heuristics, e.g.,
    - Store files always in a SE
    - Store files larger than 2 MB in SE

Replication Management Service

- Replication Management service is an internal service of the Base Layer
- Provides transparent access to data stores
- Goal: increase the degree of availability of content



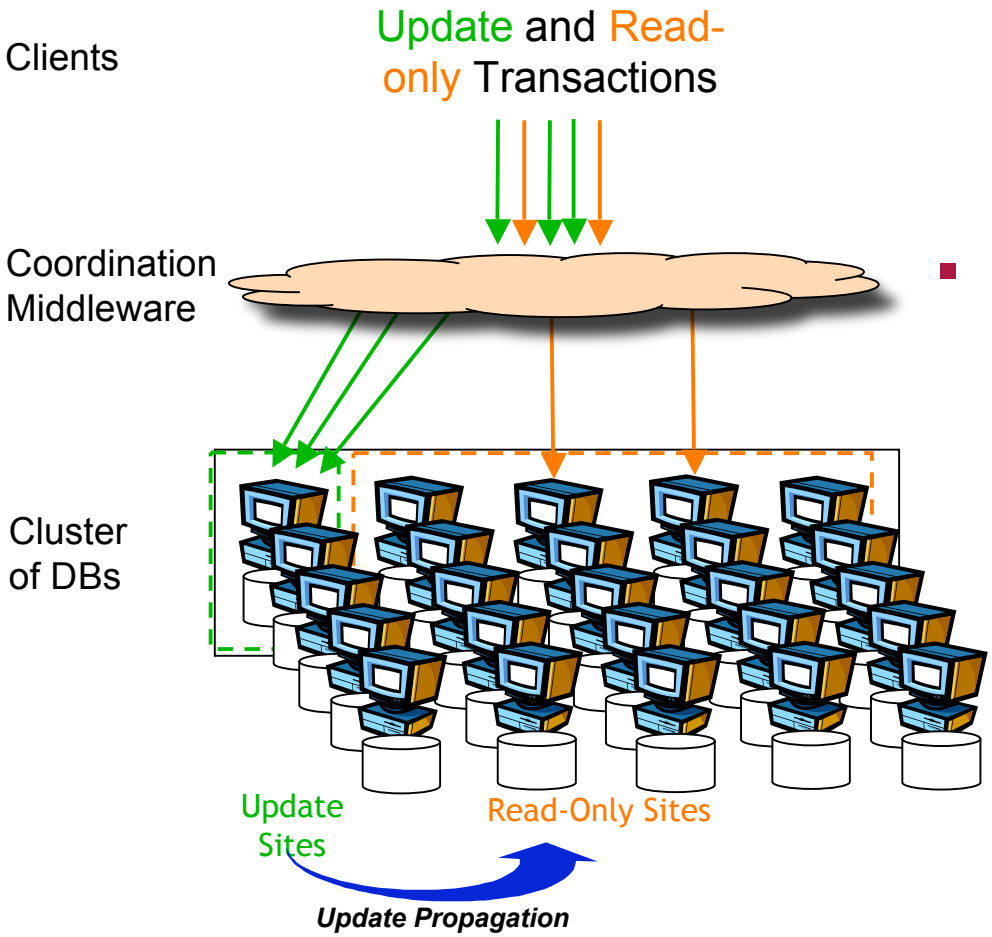
Content Management: Replication …

- Replication
  - Fully or partially duplicates data objects (e.g., documents) among the nodes of a distributed system
- Replication management: responsible for the maintenance of replicas
  - Ensures consistency of multiple copies of the same data object
  - Usually distinction between master site (original –and updateable– copy) and slave sites (replica)
  - Basic approaches for maintaining replicas : eager vs. lazy
    - Eager replication: synchronizes replicas within the boundaries of the update transaction
    - Lazy replication: decouples synchronization from the updating transaction (replication done in separate transaction)

… Content Management: Replication

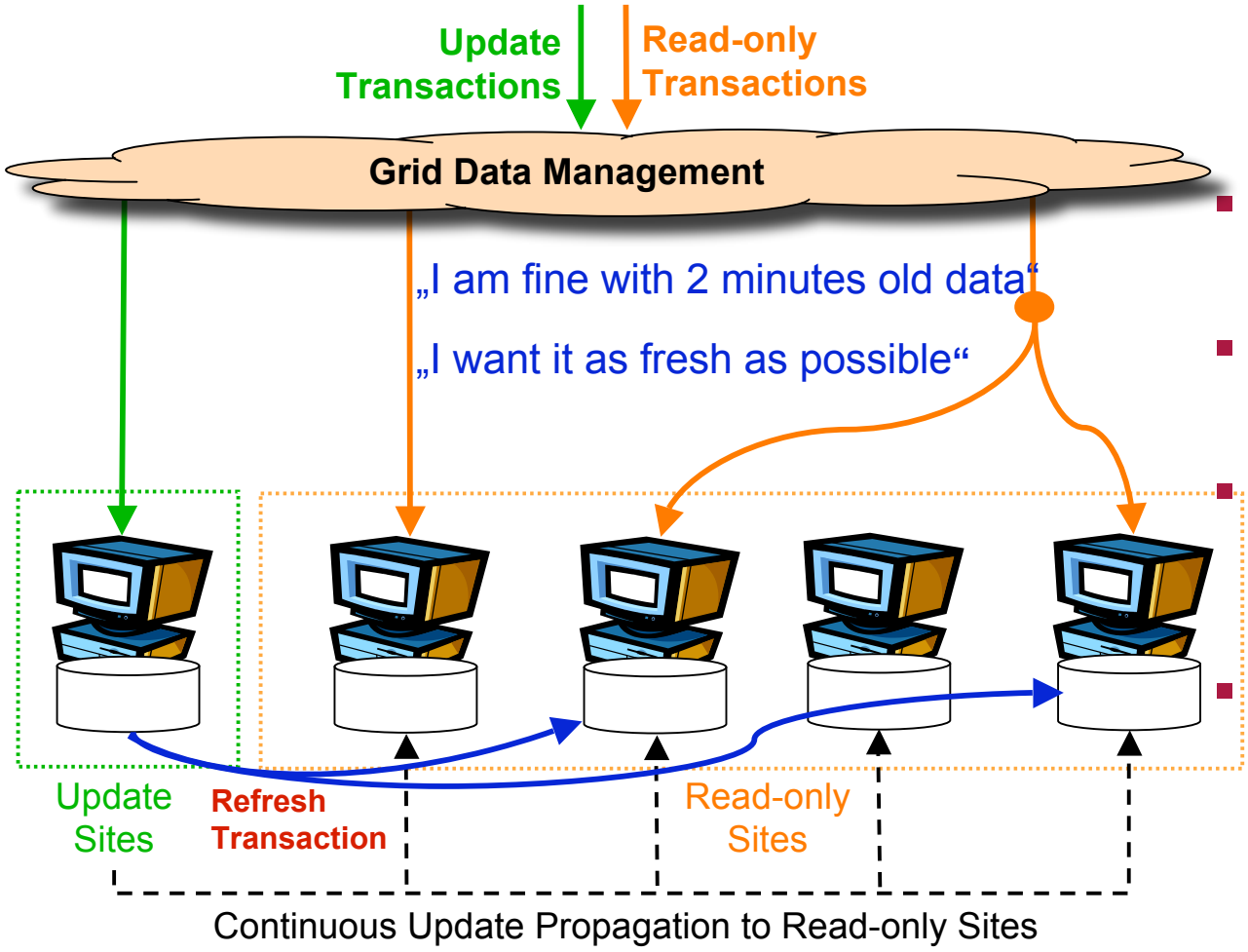
- Replication support in the Grid is currently still underdeveloped
  - Just creating copies of documents without sophisticated maintenance
- Goal in DILIGENT: Apply a sophisticated replication protocol designed for database clusters to the Grid
  - Ensure consistency of multiple copies of the same data object
  - Support data with different degrees of freshness
  - Guarantee consistent reads for all objects of a collection
    - Handle replication internally lazily, although being eager from the user's perspective
  - Adjust to changing load by dynamically creating new replicas

Example: Replication in a Cluster of Databases



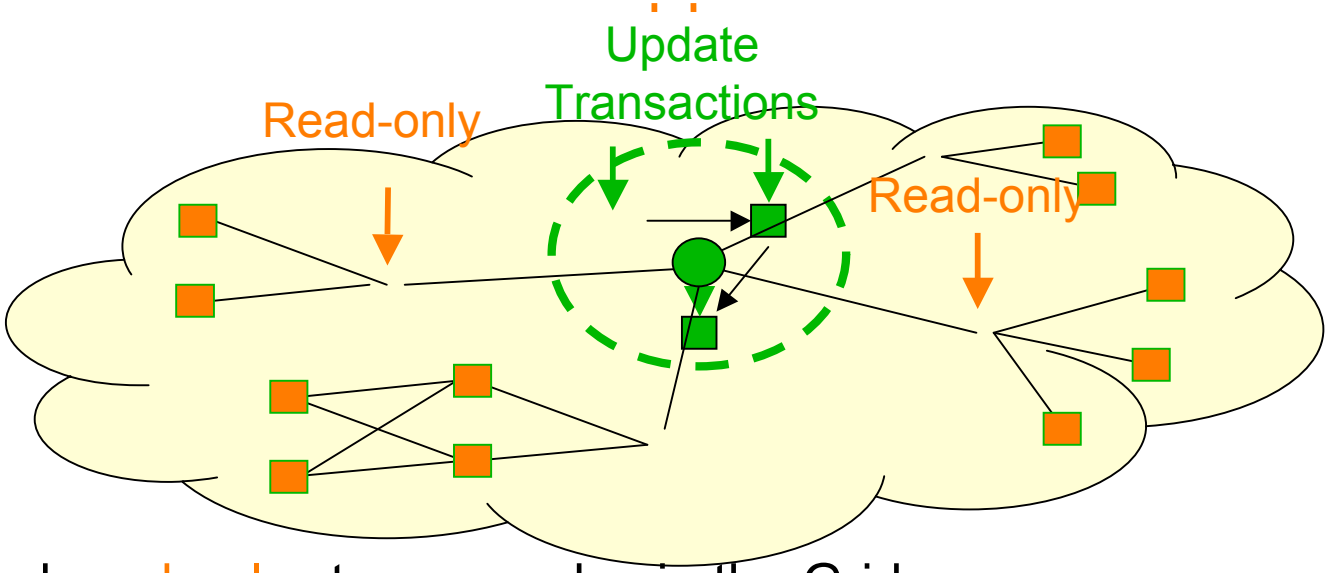
- Cluster of databases
  - Network of off-the-shelf PCs
  - Each running a commercially available RDBMS
- Middleware Architecture
  - Clients access the cluster over the middleware only
    - Two disjoint parts (update / read-only)
    - Updates occur at the update sites first, then propagated to the read-only sites
    - Replication is internally lazy, although it is eager from user's perspective
- The "Scale-out" vision
  - Adding new nodes for to increase parallelism and performance

A Sophisticated Approach to Replication: Basic Idea



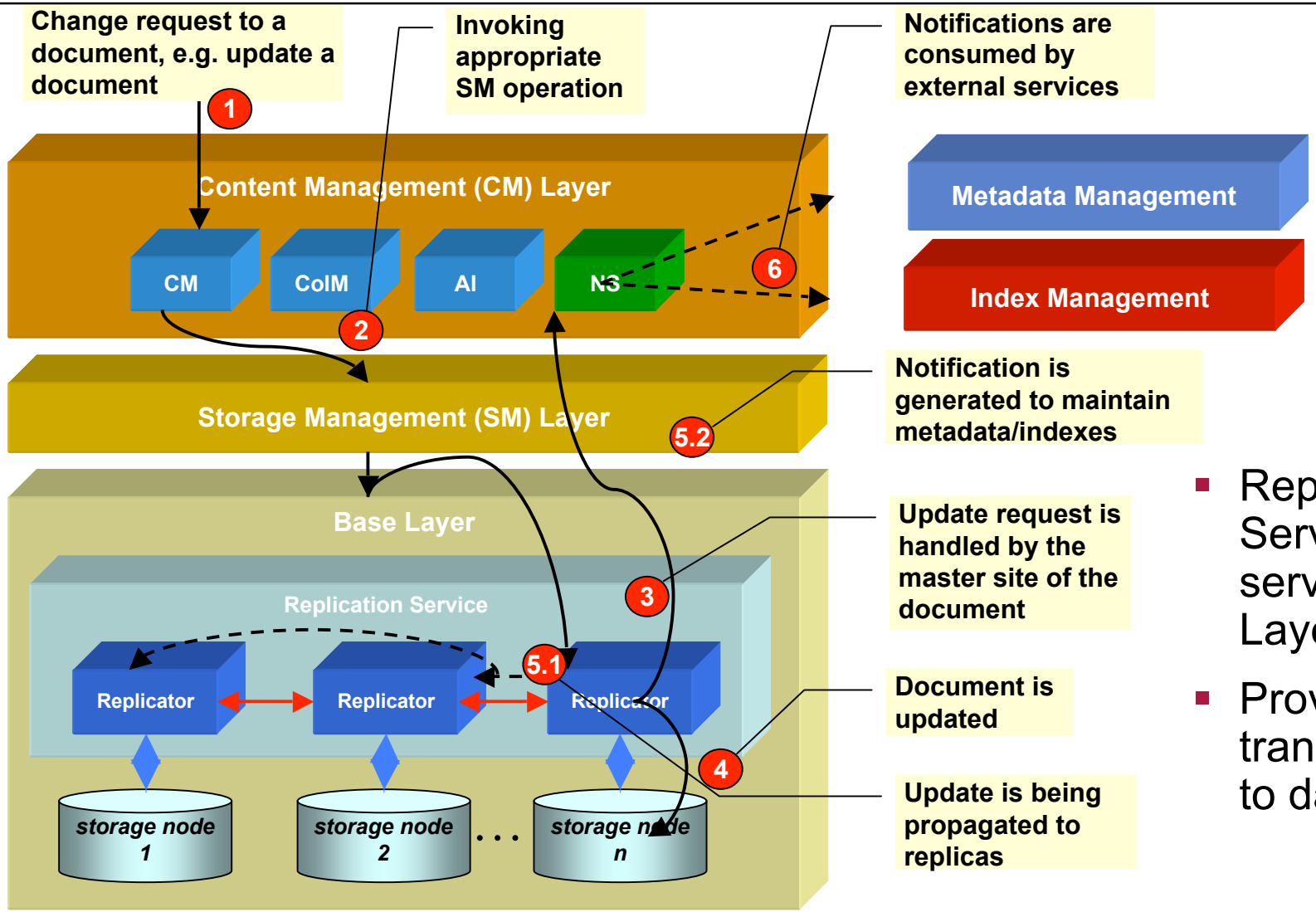
- Updates first occur at update sites, then propagated to read-only sites
- Read-only transactions may span many nodes
- Users may specify their freshness requirements
- On-demand refresh transactions bring data to the required freshness level
- Refresh/propagation is not allowed to change the version of data currently read
  - Updates occur at the update sites first, then propagated to the read-only sites

Advanced Replication in the Grid



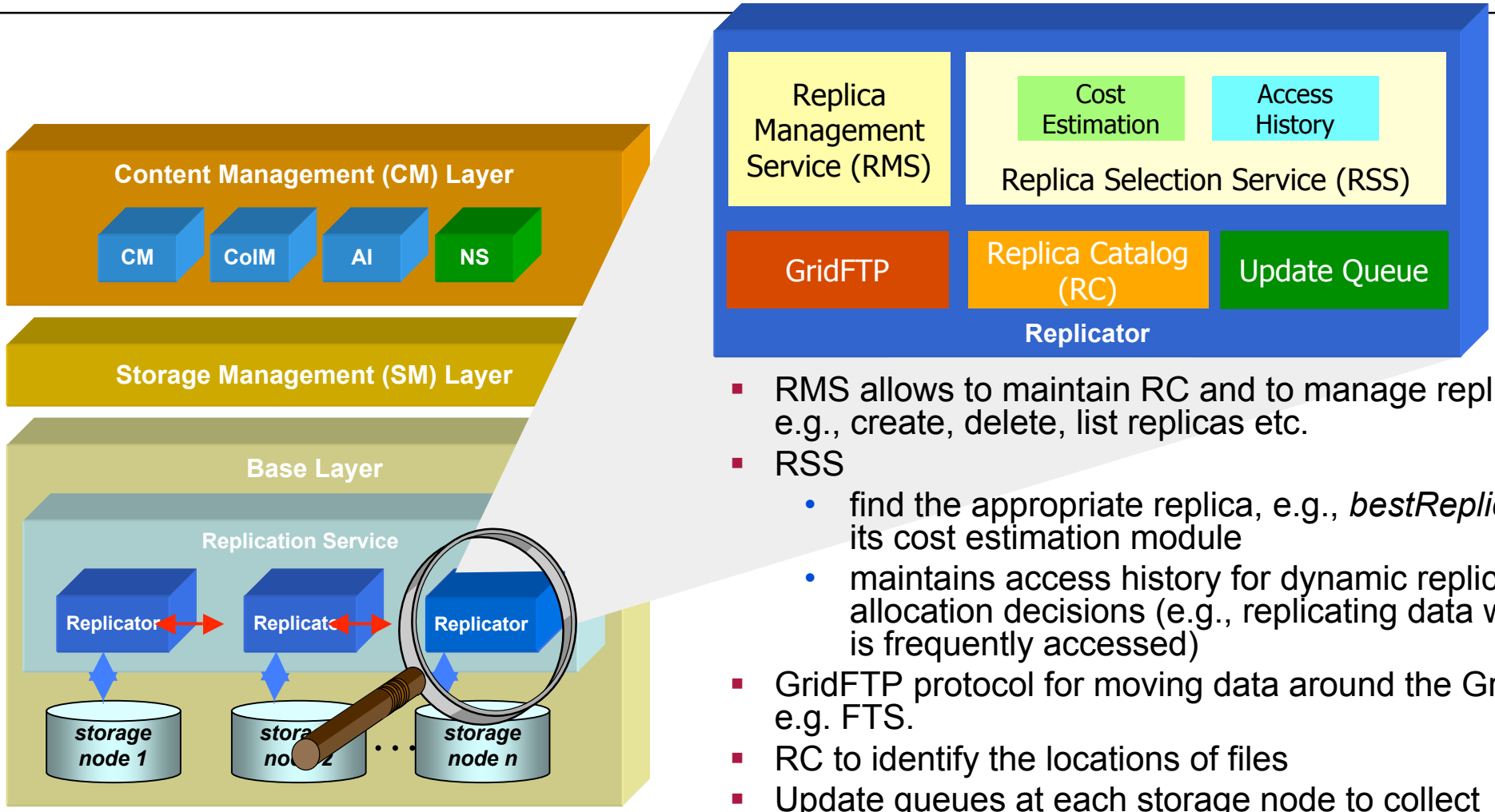
- Update and read-only storage nodes in the Grid
- A query can be served by any node while changes can only be sent to an update node
- Many update nodes per object
  - Correct serialization and propagation of updates is an issue
- Specific features for replication in the Grid
  - Replicas subscribe for changes instead
  - Large number of heterogeneous nodes T **Failures** need to be considered

Content Management: Replication in gCube



- Replication Service: internal service of the Base Layer
- Provides transparent access to data stores

… Content Management: Replication in gCube



- RMS allows to maintain RC and to manage replicas, e.g., create, delete, list replicas etc.
- RSS
  - find the appropriate replica, e.g., *bestReplica* via its cost estimation module
  - maintains access history for dynamic replica allocation decisions (e.g., replicating data which is frequently accessed)
- GridFTP protocol for moving data around the Grid, e.g. FTS.
- RC to identify the locations of files
- Update queues at each storage node to collect updates from master nodes.

Content Management: Lessons Learned

- In terms of content management, DLs are much more demanding than 'traditional' Grid applications like high energy physics or astronomy
- Basic tools and protocols are in place but they need to be extended in order to
  - Associate the different parts of an information object
  - Associate information objects and all its meta data
  - Transparently replicate information objects and their meta data

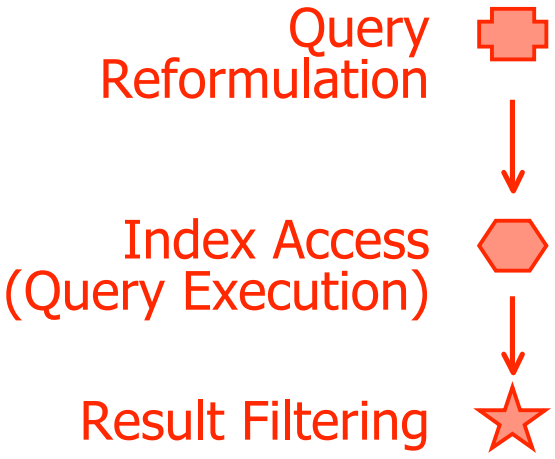
Diligent

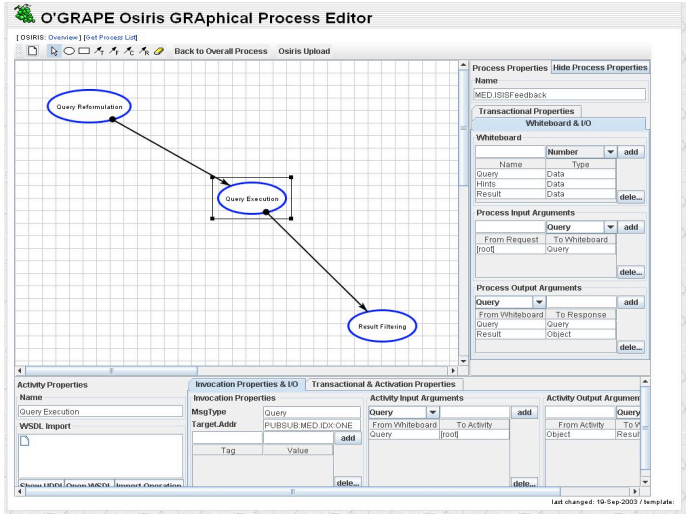
gCube: Process Management

Heiko Schuldt
UNIBAS

Short Recap: Distributed Digital Library Applications

- Grid-enabled Digital libraries are large-scale collections of digital content
  - They provide dedicated services to manage and access digital content
- Digital Library applications require to integrate these (distributed) services into a coherent whole = „Programming in the Large“ (composition of services into processes)

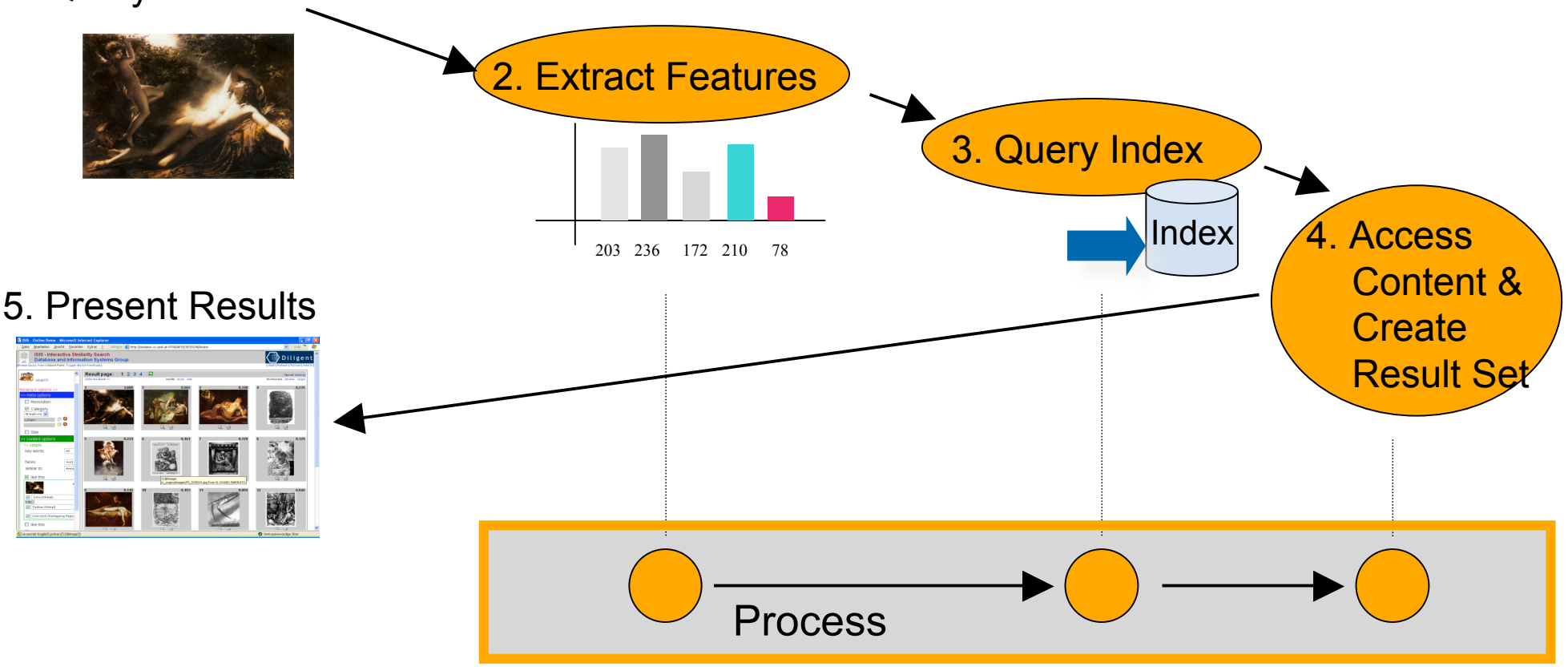




Process Management: Example Search

- Similarity search over multimedia documents

Service

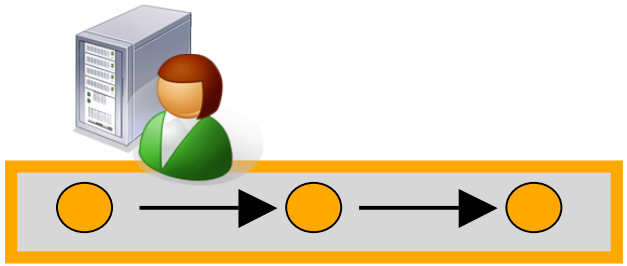


Processes in gCube

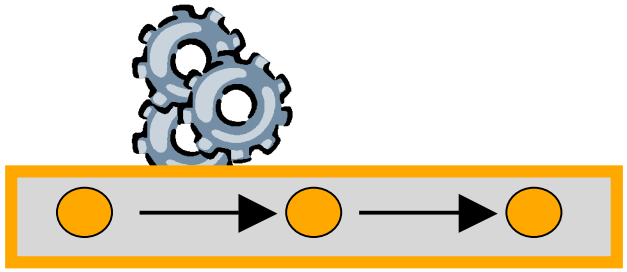
- Following the SOA paradigm, processes are the first choice in gCube to define and execute applications on the basis of available services
- gCube's approach to Process Management on the Grid consists of three main services
  - Process Design and Verification
    - Provide a graphical user interface for specification and analysis of processes
  - Process Execution and Reliability
    - Distribute process support in the Grid (avoid centralized workflow engine!)
    - Dynamic allocation of resources
    - Sophisticated failure handling
  - Process Optimization
    - Structural process modifications to maximise parallelism

Application vs. System Processes

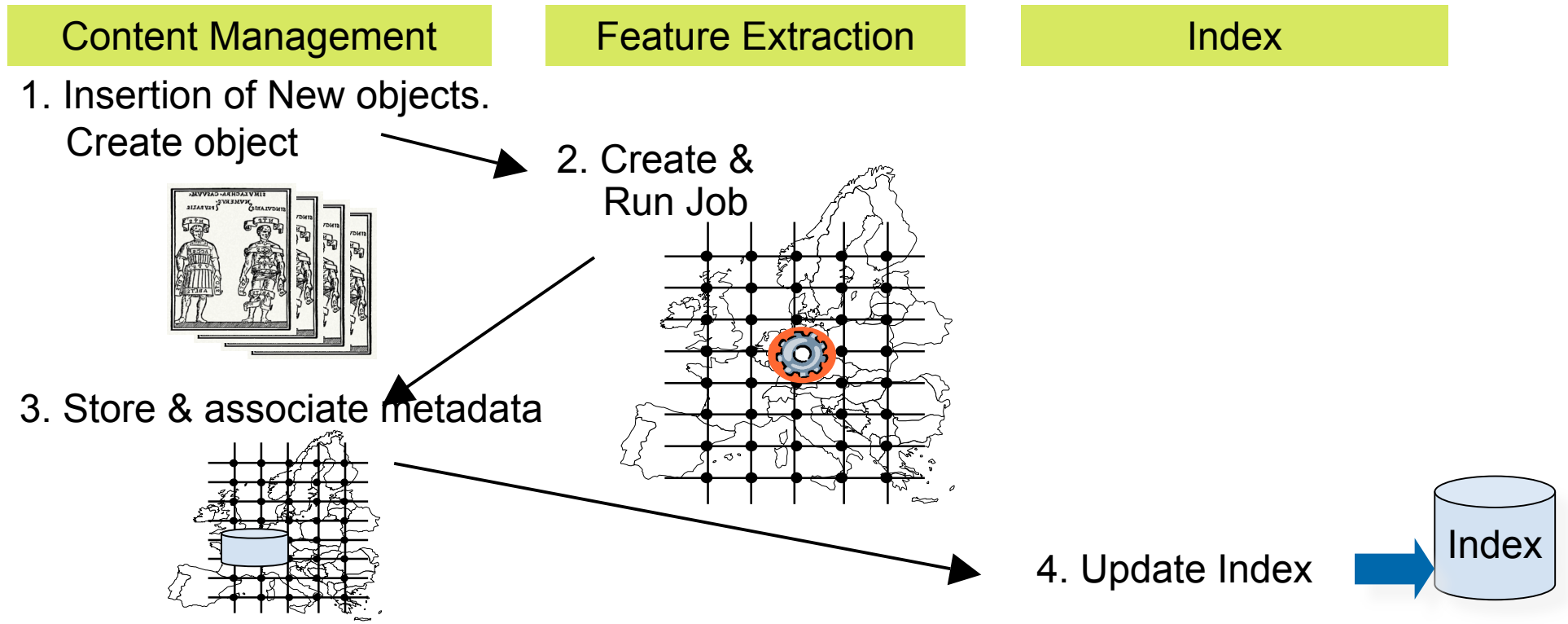
- In gCube, there are two classes of processes
- Application processes
  - Defined by the user (or at least according to the requirements of a user community)
  - Provides functionality for a particular application
  - Invoked by users



- System processes
  - Defined by the DL Administrator
  - Provides functionality needed by the gCube system
  - Automatically invoked by other gCube services
    - E.g., for administering a DL in flexible way
    - Usually invisible to the end users



Sample System Process: Make new content available

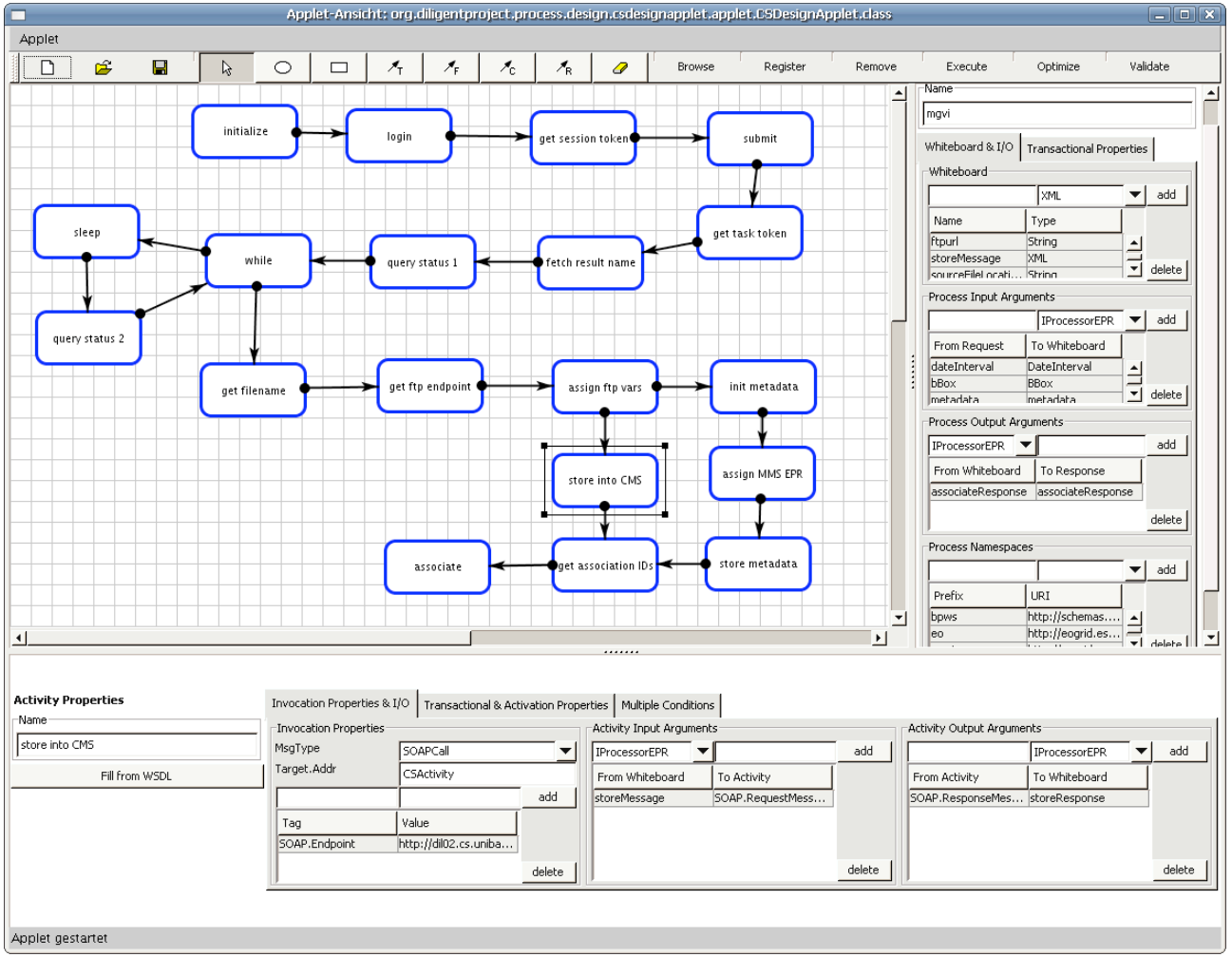


- Advantages of using System Processes and the Process Execution Service
  - Has built-in failure handling
  - Process can easily be extended/adapted without need to alter underlying services

Sample Application Process: Meris Global Vegetation Index (MGVI) …

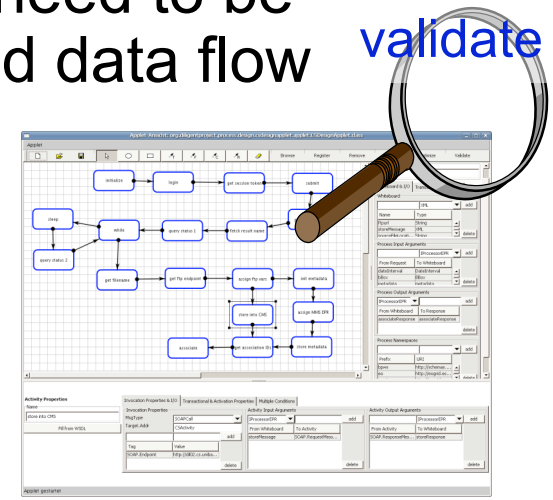
- MGVI provides information about vegetation at specified geographical region during specified timeframe
- Basis: Grid job provided by the European Space Agency (ESA); accessible via Web Service interface
- Seamless integration of (external) ESA web service and DILIGENT services,
  all invoked in a single process
- Process Steps
  - Parameters: Bounding box, date-time interval, (auto-generated) metadata
  - Log in to EOGrid
  - Submit parameterized job
  - Periodically poll until job finished (can take up to 15 mins)
  - Job stores output on intermediate FTP server
  - Tell Content Management to fetch data
  - Associate metadata with content and store in Metadata Repository

… Sample Application Process: Meris Global Vegetation Index (MGVI)



Process Design and Verification

- Unlike conventional programming from scratch, the building blocks of processes are already in place and need to be combined appropriately by defining control and data flow dependencies
  - Graphical tools for process design and specification to be provided
  - Failure situations to be anticipated, appropriate failure handling strategies have to be added to the process specification
  - Verification algorithms to be integrated into graphical process design tools
    - formally prove correctness already at build-time

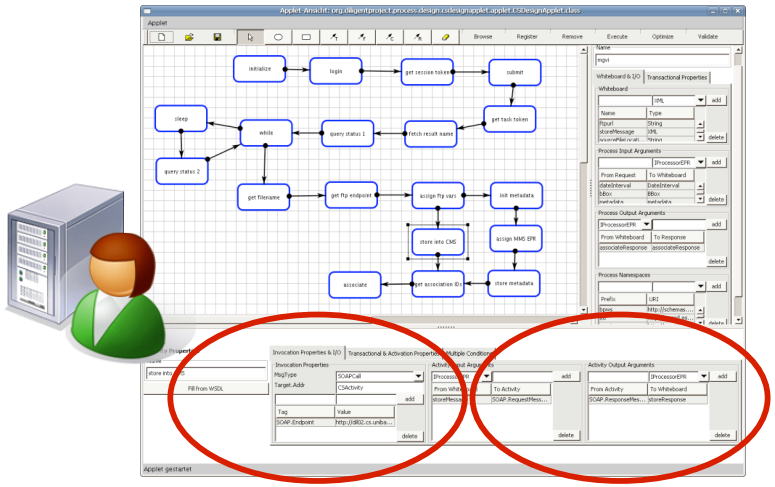


Process Definition in gCube

- gCube supports two different approaches to process design

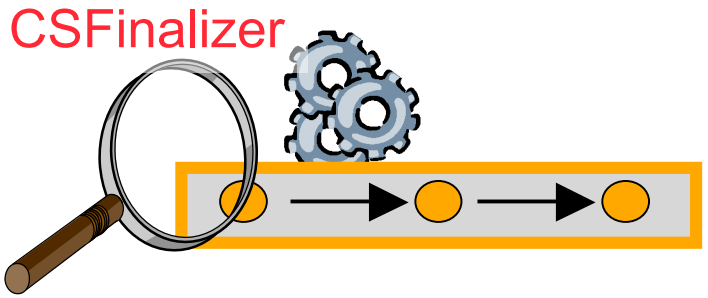
New process manually created via **`CSDesign`** applet

- List of available services retrieved automatically from the DIS
- Service parameters available for data flow specification



Process is automatically created by another gCube service (e.g., Search)

- Validated by **`CSFinalizer`**

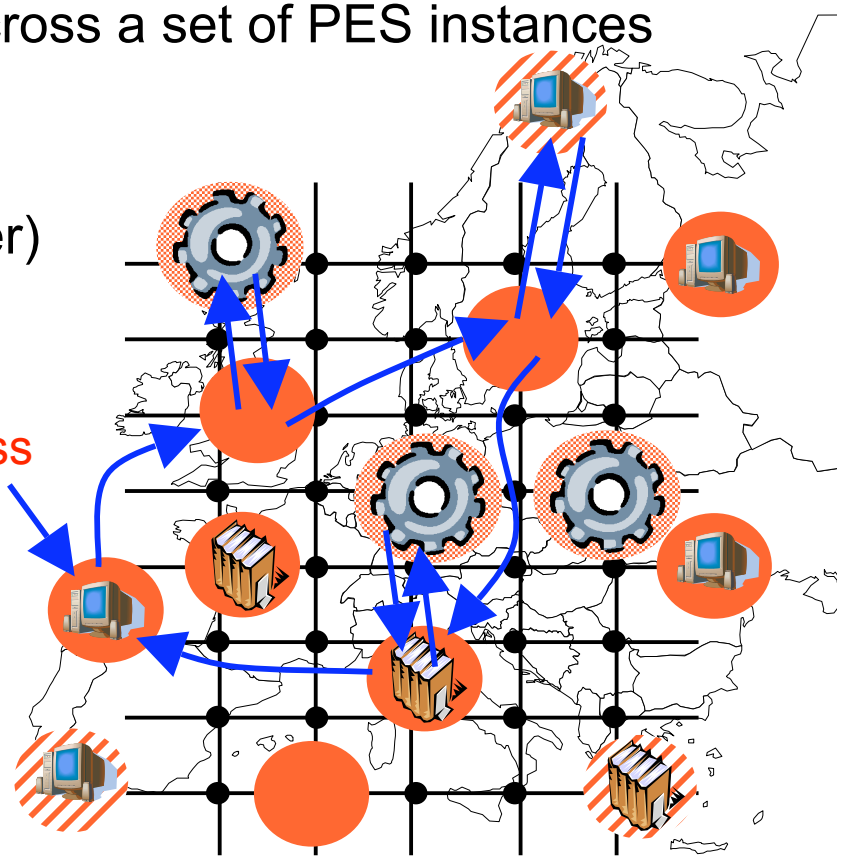


Process Execution and Reliability

- A major requirement for DL applications and the infrastructure running (process-based) applications is that correct execution –according to the process specification– can be enforced at run-time
- Reliability means that process execution should not be affected even in case of failures or changes in the environment
- Rather, sophisticated routing of service requests within a service grid is required in order to equally balance the load within the services of the grid and to ensure high response times for processes

Process Execution in gCube

- Process execution is realized in a distributed way by dedicated gCube services: PES (process execution service)
- Process execution is shared dynamically across a set of PES instances
- Services that can be invoked
  - DILIGENT services
  - gLite jobs (via generic gLite Job wrapper)
  - External Web services



DILIGENT service (with PES locally deployed)

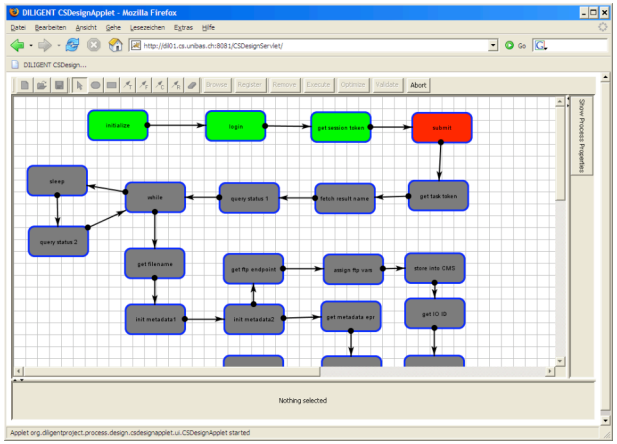
DILIGENT service (without local PES)

gLite job

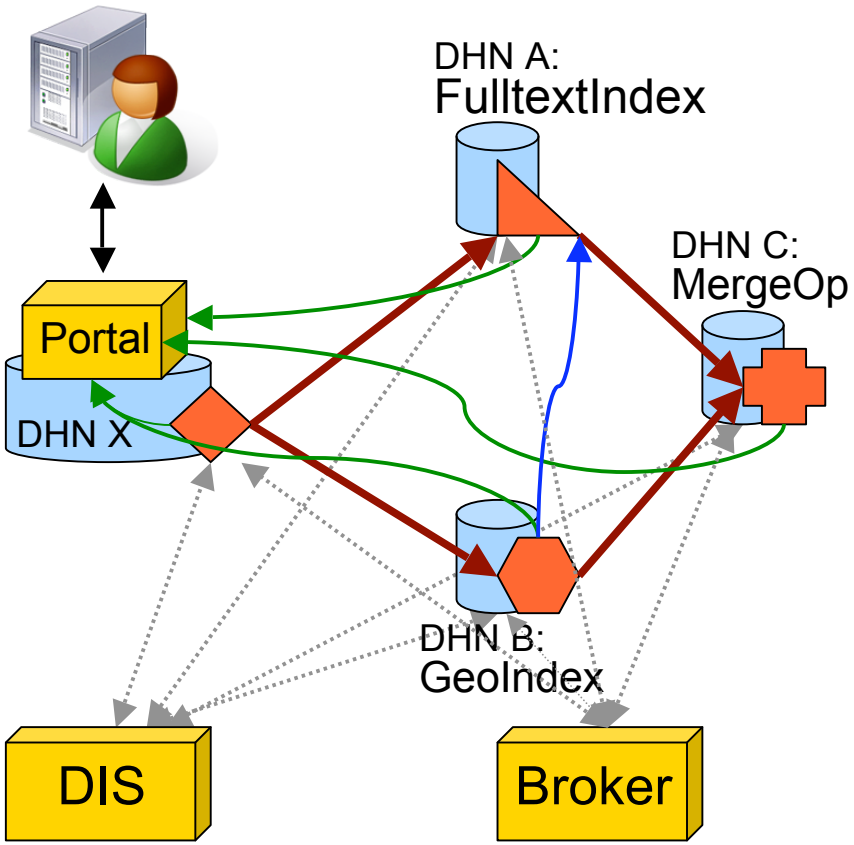
gLite job wrapper

Process Monitoring

- Distributed and reliable Process Management is one of the main features of the gCube system
  - But it is basically system functionality that cannot be seen directly
  - A monitoring interface has been added that allows the DL Administrator to keep track of running process instances
  - Based on the Process Design applet, state (= active services) is graphically highlighted
  - Uses notifications to get information on state changes (monitoring completely distributed processes is far from being a trivial task)



Process Execution: Overview



- `CSEngine` runs on every DHN
- Process orchestration distributed in the infrastructure
- Service deployment information from DIS
- Process definitions possibly from DIS
- Notifications (all brokered):
  - Process status information (monitoring)
  - Join coordination

Process flow

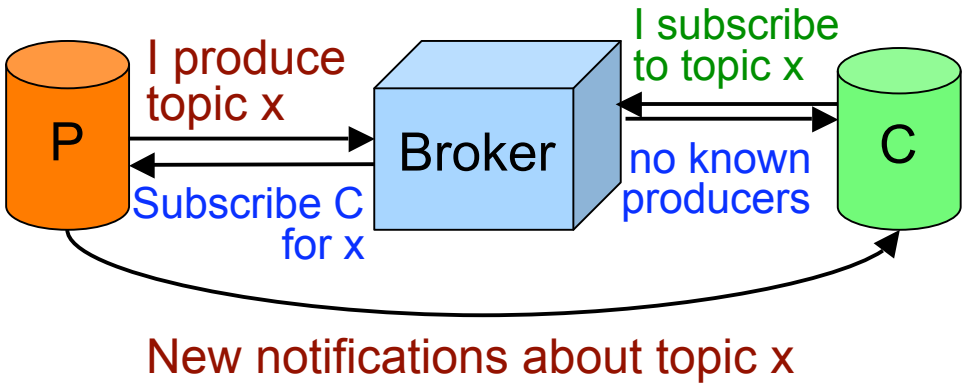
Join notifications (crucial for execution)

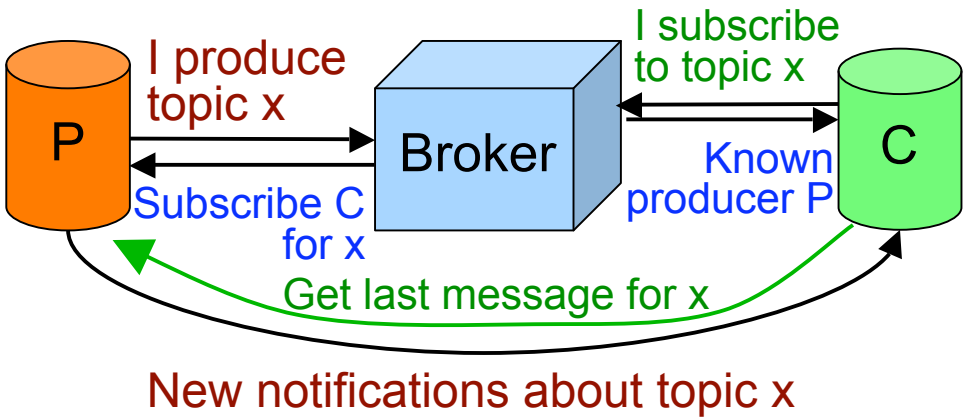
status notifications („optional“)

Collective Layer interactions

Brokered notifications

- Brokered notifications – extending WS-BaseNotifications; rationale:
  - Subscribers do not know where topic is produced
  - Producers do not know where subscribers are
  - E.g. join notifications: location of all execution nodes is dynamic in the system, have to agree on common join node
  - In addition, topic might be „made available“ before any subscribers present – need to keep the info available
  - NB: the actual notifications are „p2p“, only the subscription is brokered

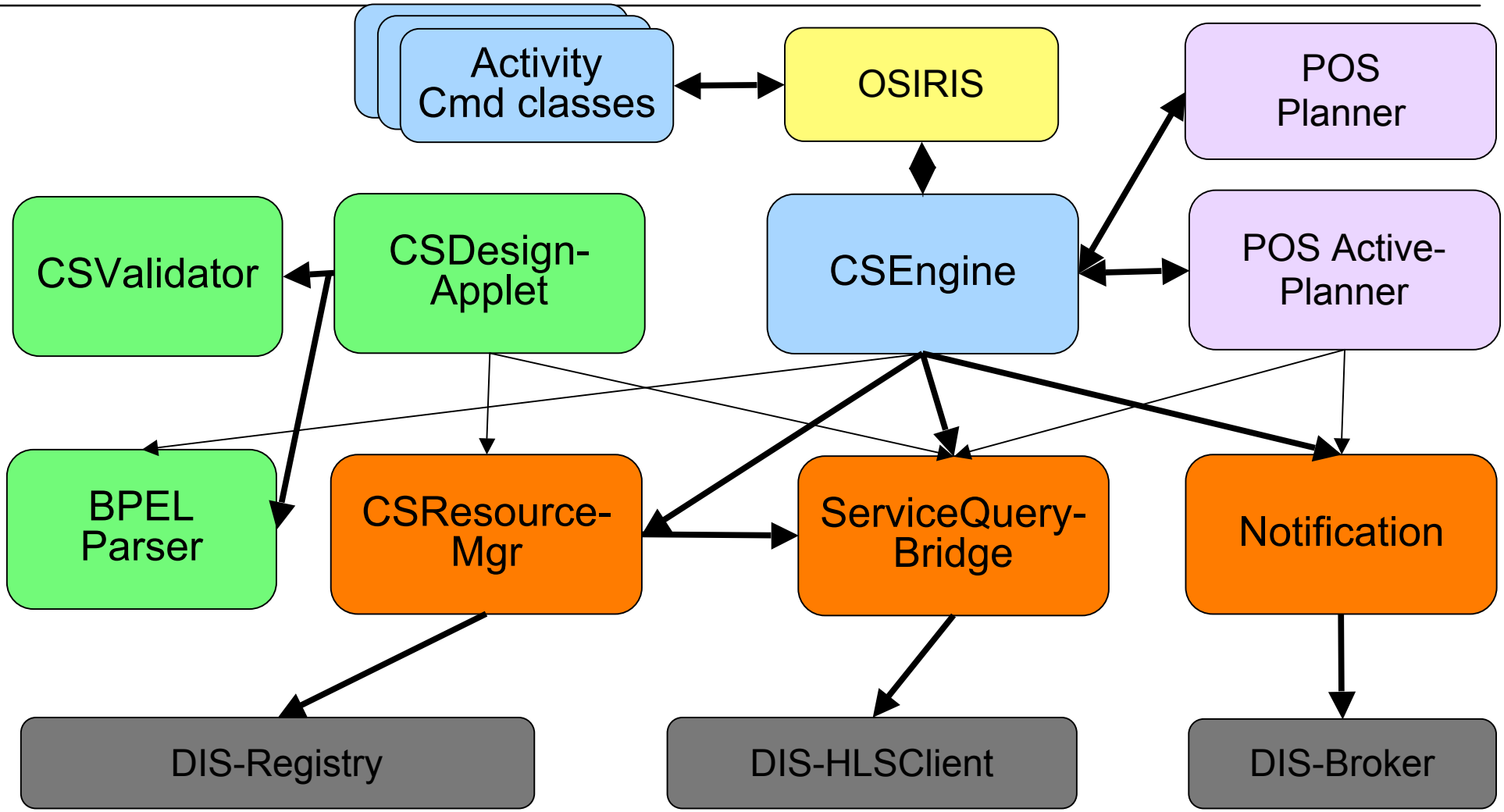




Process Optimization

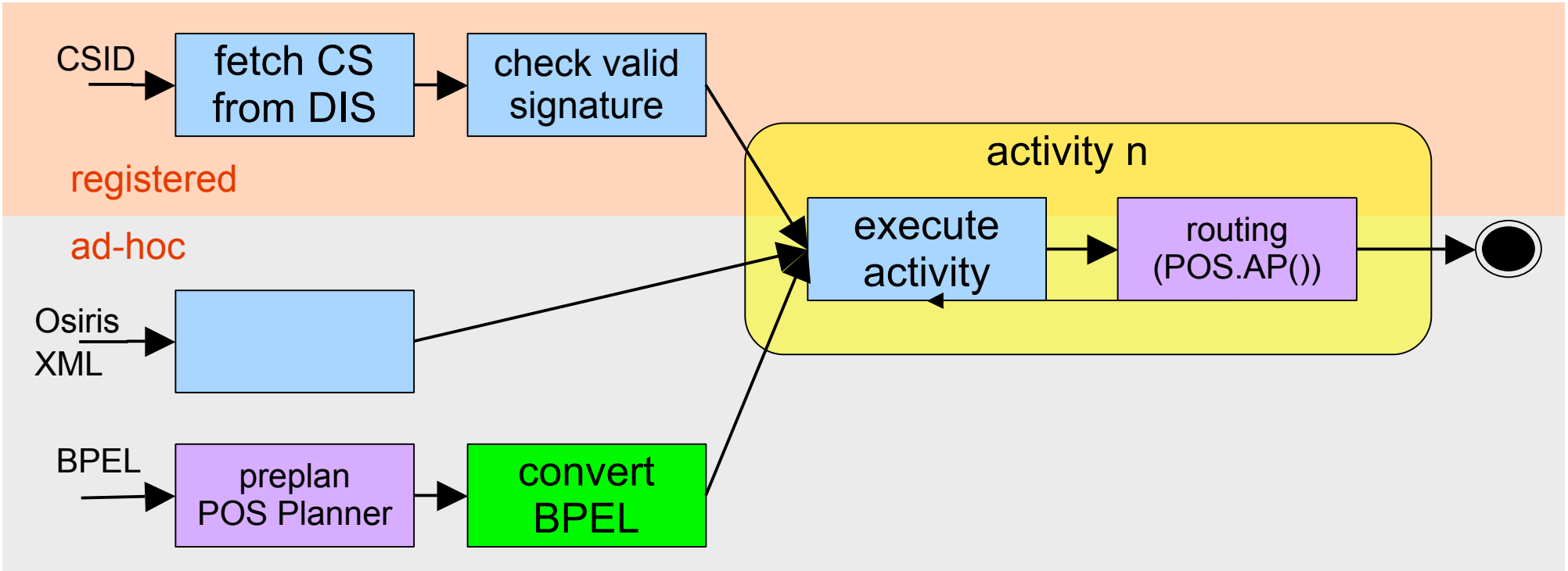
- User queries are being submitted to the Query Process Optimization Service
- Processes allowing to search within the content of the DL have to return results to their users as soon as possible
- This service consolidates information provided by services such as resource description, indexing and personalization and produces a “query” execution plan which is actually a verified, optimized process (workflow) that will ultimately deliver the desired results.
- Process optimization is an important task that may change the description of a process while, at the same time, the provable correctness of these processes must be preserved

Overview of Process Components



Process Execution: Interaction of Services

- gCube distinguishes two „types“ of processes: registered and ad-hoc
  - Registered: defined e.g. via portlet, stored in DIS
  - Ad-hoc: pass process definition on invocation (e.g. search)



Lessons Learned

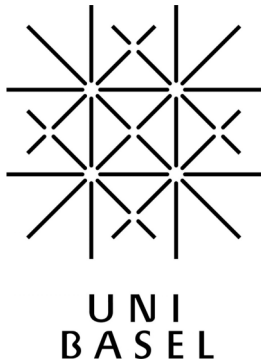
- Processes facilitate application development in a SOA environment
- Traditional process management considers a centralized process engine (or a centralized instance database)
  - this does not scale to Grid-size
  - gCube provides a fully distributed approach to process management that
    - avoids any single point of failure
    - allows to dynamically adjust to changing environments (e.g., when new services are available)

Building Digital Libraries on Grid-enabled technologies: challenges, experiences, and results

gCube: Distributed Information Retrieval in SOA

ISTITUTO DI SCIENZA E TECNOLOGIE DELL'INFORMAZIONE "A. FAEDO"

Tutorial at ECDL 2007
Sep, 16th 2007



George Kakaletris (University Of Athens)

gCube: Distributed Information Retrieval in SOA

George Kakaletris
University of Athens

Overview

- From Objectives to Concepts
- Functional Overview
- Design and operation
- Distributed Information Retrieval Elements
- Optimization
- Interoperability of Information

Objectives

- An open, feature-rich inherently distributed Search Engine
  - Composed out of diverse, autonomous, pluggable elements.
  - Capturing complex application scenarios
    - Combining information retrieval and data processing procedures
- Maximization of resources placed at the disposal of Digital Library managers and users
  - Ease of sharing of resources, avoiding mis-utilization and misuse
  - Reduction of cost of ownership and use
  - Creation of a “field” for further experimentation with IR technologies

Grid & SOA environment challenges

- Service Oriented Architecture
  - Operate on the OGSA environment
- Operate in uncontrolled dynamic environment
  - Access control
  - Uncontrolled Joins/Parts
- Consume diverse, physical and virtual, standards-compliant resources
- Provide standards-compliant resources for building higher level services
- Provide means for extending and customizing base implementation (interfaces, semantics, …)
- Overcome performance issues of Service Composition

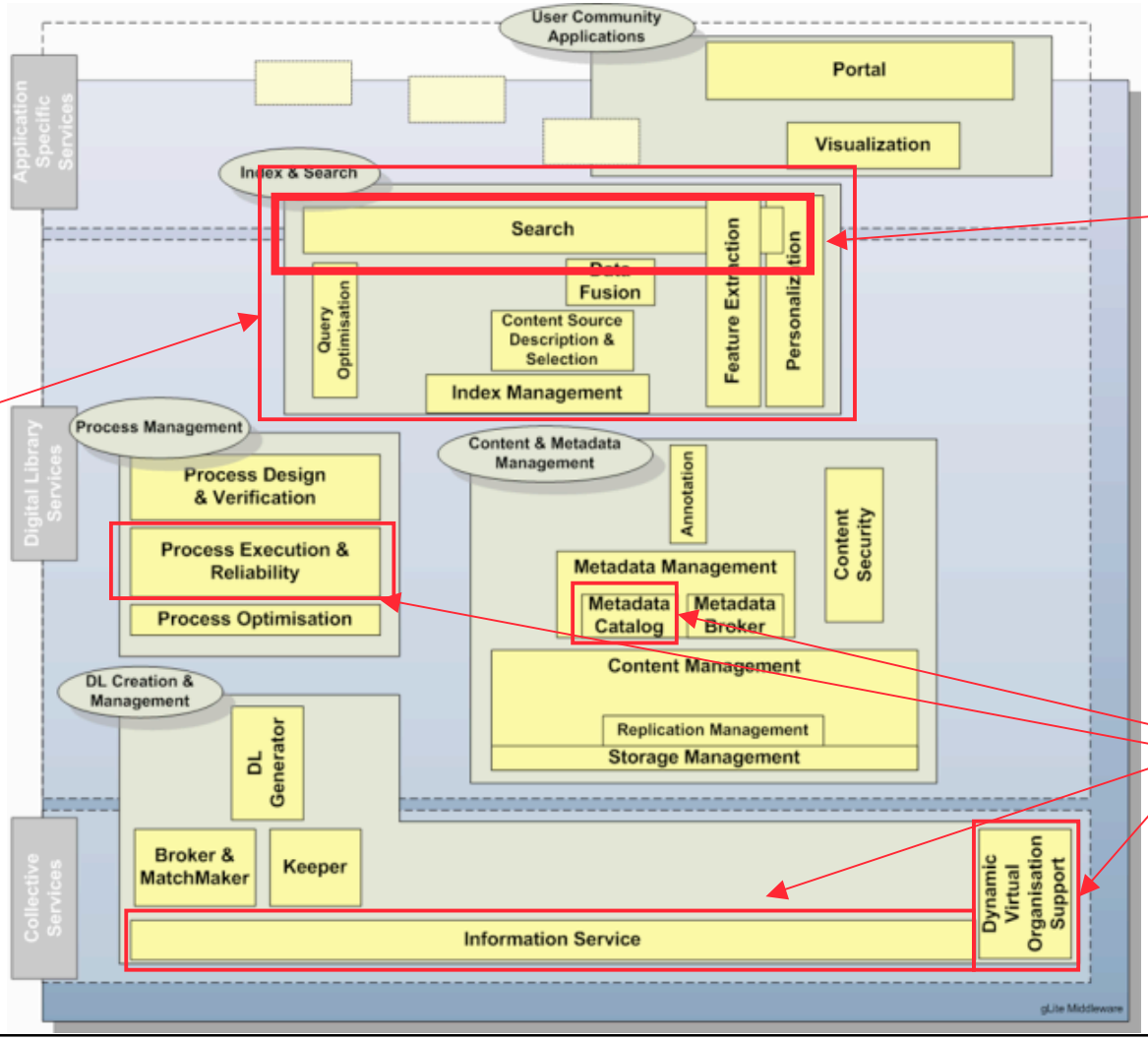
The Concepts

- A **Search Service** orchestrator of Information Retrieval
  - Feature-full, open, inherently distributed
  - Consolidates query and environment information
  - Prepares and plans retrieval execution
- A **Query Language**
  - Proprietary - Exposes the openness of the system
- Numerous **"worker" services** provide IR & data processing means
  - XML processing (joiners, sorters, transformers, filterers)
  - Lookups (FT indices, XML indices, Geo indices, external sources etc)
  - Combining of results (Fusion / Merging)
- A **data transport mechanism** (ResultSet)
  - Overcome environment restrictions towards performance
  - Standardize information exchange among IR workers

Digital Libraries on gCube middleware: Functional overview

- Supported search types
  - Structured data (fielded search / xml search)
  - Semi structured data (xml search)
  - Geospatial / temporal data (R-Tree)
  - Content based search
    - Full text search
    - Image similarity search
- Access means
  - XML-based Query Language
  - Web user interface (portal / search portlets)
  - Command line UI
- Operation highlights
  - Incremental result delivery
  - Planning & Optimisation
  - Distributed Information Retrieval features

Search Components in gCube Architecture

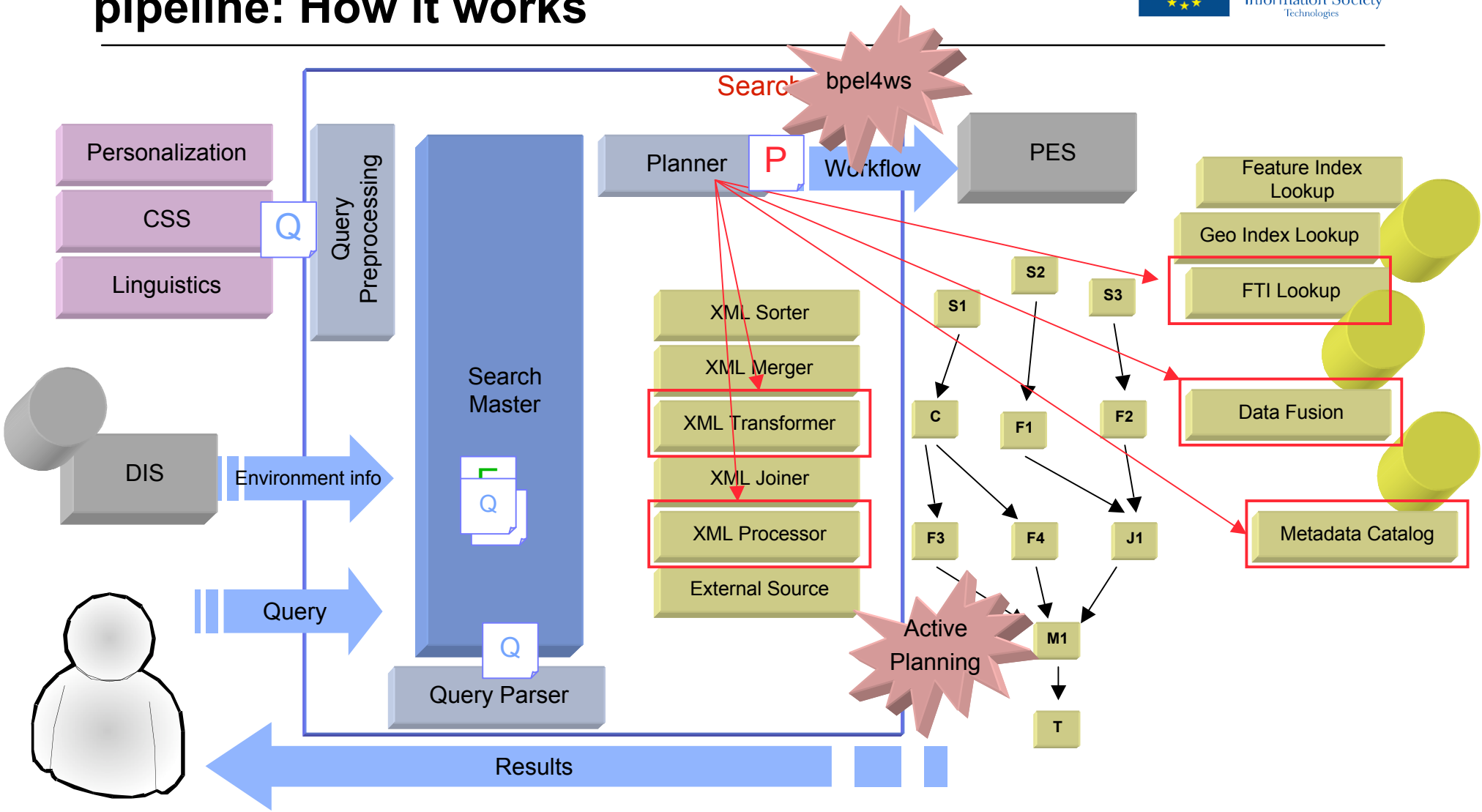


Logically co-grouped with, and directly depends on…

Search Service

Direct dependencies

Service Oriented Information Retrieval pipeline: How it works



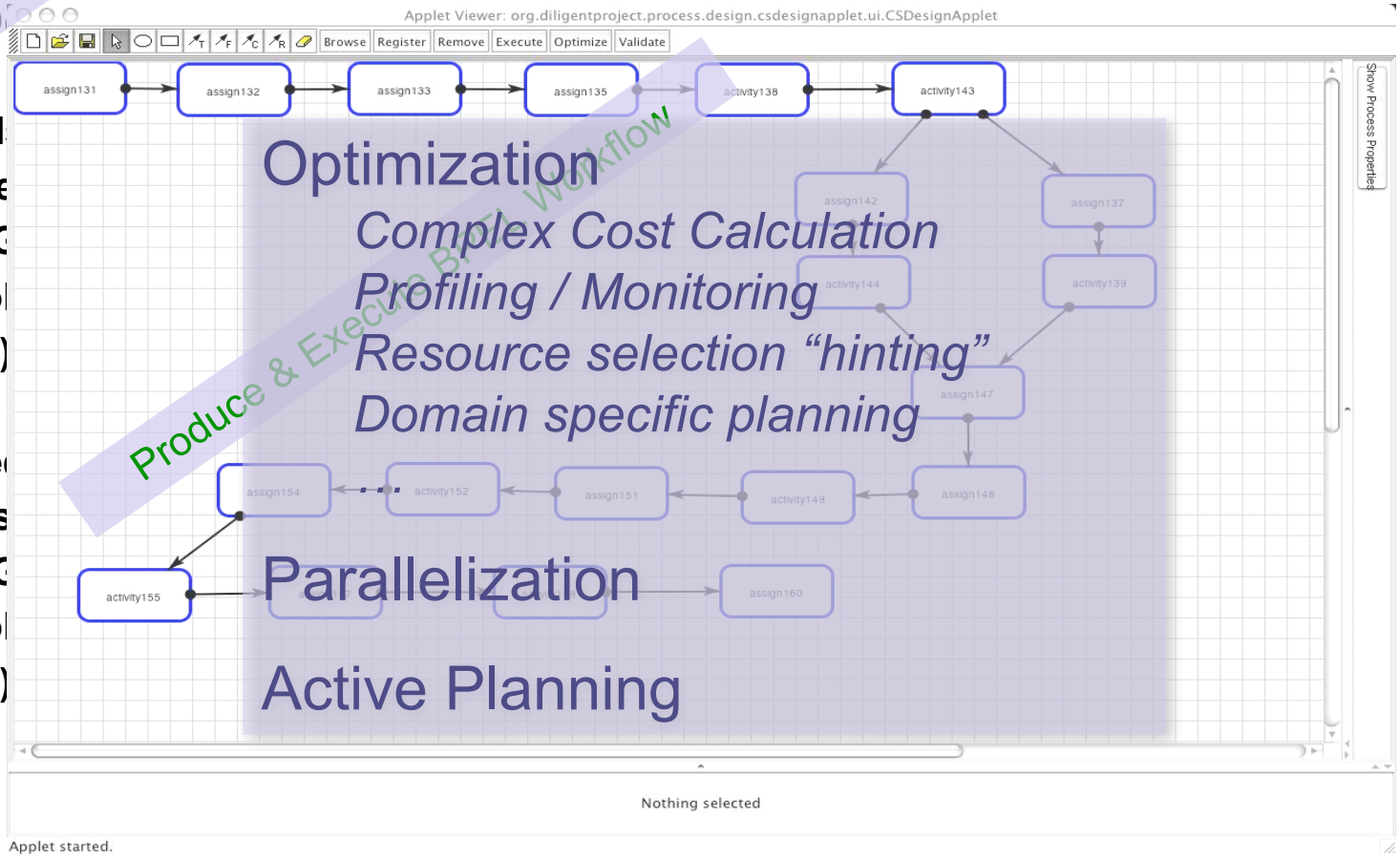
Behind the scenes: Always executing service workflows

```
project by 'title', 'description', 'subject'
    on (keeptop 20
        on (sort ASC by 'D
            on (merge
                on (fielded
                    by 'title
                    in 'ENG
                    on 'Col
                    as 'dc')

                and (fielde
                    by 'des
                    in 'ENG
                    on 'Col
                    as 'dc')
            )
        )
    )
```

Query

Produce & Execute BPEL Workflow



Behind the scenes: It can get complex…

```
project by 'title', 'date' on
   (sort ASC by 'DocID' on
      (merge on
         //MAP REPORTS
         keeptop 8 on
            (sort ASC by 'RankID' on
               (join inner by 'DocID' on
                  (fulltextsearch by 'Mediterranean' in 'ENGLISH' on 'd369b3e0-fa4c-11db-a297-9c01d805f283')
               and
                  (fulltextsearch by 'Environmental' in 'ENGLISH' on 'd369b3e0-fa4c-11db-a297-9c01d805f283')))

         keeptop 8 on (sort ASC by 'RankID' on (join inner by 'DocID' on (fulltextsearch by 'Mediterranean' in 'ENGLISH'
on 'd369b3e0-fa4c-11db-a297-9c01d805f283') and (fulltextsearch by 'Environmental' in 'ENGLISH' on 'd369b3e0-fa4c-
11db-a297-9c01d805f283')))
         // EEA reports
         keeptop 8 on
            (sort ASC by 'RankID' on
               (fieldedsearch by 'date' contains '*1999*' on
                  (join inner by 'DocID' on
                     (fulltextsearch by 'air polution' in 'ENGLISH' on '25ad3c50-fa41-11db-a270-9c01d805f283')
                  and
                     (fulltextsearch by 'european' in 'ENGLISH' on '25ad3c50-fa41-11db-a270-9c01d805f283')
                  )
               )
            )
      )
   )
```

Major Information Retrieval Worker Elements

Indices

- Serve queries via proprietary languages
- Can manage linguistic issues
- Types
  - Forward :
    - B-Trees (field lookups).
    - R-Trees (geo-spatial search).
    - High-dimensional VA files (content based search).
  - Inverted:
    - Full Text Indices

Search Operators

Metadata Management Service

- (Provides Metadata)
- Serves direct XML queries on Metadata

Distributed Information Retrieval Components

- Describe and select Content Sources
- Fuse results stemming from different sources

Search Operators: A Closer Look

- Sorting / Filtering results
  - Index lookups
  - Metadata processing (xpath/xquery)
  - Row-by-row evaluator
- Joining results
  - Joining metadata from various sources.
  - Inner / Outer joins
- Merging results
  - Plain unions.
  - Data Fusion to merge ranked lists.
- Querying External Sources
  - Google, JDBC data sources, ISIS/OSIRIS system etc.
- Aggregation functions
  - Calculate values over entire ResultSets.
  - Necessary for conditional execution.
- Transformations
  - Powered by XSL/XSLT.
- Invocation of custom processors.

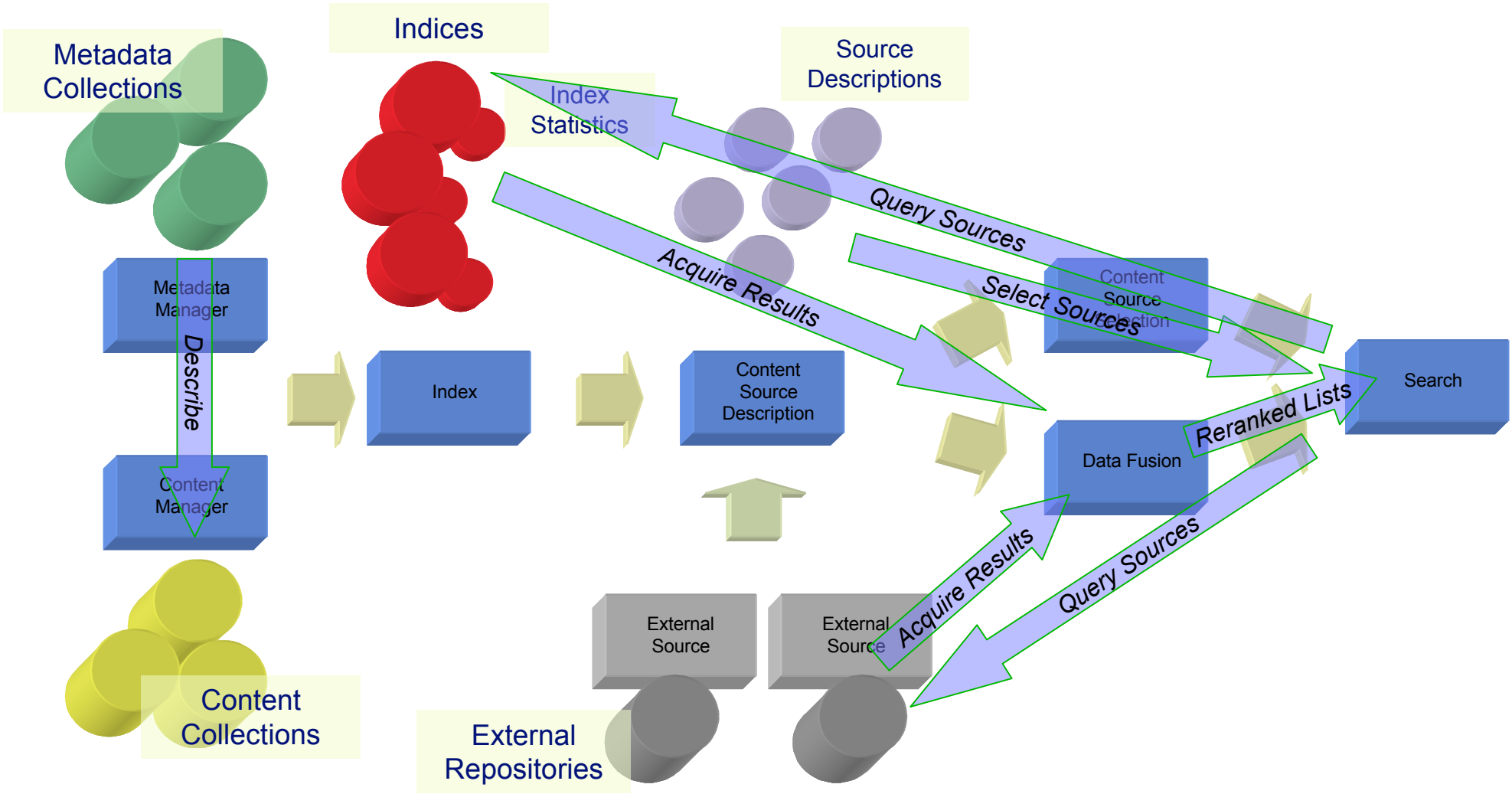
Search operators are:

- typical Web Services or Web Service Resources
- described n DIS via appropriate Profile
- forced to comply with the ResultSet data exchange mechanism Search-specific usage

Acquiring information from different content sources

- The case
  - Several content sources
    - Potentially thematically focused
    - Cost of querying
  - Different ranking estimation
    - Different structures for managing metadata / information retrieval
    - Different content
- The challenge
  - Selecting the appropriate sources
  - Merging the results in meaningful lists
- Fusing
  - Re-ranking
  - Valid only on ranked result lists

Selecting Sources and Fusing Results: "Under the hood"



Opportunities for optimization

- Pre-query optimization:
  - Keeper service monitors and adapts the DL layout for optimal resource usage.
- Content Source Selection:
  - Filters out collections unlikely to be contain what the user is looking
  - Uses query supplied terms and automatically pre-constructed Content Source Descriptors
- Query Planning:
  - Cost based optimization performed.
  - Heuristics and space-search
- Process Execution:
  - Process optimization service selects and allocates the appropriate resource to carry out a task.
- On-The-Spot processing:
  - ResultSet mechanism to allow local filtering of large XML chunks of data.
- Further mechanisms to facilitate efficient searches:
  - Indices.
  - ResultSet transport mechanism to bypass WS-* shortcomings and facilitate paged data exchanges.

Interoperability of Information

- Enabled by the infrastructure
  - Multiple schema hosting Metadata Management Service
  - Powerful schema transforming engine: The Metadata Broker
- Supported by Search
  - Selection of common schemas for cross-collection semistructured searches
  - Support for on-the-fly projection
- Assisted by the user interface
  - Composing queries to exploit search capabilities
  - Exploits administrator's or end-user's knowledge on hosted information structure

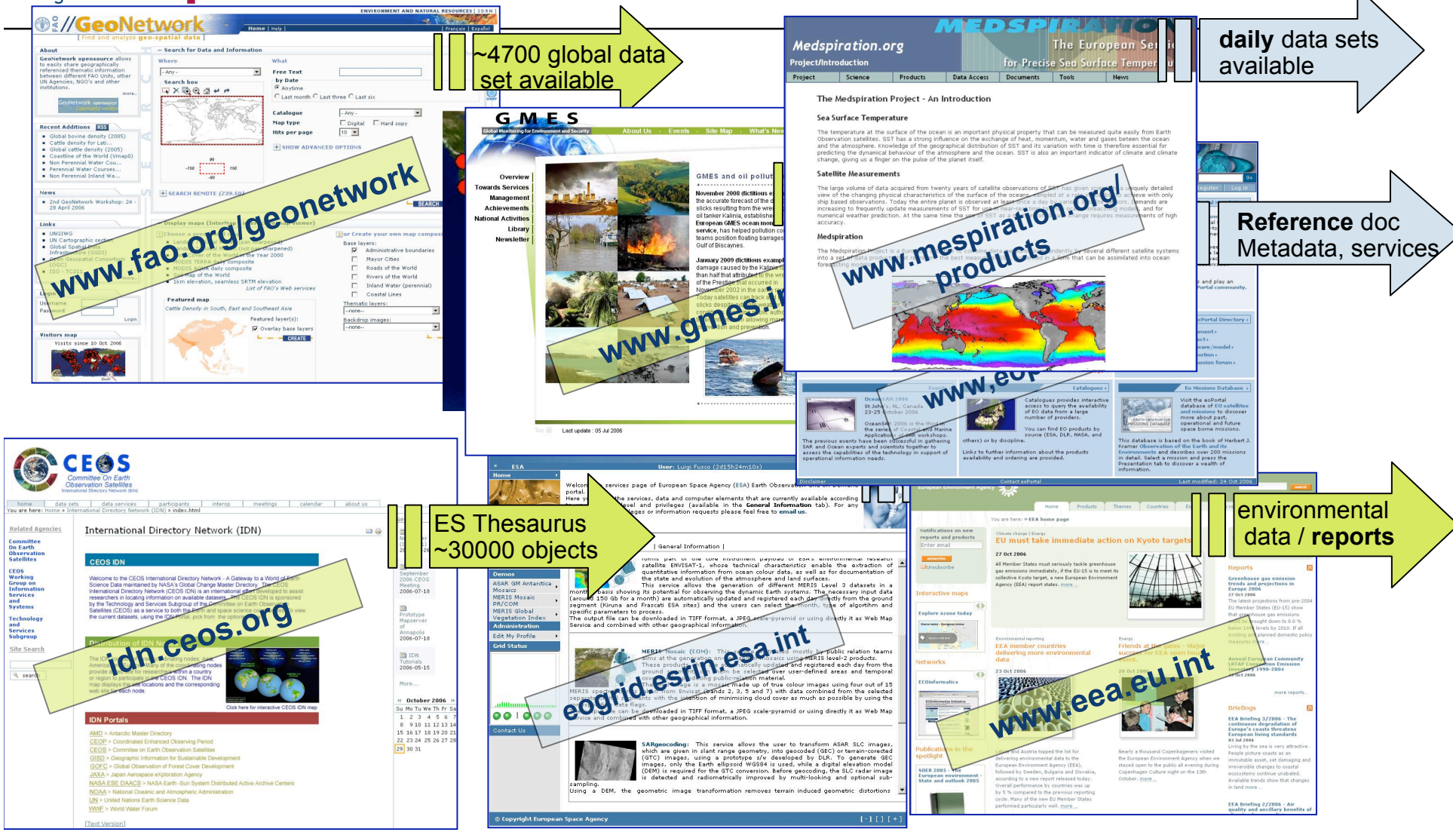
Questions

Applications

ImpECt use case

Build an ad-hoc user-defined virtual research environment where content, applications and services collaborate together to generate on-demand new valuable information, e.g. complex environmental reports

ImpECt Resources from the Web…



ImpECt Resources – Services

- ✓ High level geophysical products generation (e.g. mosaics, chlorophyll distribution profiles, vegetation index, atmospheric profiles)
- ✓ Environmental reports generation
- x Data analysis and visualisation services

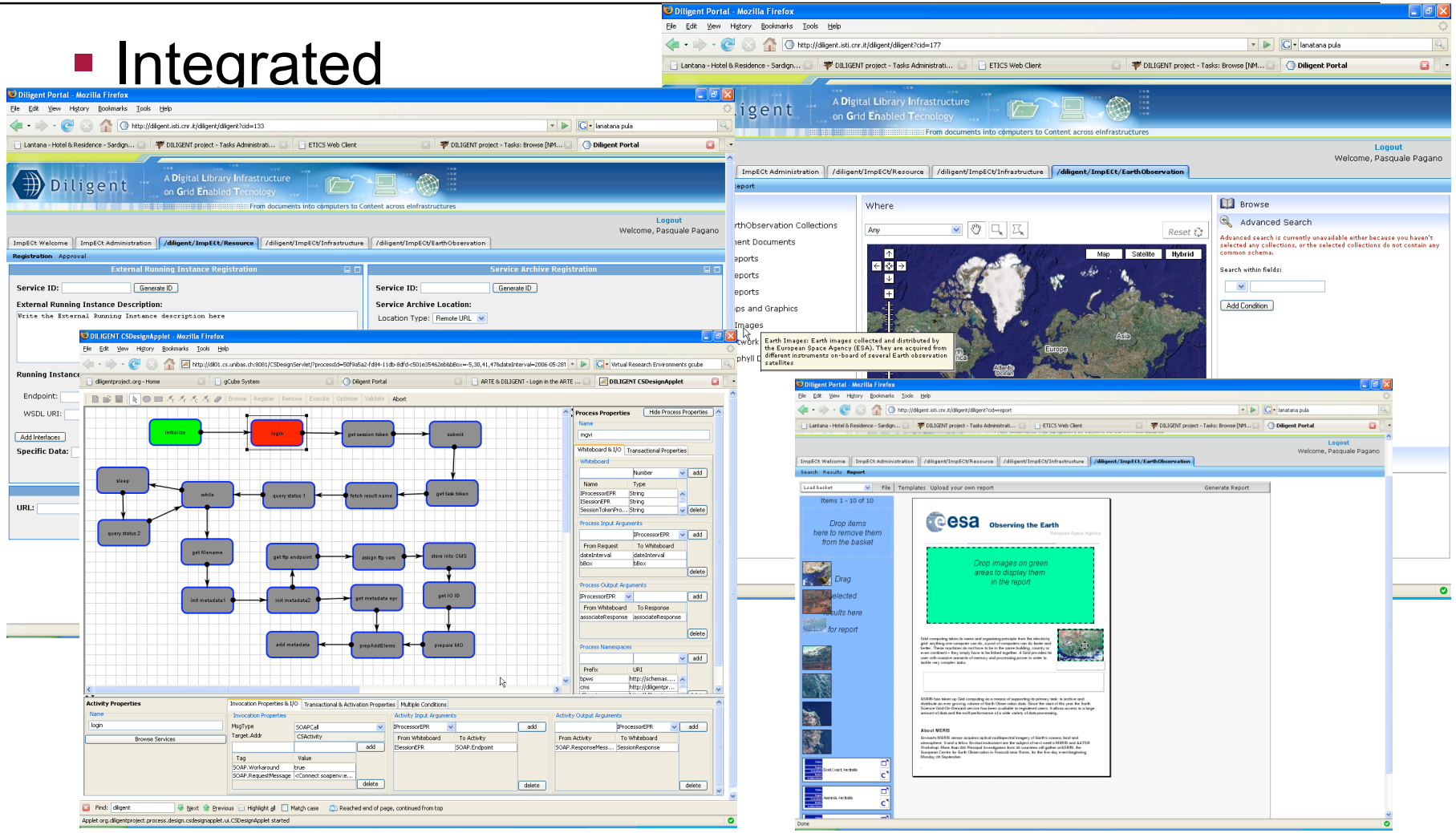
from

- ✓ ESA Grid-on-demand (eogrid.esrin.esa.int)
- ✓ Diligent ad hoc developed
- x EO web portal (www.eoportal.org)
- x GMES services (www.gmes.info)

x will be integrated soon

Diligent for ImpECt

- Integrated



Information Society Technologies

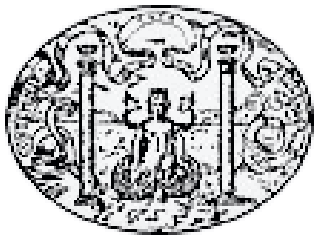
ARTE use case

ARTE (Applicazione di Ricerche e Tecnologie di Editoria digitale) is a community of researchers coming from scholars located all over the world, that work together to set up the basis for a new research discipline that merges experiences from the humanity, social science, and communication research areas.

- SNS (Scuola Normale Superiore): Studies the complex relationship between words and images in the literary tradition with computerized tools
- RAI: Italian's public broadcaster who aims to disseminate history, science, art and culture through audiovisuals

What is it about ?

- Sharing
  - “machinery”
  - information and knowledge
- Reduce cost of ownership and use
- Open new opportunities for processing content
- Bring together multidisciplinary domains:
  - To user service
  - To research scope
- Service domain aware users
- *Inter-operability of information*
- A test-bed for proving and discarding concepts



ARTE Archives Past Obstacles Volumetrics & Specialties

- Heterogeneous collections of high diversity
  - Several data formats
  - Embedded data issues
  - Different availability
- Unconventional collection constituency
  - "Small" number of items per collection
  - Complex documents with well defined metadata
- Images resistant to similarity search
  - High complexity
  - Black-and-white
- Huge volume videos without rich metadata

Links

Project Web Site: www.diligentproject.org

Info: info@diligentproject.org

Software Web Site: www.gcube-system.org