Black Sea Catchment Observation and Assessment System supporting Sustainable Development

Geospatial and Grid infrastructures interoperability in enviroGRIDS

Dorian Gorgan, Denisa Rodila
Computer Science Department
Technical University of Cluj-Napoca
{dorian.gorgan, denisa.rodila}@cs.utcluj.ro

www.envirogrids.net

Outline

- Scope
- Objectives
- Context
- Why, When and Where do we need Grid technology?
- Challenges/Problems
- Solutions and Approaches
- Discussions
- Conclusions
- References

Scope

- **Design and implement an architecture to support integration of geospatial domain (represented by OWS standards) in the Grid environment**

Objectives

- Discuss and analyze different solutions and different approaches for integrating OGC services into the Grid environment based on the gridification level
- Implement and exemplify the analyzed concepts on different OGC services
- Propose new standards in the OGC and OGF collaboration

enviroGRIDS Project

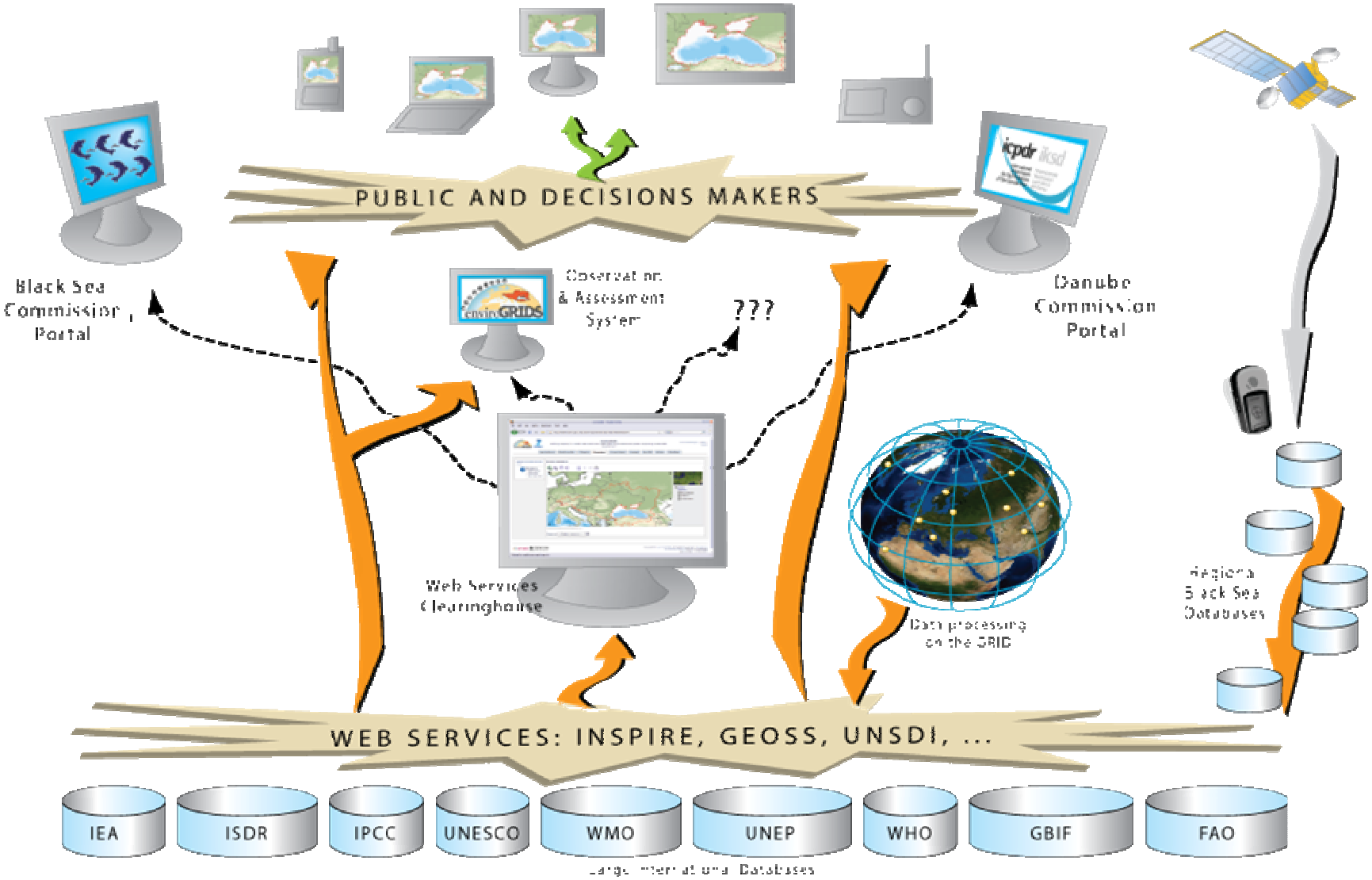
- **enviroGRIDS - Black Sea Catchment Observation and Assessment System supporting Sustainable Development**

  http://www.envirogrids.net/. Funded by the European Commission (April 2009 – March 2013), Contract 226740
- Large consortium of 27 partners
- Coordinator: University of Geneva, Switzerland

Building Capacity for a Black Sea Basin Observation and Assessment System supporting Sustainable Development

Map source: "Eutrophication in the Black Sea region; Impact assessment and causal chain analysis", Global International Waters Assessment (GIWA).

enviroGRIDS



General Objectives

- Gap analysis
- Spatially explicit regional scenarios of development
- Modeling of large scale, high resolution distributed hydrologic processes
- Develop access to real time data from sensors and satellites
- Streamlining the production of indicators on sustainability and vulnerability of societal benefits
- Develop early warning and decision support tools at regional, national and local levels
- Build capacities in the implementation of many new standards and frameworks

Grid Oriented Objectives

- Link, gather, store, manage and distribute key environmental data
- Gridification of applications
- Build capacities in the implementation of several new standards for sharing geospatial data.
- Main references are GEOSS (Open Geospatial Consortium), OGS (Open Geospatial Consortium), and INSPIRE Directive (Infrastructure for Spatial Information in the European Community)

Context

- Open Geospatial Consortium – OGC
- Open Grid Forum – OGF
- gLite OWS - G-OWS
- Research projects: SAW-GEO, CYCLOPS, GDI-Grid, GEO-Grid, DEEGREE
- Collaboration between OGC and OGF should provide the necessary infrastructure for developing tools, software and services for multiple communities

Context

- **Research and development projects working on the interconnection between Grid and GIScience [Maué et al, 2009] :**
  - The British **SEE/SAW-GEO** project

    (http://edina.ac.uk/projects/seesaw/) focuses the integration of security mechanisms into spatial data infrastructures by utilizing Grid technologies. SAW-GEO focuses also on workflows within spatial information systems and is the short form from Development of Semantically-Aware Workflow Engines for Geospatial Web Service Orchestration.
  - The European project Cyber-Infrastructure for Civil protection Operative Procedures - **CYCLOPS**

    www.cyclopsproject.eu) utilizes Grid technologies for civil protection and also operates on spatial data sets.

Context

- The German **GDI-Grid**

  (www.gdi-grid.de) integrates OGC and OGF standards on the German eScience infrastructure D-Grid (www.dgrid.de). Based on three selected use cases (flood simulation, noise propagation and emergency routing) a spatially-enabled Grid middleware will be developed. Also running on D-Grid is the Collaborative Climate Community Data and Processing Grid (C3Grid, www.c3grid.de). C3Grid develops a grid-based research platform for earth systems research.

- The Global Earth Observation Grid - **GEO-Grid**

  is a world-wide initiative for the Earth Science community (www.geogrid.org). It aims to integrate a great variety of Earth Science Data (satellite imagery, geological data, etc.) through virtual organizations while keeping use restrictions on classified data sets.

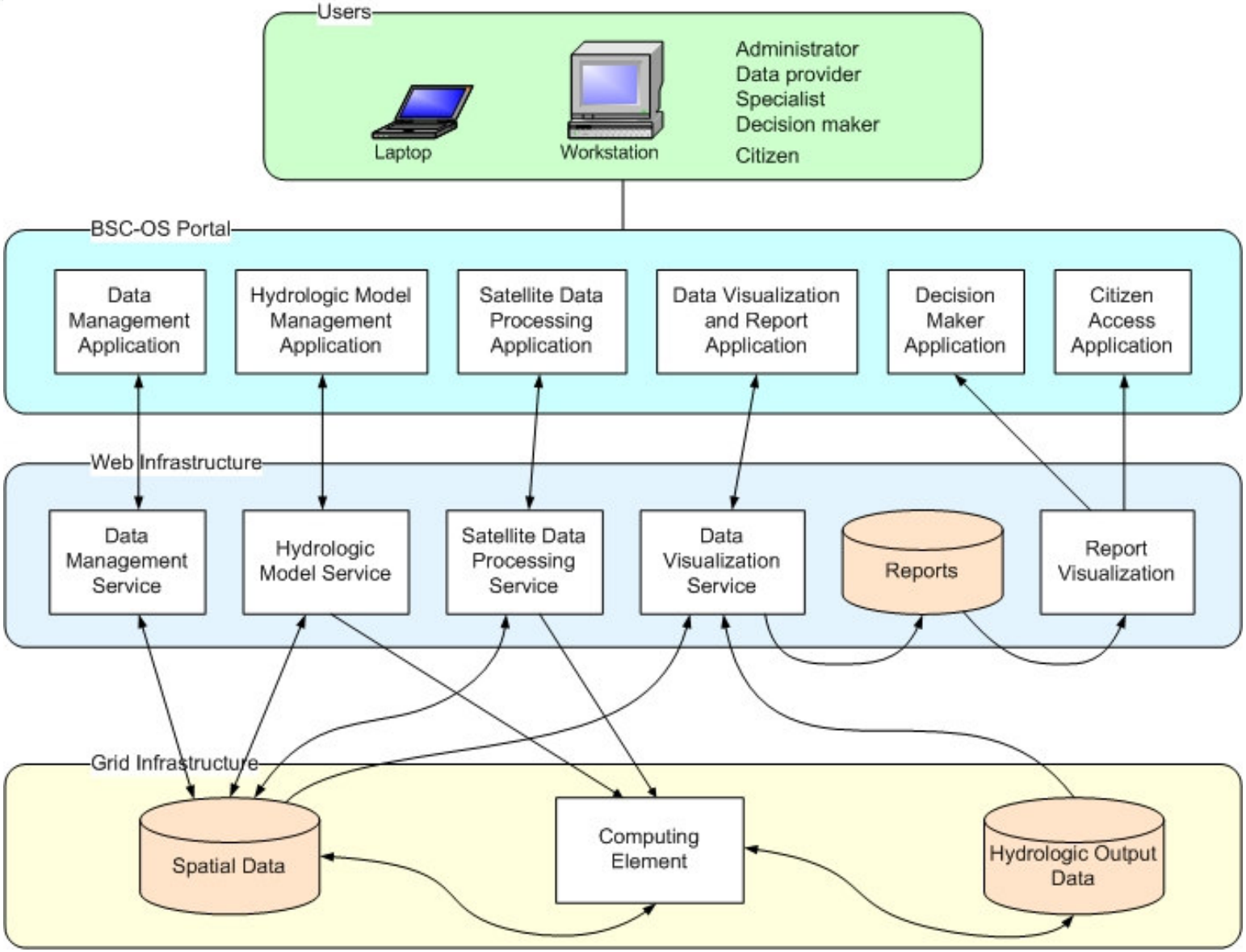
Context

- **DEEGREE** is a Java open source framework which can be used for setting up web based spatial data infrastructures.

  Its entire architecture is developed using standards of the Open Geospatial Consortium (OGC) and ISO/TC 211 (ISO Technical Committee 211 – Geographic Information/Geomatics)

  - *Based on the services implemented in this research project, we plan to perform our Grid integration solutions*
  - *http://www.deegree.org/*
  - *https://wiki.deegree.org/deegreeWiki/*

Data Flow in enviroGRIDS



Challenges

- Geospatial data come from multiple heterogeneous sources
- Geospatial data have to be accessed, integrated, analyzed and presented across a distributed computing environment
- Processing and storing resources in different formats
- Security and digital rights management
- User authentication and authorization

Why do we need Grid technology?

- Handling massive data
- Need for real time data processing
- Information models and architectures
- Catalogs and discovery services (find and access data)
- Data, execution and workflow services (process data)
- High performance computing – HPC
- High throughput computing – HTC
- Significant gain in performance
- Event monitoring and management services
- Workflow management
- Security

Why do we need Grid technology?

- Execute computational intensive calculations on very large amounts of data by using web services, by standard approach
- Provide a distribution of calculations and datasets on grid nodes with possibility of distributed high-speed data transfer

When do we need Grid technology?

- **The advantages introduced by the Grid infrastructure are visible only in certain cases:**
  - the time required to execute a request is considerable larger than the overhead introduced by job creation and management on the Grid
  - several requests are made in parallel
  - a request can be split into several parallel sub-request

When do we need Grid technology?

- **For simple requests, the overhead introduced by the Grid (job creation and management) is not compensated and the execution time is greater**
- **One approach : introduce the Grid only for the requests for which the overhead is compensated and the execution time is improved**
- **To differentiate the requests for which the Grid can bring an improvement from those for which it introduces additional overhead, some analysis have to be made regarding:**
  - the type of requested service
  - the request parameters
  - the type of functionality executed inside the service

OGC Web Service

- **Layered on top of Internet standards: HTTP, URLs, MIME, and XML World Wide Web standards**
- **OWS standards includes:**
  - Web Map Service – WMS
  - Web Feature Service – WFS
  - Web Coverage Service – WCS
  - Web Processing Service – WPS
    - Define basic request-response interactions for remote execution of a service
    - The only service able to store intermediate results at an external resource and use it as input data in a later service call
  - Catalog Service for Web – CSW

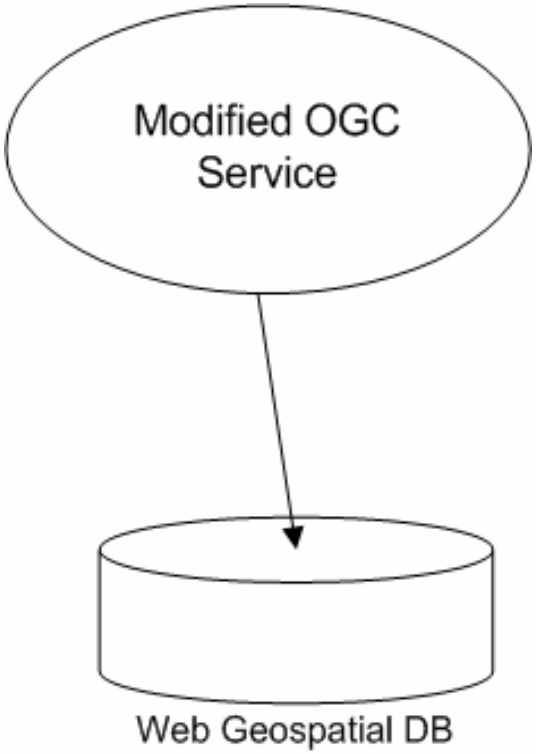
When do we need Grid technology?

- **An estimation should be made regarding the boundary beyond which the advantages of the Grid are visible in the execution time**
- **A modified OGC service will be adapted to support different execution flows depending on the decision made regarding the complexity of the request and the necessity of the Grid as execution environment**
  - execute the service directly
    - on Web databases
    - on Grid database
  - split the initial request into several jobs and send them to execution on the Grid
    - the workers connect to Web databases
    - the workers connect to Grid databases

Where do we need Grid technology?

- Case 1: **The service uses Web Geospatial database.**

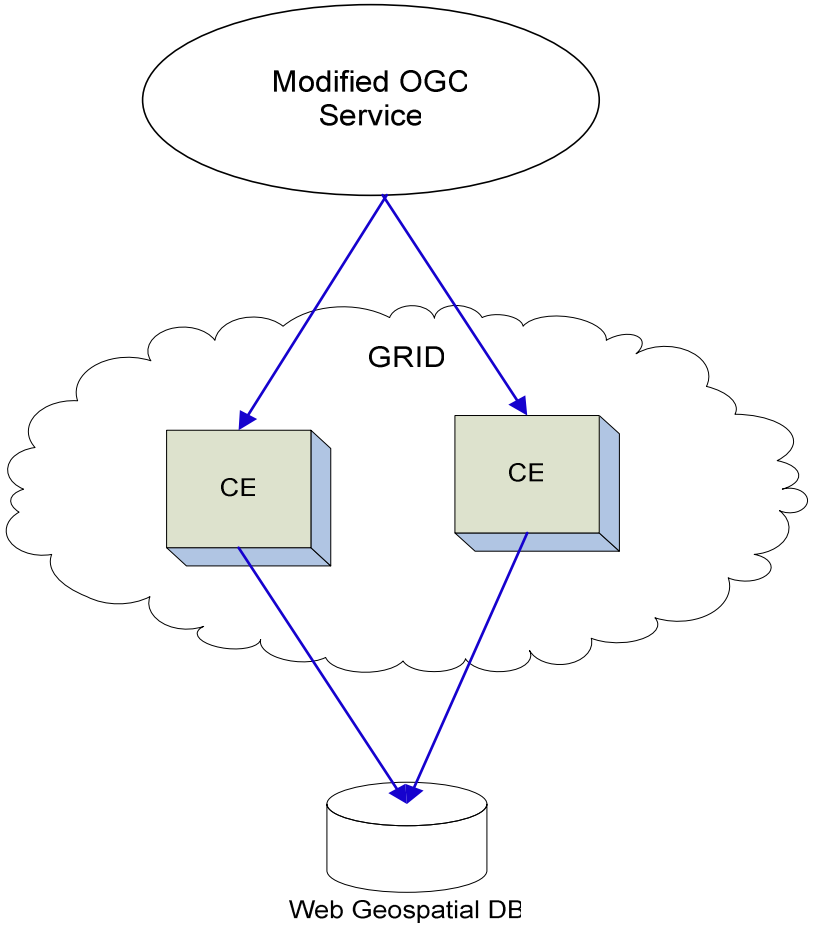
  The simple request (those for which the execution time does not exceed the overhead introduced by Grid) are executed directly on the Web server and the Grid environment is no longer used



Where do we need Grid technology?

- Case2: **The service uses Web Geospatial database but it is using the Grid environment for the execution.**

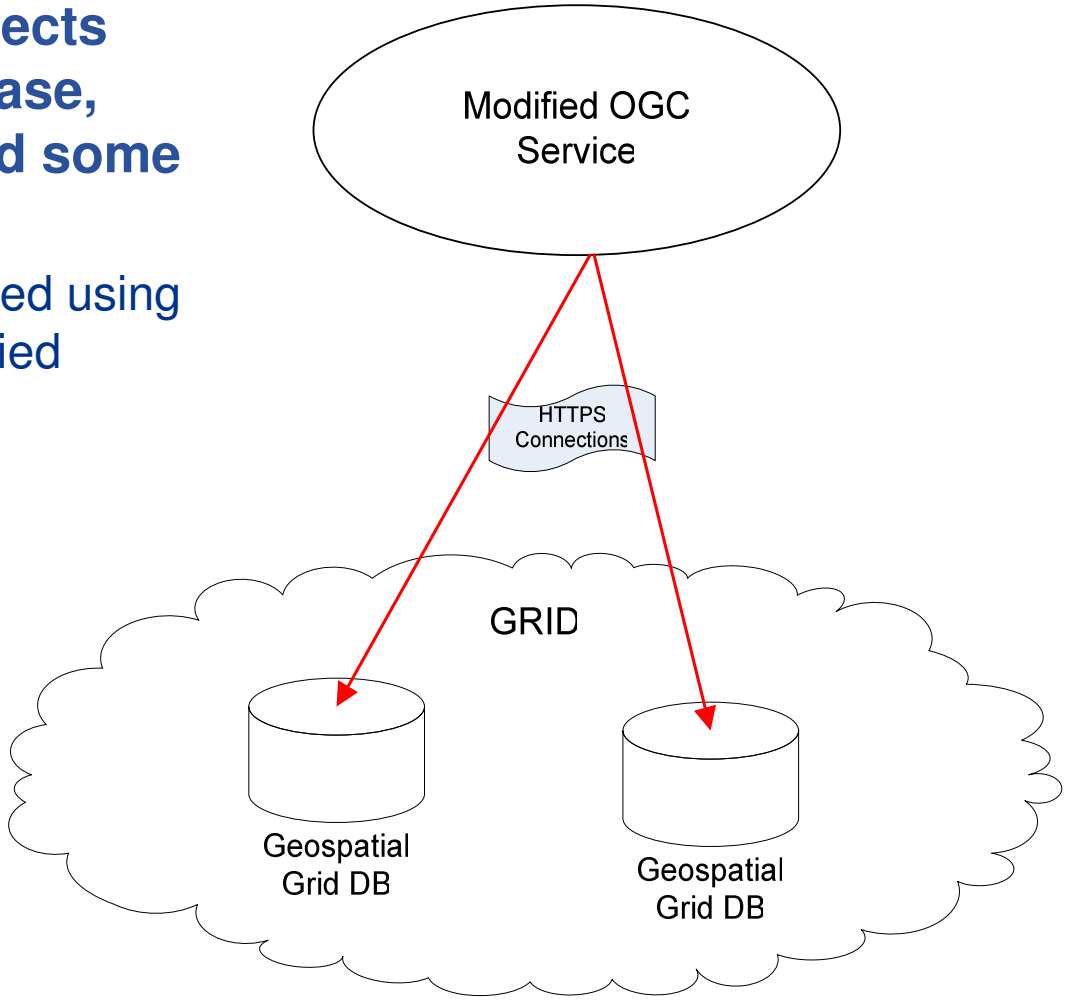
  The request is split into several sub-requests which are executed on individual workers. The workers connect to the Web Geospatial database and retrieve the necessary data.



Where do we need Grid technology?

- Case 3: **The service connects directly to the Grid database, using grid certificates and some special libraries.**

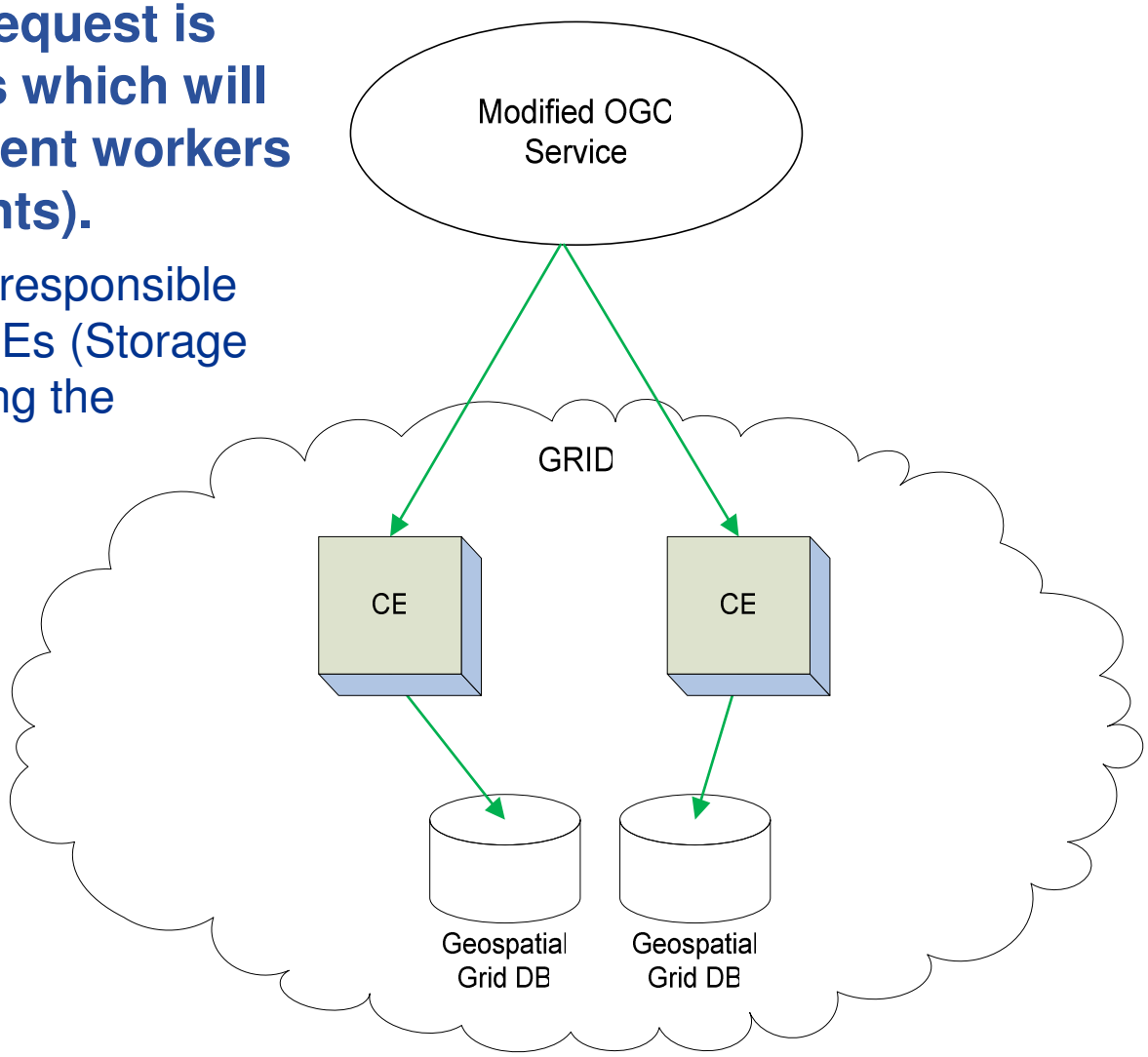
  The connection is established using HTTPS and the data is copied using dedicated scripts.



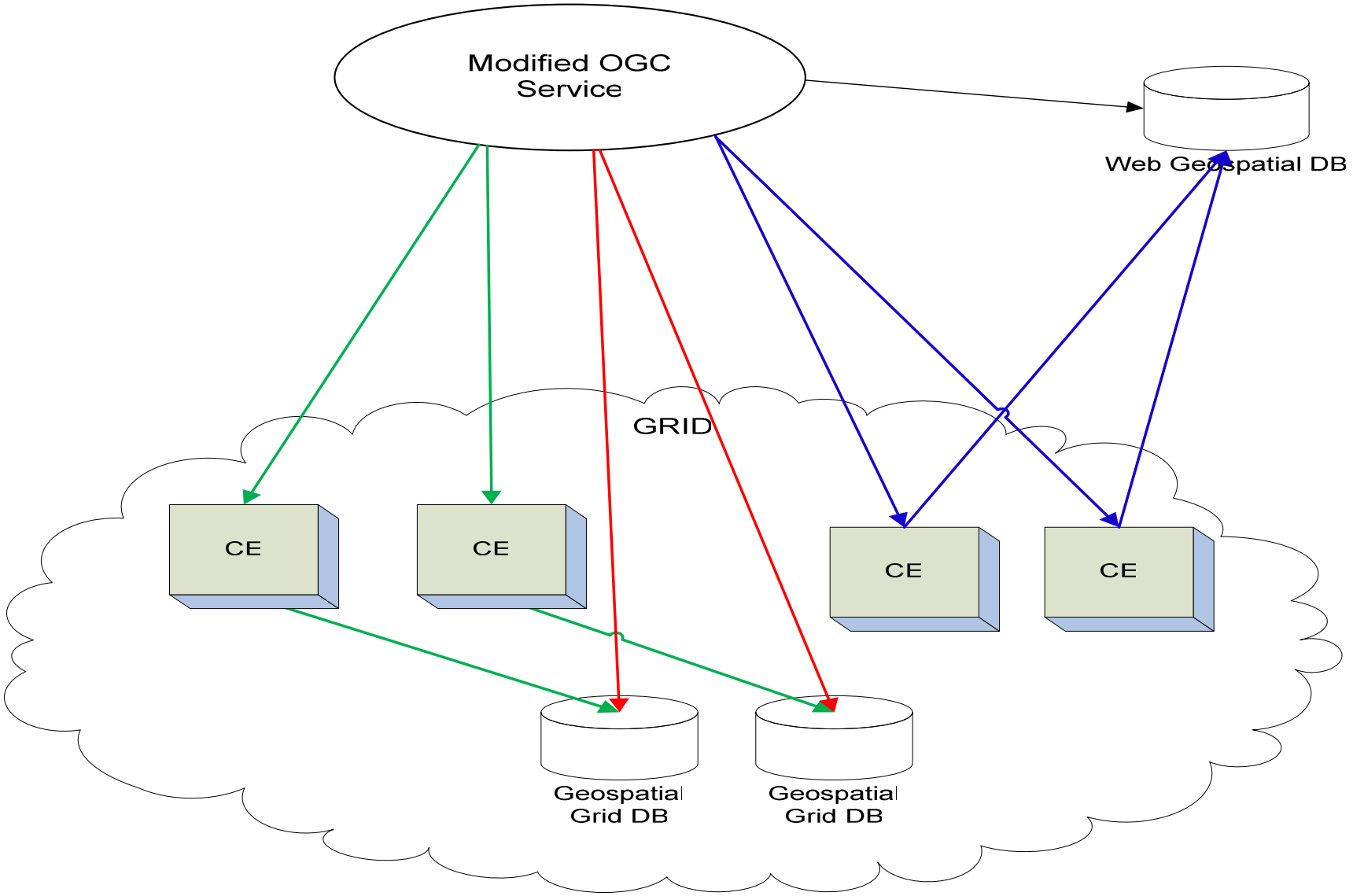
Where do we need Grid technology?

- Case 4: **The service request is split into several jobs which will be executed on different workers (Computation Elements).**

  The workers are each responsible for connecting to the SEs (Storage Elements) and obtaining the necessary data.



Grid technology – General case



Integration challenges – OWS vs Grid Services

- Creation
- Persistence
- Security concepts
- Service description
- Search and discovery
- Messages

Gridification of OGC services

- **Gridification = casting of existing applications and services into the framework of a grid environment**
- **The gridification of OGC services must be done while:**
  - Maintaining the functionality and the interface in the geospatial context
  - Take advantage of the Grid architecture in executing OGC workflow services

Gridification of OGC services

- **The parallelism offered by the Grid technology can be applied at different levels:**
  - Data parallelism
  - Computing parallelism

Partitioning of Spatial Data

- **Three methods for data partitioning:**
  - Object type

    Processing of data can be parallelized by object types, e.g. buildings, transportation areas, water bodies, etc.
  - Computation nodes

    Different operations can be executed on the whole dataset, distributed on different computation nodes
  - Data tiling

    The whole dataset of a territory will be divided into tiles, which can be processed in parallel

Gridification of single OGC service

- **Two points are important while gridification of a single OWS:**
  - Maximize the performance of processing large amounts of spatial data implies optimizing the distribution of data and services within the grid
  - Interconnect the gridified web services to reach a high level of parallelism

Gridification of single OGC service

Existing approaches:

1. **The interface of the OWS is realized outside the grid environment [Di et al., 2003]**
   - A server for each web service is running inside the Grid which contains the full functionality of this web service, including the data management and capabilities
   - The interface represents a facade to the server as a grid service with its functionality
   - An OWS can use the grid-enabled service by accessing the interface at the SDI network.
2. **Encapsulates OWS into a grid layer [Shu et al., 2006]**
   - Additional portTypes (interfaces) are implemented to adapt the needs of the grid
   - The access to OWS interfaces is done by internal invocation of such grid services.

Gridification of OGC workflows

- **The user makes a request to a workflow which contains OGC service calls to retrieve data stored both on the Grid and on the Web. The Proxy component will classify the request as a Grid request and only after that the workflow will be parsed and decomposed in simple service calls similar to the first case**
- **Three different approaches on how to map OGC service workflows structures to a grid environment [Krüger et al, 2008]:**
  1. The grid environment can be used only for executing calculations of a number of single OWS
  2. The grid environment is used for a SDI workflow in realizing workflow of pure grid services
  3. Maintain an SDI service workflow while using the grid environment for calculation and distributed data transfer between connected services

Conclusions on Gridification of OGC workflows

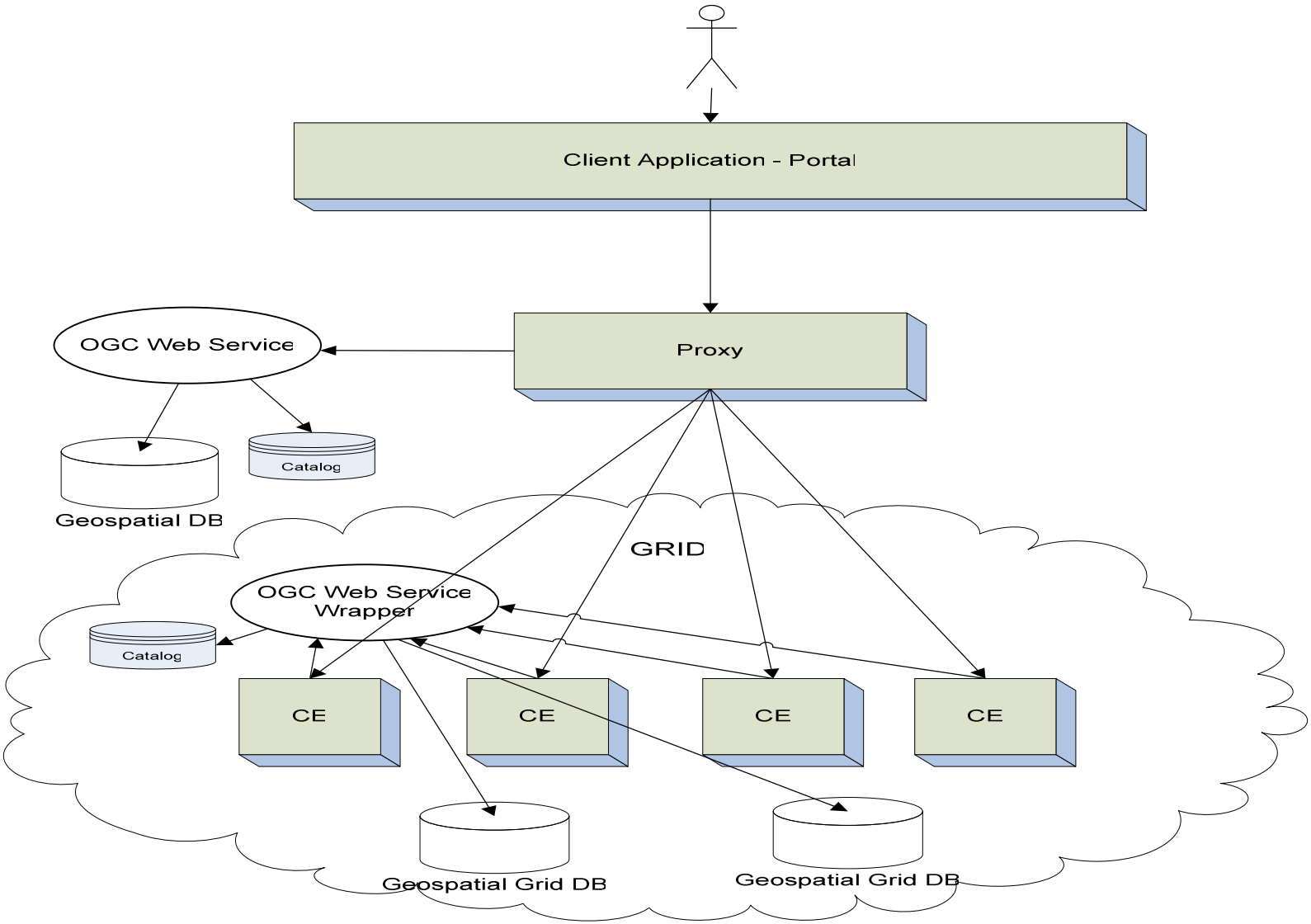
- The first solution is the best choice with an independent gridification of several services
- The second solution can be used if there is no requirement of maintaining an interconnection to the SDI network
- The third solution realizes a distributed processing of calculations and a distributed data and information flow between services

Scenario example

Scenario:

- **The user makes a request to the Portal application**
- **The portal application forwards the request to the Proxy server**
- **The Proxy server identifies the Web and the Grid calls**
  - For the SDI calls it executes the OGC service directly
  - For the Grid calls it uses the Ganga functionalities to submit the job to the Grid (split, execute, merge)
- **The Proxy waits the results from the SDI and Grid environments, merge the final result and send it to the client**

Scenario example

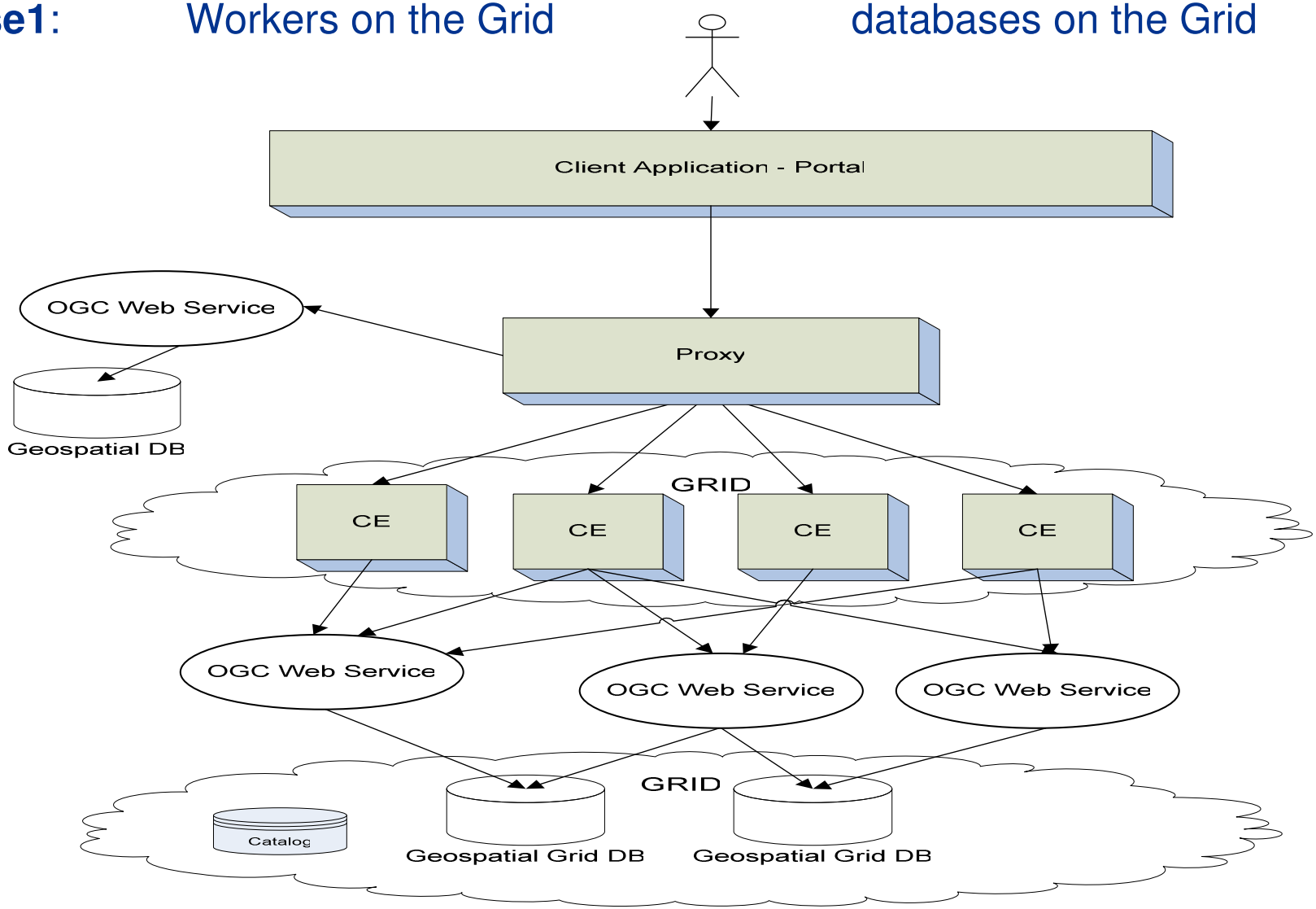


Gridification of OGC services - discussions

- **Depending on the location of the components involved in the OGC interoperability with the Grid, we can distinguish the following cases illustrated also in the following diagrams:**
  - **Case1**: Workers and databases on the Grid
  - **Case2**: Workers on the Grid, databases on the Web
  - **Case3**: No workers, databases on the Grid

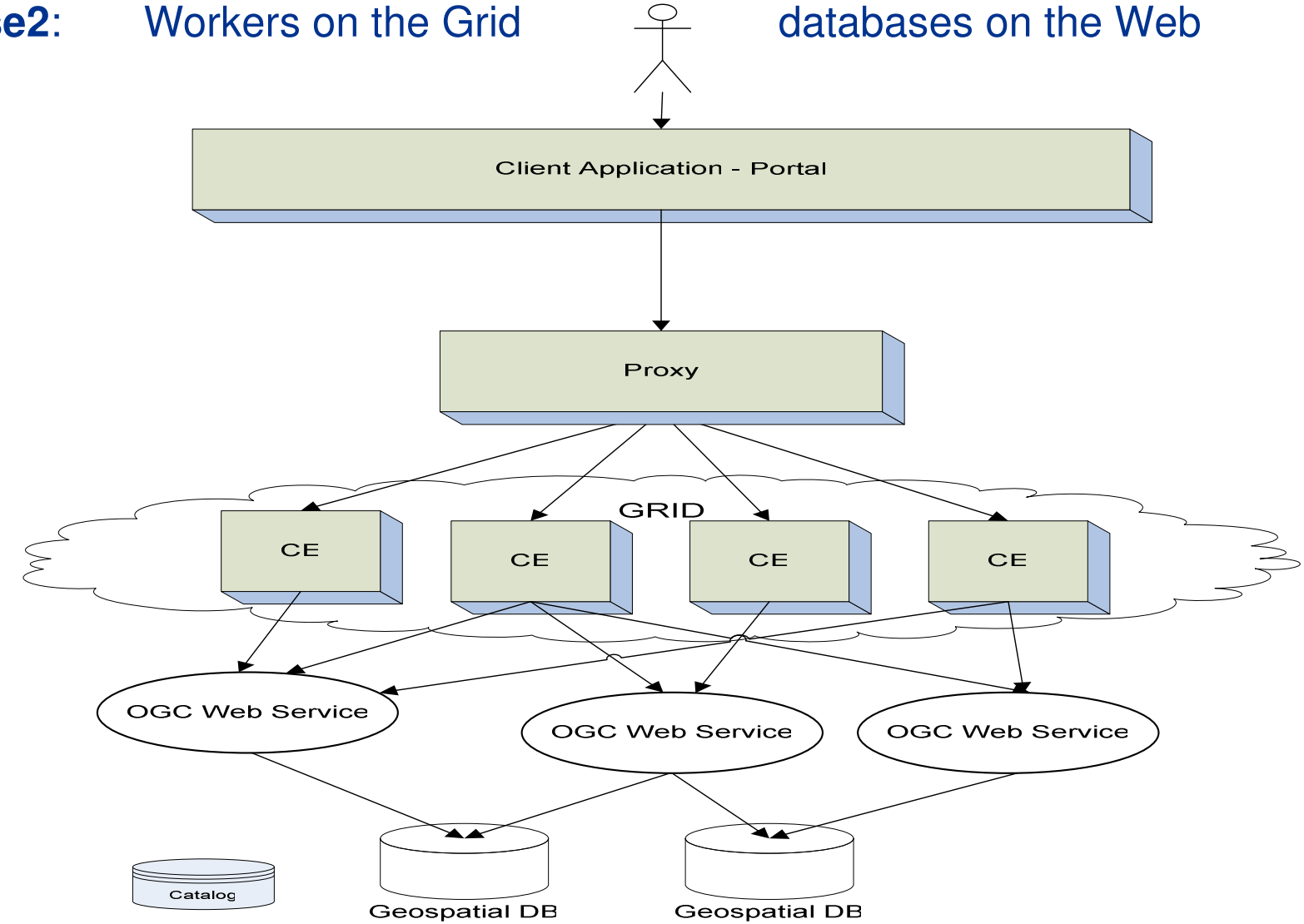
Gridification of OGC services – Case1

Case1: Workers on the Grid databases on the Grid



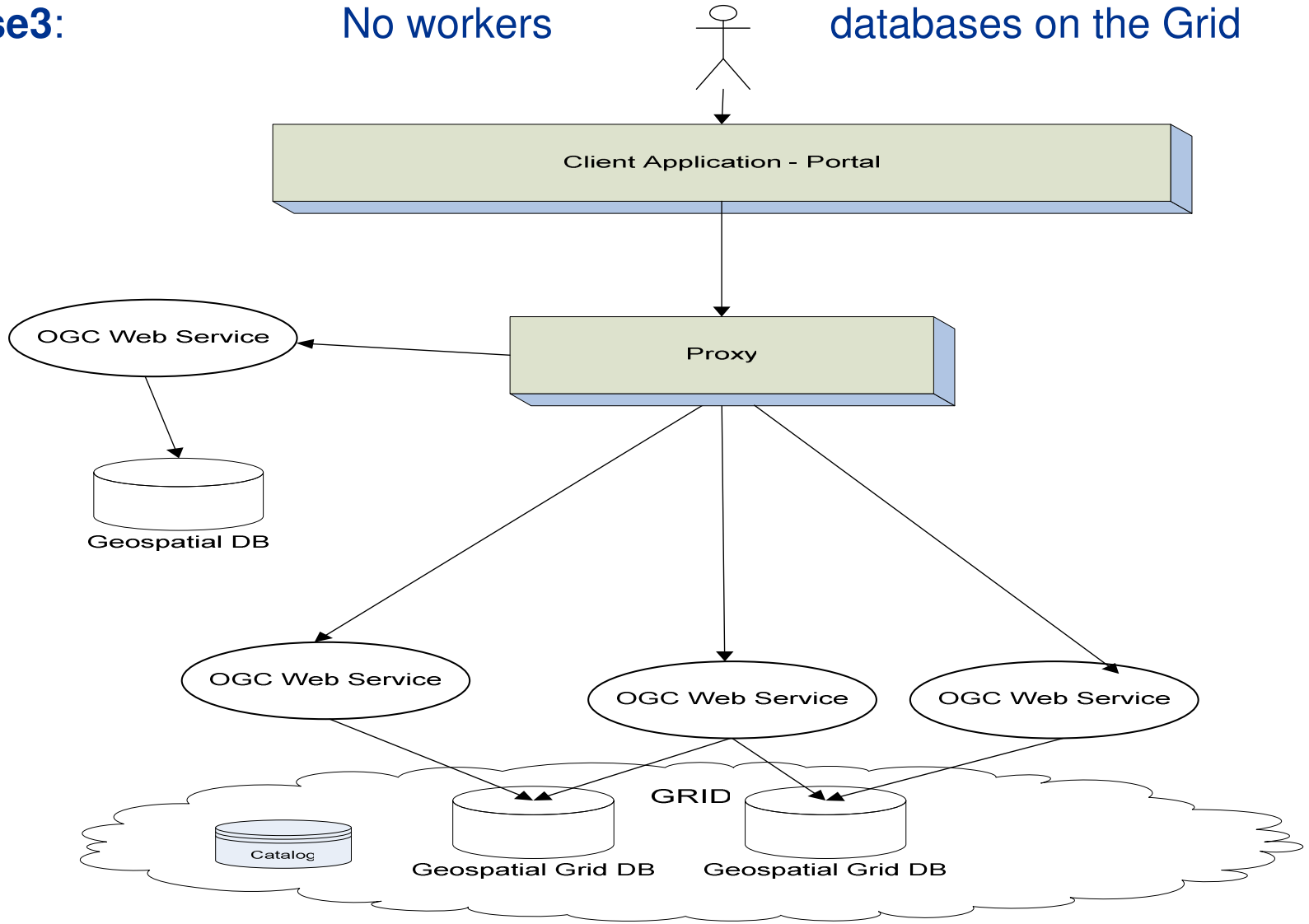
Gridification of OGC services – Case2

Case2: Workers on the Grid databases on the Web



Gridification of OGC services – Case3

Case3: No workers databases on the Grid



Conclusions

- **The integration of OGC Web services into the GRID environment is a complex process and has the following target points:**
  - Data parallelism management
  - Data security
  - Complex execution parallelism
- **The OGC service interoperability has practical applications in EnviroGRIDS project through Portal access**

Black Sea Catchment Observation and Assessment System supporting Sustainable Development

Thanks or attending!

Dorian Gorgan, Denisa Rodila
Computer Science Department
Technical University of Cluj-Napoca
{dorian.gorgan, denisa.rodila}@cs.utcluj.ro

www.envirogrids.net