IL NUOVO CIMENTO Vol. 28 C, N. 2 Marzo-Aprile 2005
DOI 10.1393/ncc/i2005-10182-5

# A grid portal for Earth Observation community(*)

G. Aloisio, M. Cafaro, G. Cartenì, I. Epicoco and G. Quarta

*Center for Advanced Computational Technologies ISUFI, Università di Lecce*
*National Nanotechnology Lab/INFM CNR*
*via per Arnesano, 73100 Lecce, Italy*

(ricevuto il 31 Gennaio 2005; approvato il 31 Maggio 2005)

**Summary.** — Earth Observation techniques offer many powerful instruments for Earth planet study, urban development planning, military intelligence helping and so on. Terabytes of EO and geospatial data about lands, oceans, glaciers, cities, etc. are continuously downloaded through remote-sensing infrastructures and stored into heterogeneous, distributed repositories usually belonging to different virtual organizations. A problem-solving environment can be a viable solution to handle, coordinate and share heterogeneous and distributed resources. Moreover, grid computing is an emerging technology to solve large-scale problems in dynamic, multi-institutional Virtual Organizations coordinated by sharing resources such as high-performance computers, observation devices, data and databases over high-speed networks, etc. In this paper we present the Italian Grid for Earth Observation (I-GEO) project, a pervasive environment based on grid technology to help the integration and processing of Earth Observation data, providing a tool to share and access data, applications and computational resources among several organizations.

PACS 02.70.-c – Computational techniques.
PACS 01.30.Cc – Conference proceedings.

## 1. – Introduction

A huge quantity of Earth Observation (EO) data is produced daily by many satellites launched by several worldwide space agencies. Data belong to different types such as optic, infrared, radar images, etc. Generally, these images represent semi-finished products and the end user further processes these in order to extract relevant information very useful in different scientific areas, such as geology, climatology, oceanography, natural disaster monitoring and prevention, and so on. Often, data coming from a specific

(*) Paper presented at CAPI 2004, 8° Workshop sul calcolo ad alte prestazioni in Italia, Milan, November 24-25, 2004.

source can contain complementary information with respect to other sources and integrating all of these data could result in further, innovative and exhaustive knowledge about a topic of interest. Nevertheless, securely and transparently storing, managing, and accessing this huge amount of data spread over distributed systems is a challenging problem. Moreover, the time needed to process and integrate the data is very long and for specific EO applications (such as emergency response, natural disaster monitoring and prevention) the use of high-performance resources greatly reduces the time required for data processing. Even though the availability of these resources is limited, their sharing among different organizations gives a clear advantage.

A problem-solving environment [1, 2] can be a viable solution to handle, coordinate and share heterogeneous and distributed resources. A PSE can be designed to provide transparent access to heterogeneous, distributed computing resources for collaborative computational science and engineering. From the architectural point of view, grid computing [3, 4] is an emerging technology to solve large-scale problems in dynamic, multi-institutional VOs coordinated by sharing resources such as high-performance computers, observation devices, data and databases over high-speed networks, etc.

The aim of our Italian Grid for Earth Observation (I-GEO) project [5] is the development of a pervasive environment, based on grid technology, to help the integration and processing of EO data, providing a tool to share and access data, applications and computational resources among several organizations. Such an environment has been designed and developed taking into account these issues and now we are testing it on a national grid. The people involved in the testbed are computer scientists, physicists, experts in Earth Observation domain and so on. The heterogeneity of the involved people, guarantees that required feedback will be properly considered.

The outline of this paper is as follows: in sect. **2** we illustrate the main requirements we have identified; in sect. **3** we show the architecture of the system analysing in depth each module of the proposed environment and finally, in sect. **4**, we conclude the paper highlighting future work.

## 2. – Requirements and main challenges

Our aim is to design and build a Grid-Based Problem Solving Environment to allow sharing of EO and geospatial data, software and computational resources for EO and geospatial data processing among different organizations, through high-speed networks (see fig. 1).

The following requirements have been identified for this grid environment:

- Sharing of computational resources among different organizations.
- Sharing of EO and geospatial data.
- Sharing of software resources (with particular emphasis on EO and geospatial data processing applications).
- Management of resources. This is fundamental to allow a dynamic and flexible configuration of the grid.
- Transparent access to computing resources trough a graphical interface.
- Efficient usage of the resources; this implies handling jobs scheduling, resource management and fault tolerance.

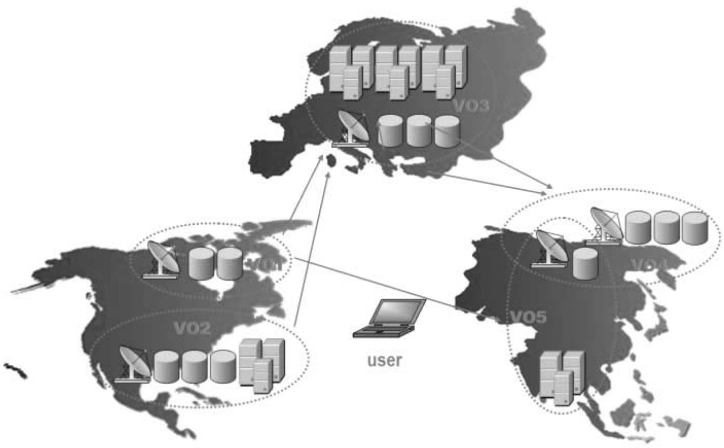


Fig. 1. – The geographically spread resources must be shared among different VO.

– Efficient usage of the network links. Indeed, the amount of data to be transferred from end users to nodes and vice versa is quite large. The data movement, when considering that it can be a bottleneck, impacts on the global performance of the architecture.

– Composing and compiling new processing applications based on existing ones: the system should allow users to build complex applications by means of a friendly user interface that will hide low-level details related to grid infrastructure and resources.

– Supporting different data formats. This requirement comes from the need to allow applications to interact with data provided by different producers.

In order to meet these important requirements, the architecture has to face several challenges such as efficiency, extensibility, security, robustness and flexibility. More in depth:

*Centralized/Peer-to-peer approaches*: a centralized approach is very easy to manage but it introduces many inherent problems from a performance, scalability and fault tolerance point of view. In our architecture we consider a peer-to-peer infrastructure which is highly scalable and extensible.

*Heterogeneity*: a grid environment often is highly heterogeneous from a hardware (*i.e.* different platforms, networks, etc.) and software (*i.e.* different DBMSs, operating systems, etc.) point of view. Moreover, systems which try to bring together programs, services and protocols implemented by different developers cannot communicate unless using common standards, *e.g.*, for network communication. To address this issue we designed a cross-DBMS (by means of several wrappers, one for each specific DBMS) and cross-platform, based on a Service Oriented Approach.

*Security*: in a distributed environment, security (confidentiality, integrity, etc.) is an extremely complex and multi-faceted problem. Resource owners (represented by trusted administrators) at each site must be able to control which principals can access the resources and under what conditions (authorization requirement and access policies). We address this issue performing mutual authentication between the system actors, authorization processes based on Access Control Lists (ACL) and data encryption/signing to prevent message eavesdropping and tampering.

*Efficiency/robustness*: the system has to provide efficient and stable access to distributed Earth observation system data sources. As pointed out by internal tests, the software performs well in terms of stability, robustness and efficiency. We plan to test the system performance in a European Testbed (such as GridLab [6, 7]) and to publish issues and results in a future work.

## 3. – Architecture

In order to meet the previous requirements, the designed architecture includes the following modules, as depicted in fig. 2:

- The I-GEO Web interface. This component allows accessing grid services and resources. It permits to search data over the distributed catalogue, to use the services and to administer the resources. The Web interface consists of three main sub-components: the data search engine interface, the application composition/submission interface and the system management interface.
- The I-GEO Information System: it is an information source that contains an ontology about applications usually employed in the remote-sensing field. Information related to an application includes: a) input and output data formats; b) application capabilities (we distinguish applications with pre-processing capability, post-processing or data format conversion capability); c) information needed to launch the application, *i.e.* hostname, pathname, shared libraries on which the application depends, environment variables, application arguments and so on.
- The I-GEO Distributed Data Management System: based on a common metadata schema for describing EO and geospatial products, it uses some modules belonging to the Grid Relational Catalog Project [8] to provide transparent, efficient and secure heterogeneous data integration.
- The I-GEO Workflow Management System: an integrated workflow management system for EO applications that includes a web-based user interface and a resource manager optimized for EO applications; it interacts with underlying services.
- The I-GEO Scheduler and Monitoring Module: it is the component responsible for job scheduling and file transfer taking into account available computational resources, the locations where the datasets are stored and where the services are installed on, and several performance parameters provided both by the Network Weather Service [20] and by a historical performance archive.

**3**·1. *I-GEO Web interface*. – This component is the presentation layer of the PSE: it consists of a set of interfaces (implemented using classics web technologies such as dynamic web pages using Java server pages, Java applets, CGI and so on) that exploit the logic provided by the business logic modules:

- The Data Search Engine Interface leverages fast CGI programs (for performance reasons), to retrieve data and present them (via HTTP protocol) within properly formatted HTML pages. This sub component is layered on top of the Distributed Data Management System (see subsect. **3**·3) and allows users submitting geographic or metadata-based queries (the users can define a bounding box over an Earth map or define as search criteria a time interval or an image type).

- The Application Composition/Submission Interface allows building complex applications by means of a friendly user interface (a Java applet) that hides low-level details related to grid infrastructure and resources. This component relieves the user from the burden of defining low-level details and implements mechanisms assist and guide the user during the specification of her workflow.

- The System Management Interface is the component that permits configuration of grid resources. Through this component, the user administrator has the possibility to properly configure the resources and the contributor can configure services and applications; allowed operations include addition, modification and deletion of resources and services. This component interacts directly with the Information System through the exchange of XML messages.

**3**˙2. *I-GEO Information System.* – In distributed environment, the description and discovery of resources and services (like a software component or computational resource), is a fundamental issue [9]. As a matter of fact, the involved resources can be heterogeneous, highly dynamic, geographically distributed and so on. The information about these resources is needed, for instance, in order to allow automatic jobs submission, to schedule job execution according to the available scheduling algorithms, and to allow for example resource brokering. In this context, information plays a central role. In our project, we have designed an *ad hoc* configuration repository specialized to describe applications and tools for remote-sensing data processing and management [10]. In our approach, we have considered three kinds of resources: hardware, software and data. For each component, we have derived and implemented an information model that describes the component itself.

- The MetaSoftware schema: we have collected a set of relevant information able to completely describe a software object. The collected information mainly belongs to two classes: information characterizing an application and information about its performance. The former is useful for resource discovery, the latter can be used by the scheduler to define a submission schedule that minimize, for example, the completion time.

- The MetaData schema: in order to realize a MetaData schema that involves the most important information about remote-sensing data, we have considered the ISO TC/211-19115 standard [11]. From this standard we have derived a set of raw metadata. The most important metadata we have considered address: product identification and distribution, data quality, platform and mission, spectral properties, maintenance, generic information, spatial representation, reference system and other information related to the TIFF data format. This set of raw metadata is mapped onto a uniform metadata set derived from the ISO standard itself in order to have a homogeneous set related to the following missions: ENVI, ERS1, ERS2, RadaraSat1, SLR1, SLR2, SRTM. We have obtained a set comprising about 200 metadata that describe all of the mentioned kinds of Earth Observation products thoroughly. This metadata set is structured utilizing a relational schema. Moreover, the CEOS [12] data format specification is considered in order to achieve a good description of input and output data formats for remote-sensing applications. We have considered, for each format, the files associated to the product.

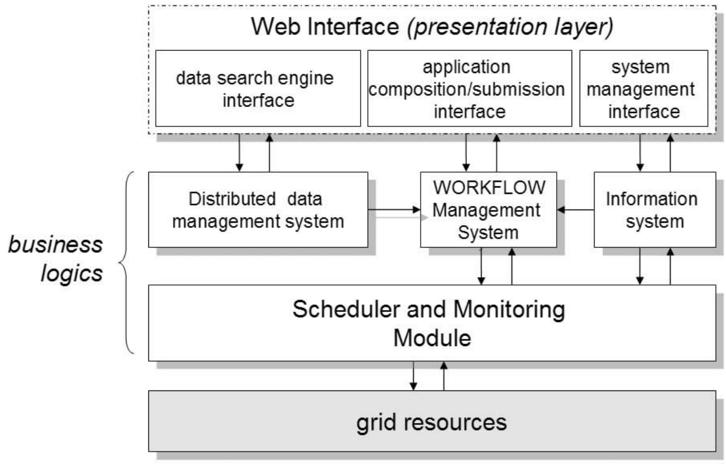


Fig. 2. – The mains components of the I-GEO system.

– Computational resource information schema: the extension of the information schema related to computational resource is needed in order to enhance scheduling algorithms and, more generally, to address grid-aware application requirements. Taking into account that the grid is inherently dynamic, *i.e.* machine load and availability, network latency and bandwidth change continually, we have adopted a hybrid solution keeping all of the significant aspects from the existing approaches. We have analyzed the Globus information schema, the GridLab information schema considering also other schemas, *e.g.*, GLUE and Nordugrid schema.

**3**˙3. *I-GEO Distributed Data Management System.* – The I-GEO Distributed Data Management System [13] (see fig. 3) consists of several components such as the GRelC Service (GS to access data resources), the Enhanced GRelC Gather Service (EGGS—to collect/merge data and forward queries) and a Web Application (the interface already described in subsect. **3**˙1). The GS and the EGGS modules (already implemented at the CACT/ISUFI of the University of Lecce) belong to the Grid Relational Catalog Project [8,14], which provides efficient, transparent and secure data access and integration and management services in a grid environment.

As grid middleware in the architecture we used the Globus Toolkit [15]. It is released under a public license and successfully used in many Grid projects. Currently we have been developing these services as Web Services, exploiting the gSOAP Toolkit [16]. The gSOAP toolkit is well suited for the conversion of legacy application using SOAP because its main feature is a transparent SOAP API. As a matter of fact, gSOAP hides irrelevant SOAP-specific details from the user through the use of compiler technology. The gSOAP stub and skeleton compiler can be used to automatically map native and user-defined C and C++ data types to semantically equivalent SOAP data types and vice versa. SOAP interoperability is thus achieved without explicit knowledge of SOAP details. To guarantee a secure data channel, we also implemented Grid Security Infrastructure (GSI) [17] support, available as a gSOAP plugin [18].

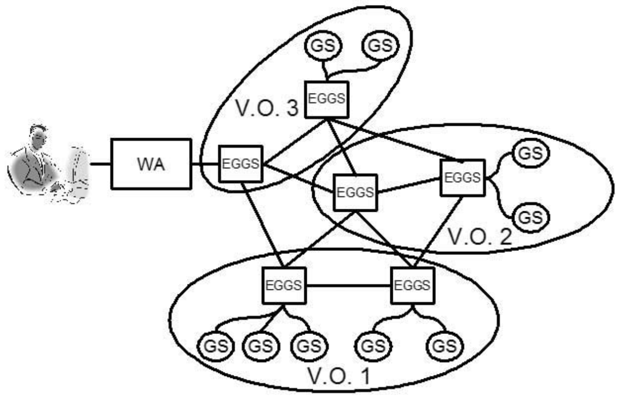


Fig. 3. – The I-GEO Distributed Data Management System.

**3**˙4. *I-GEO Workflow Management System.* – The Workflow Management System interacts with the information system to get information about the status of resources, to monitor submitted jobs and to broker grid resources according to application requirements; it includes a resource management module optimized for grid application described as workflow (fig. 4). This module allows: i) checking the consistency of workflows submitted by users; ii) detecting the coherence of the workflow with respect to the semantic of the involved applications in the EO context (this mean that, for istance, if the user instantiate erroneously a pre-processing application having as ancestor a post-processing application, then the system will warn the user); iii) checking data format compatibility between linked applications.

Sharing the user's knowledge is also valuable, thus the Workflow Management System includes the possibility to share the definition of complex application for three main reasons: a) a user's workflow can be used to compose an even more complex application; b) the workflow can be submitted again without redefining it each time; c) an EO scientist can share her novel application with all of the users that can access the Problem Solving Environment provided by the grid portal.

**3**˙5. *I-GEO Scheduler and Monitoring Module.* – The Scheduler and Monitoring Module provides task scheduling on the grid resources. This module is in charge of job submission, input/output file staging and job tracking. The job scheduling process applied to a grid environment is gaining considerable attention. In the context of EO applications, a frequent problem is the use of a huge amount of data, an aspect that sometimes implies job execution minimizing the data transfer rather than the execution time of the application. The I-GEO Scheduler and Monitoring Module uses an extended and integrated information system layered on several available tools: the iGrid Information System [19] to gather both static and dynamic information about filesystem, local resource manager (PBS, LSF, etc...); the NWS [20] package to gather forecasting information about CPU load, memory availability, network bandwidth and latency; and it accesses iGrid to retrieve the available application data too. Because of its component-based structure, such an integrated Information System requires a common interface to allow a uniform data access. Through this interface, the scheduler is able to obtain the needed information as soon as it is available reducing the communication overhead. The integrated information

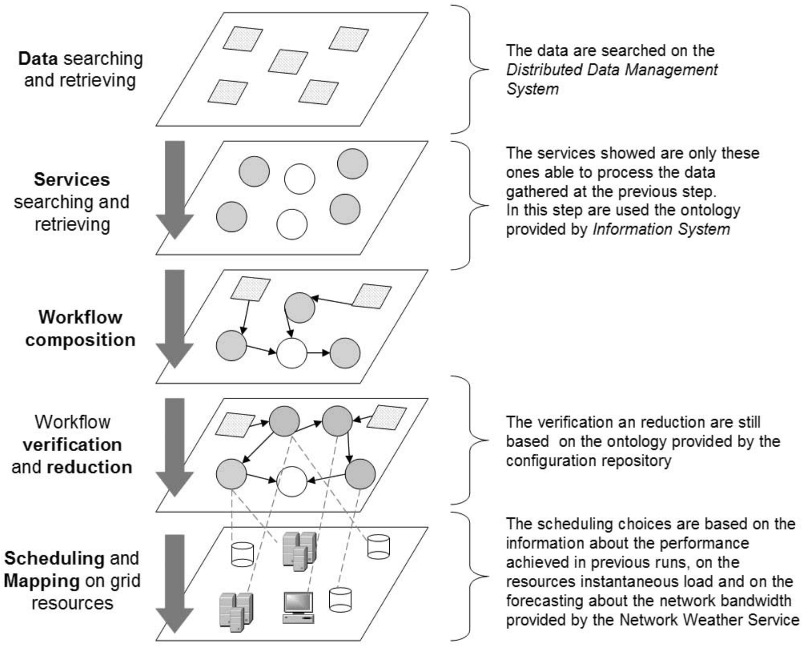


Fig. 4. – The I-GEO Workflow Management System.

system is based on the relational data model, uses SQL as query language, and supports postgresql and mysql DBMS as back-end. Our information system can also be thought as a wrapper to join heterogeneous information sources in relational DBMS.

The scheduler adopts an adaptive approach: it does not use a single algorithm (we are dealing with a NP-complete problem) but it uses a well-known set of scheduling algorithms, each one presenting a good behaviour in a bounded region of a bi-dimensional space defined by the heterogeneity level of Grid resources and applications.

Under conditions of high heterogeneity of grid resources and low heterogeneity of applications, the scheduler runs an algorithm (an online heuristic) which has been tested to consistently better minimize the Completion Time makespan. In order to meet the requirements of the architecture, the scheduler is able to modify, remove and insert new algorithms to obtain a better response in pre-defined conditions of heterogeneity of resources and applications. The only requirement is to use algorithms which minimize the Completion Time makespan of the scheduled workload.

## 4. – Results and conclusions

The proposed environment is currently being tested on a national grid testbed based on SPACI (Southern Partnership for Advanced Computational Infrastructures [21]) geographically spread resources; this infrastructure can provide an aggregated computational power of about 1800 Gflops.

The environment we propose is a PSE; it must satisfy criteria like usability, reliability, fault tolerance, robustness; it is layered on several known grid technologies for managing distributed resources. The test we performed are mainly focused on the evaluation of

TABLE I. – *Hardware resources provided by the PSE.*

| Description | Operative system | Location |
|---|---|---|
| 60 nodes, 2 proc. Each HP XC 6000, Itanium II | RedHat Enterprise 3 | Lecce |
| 4 nodes, 2 proc each, DEC Alpha 733 MHz SMP | Tru64 unix v5.1A | Bari |
| 4 nodes, 4 proc each, Compaq ES40 AlphaServer SC | Tru64 unix v5.1A | Lecce |
| 1 node, single proc. PC Intel Pentium III 850 MHz | RedHat v7.2 linux | Bari |
| 5 nodes, single proc, PC AMD Athlon 1.8 GHz | Mandrake linux v10.1 | Lecce |

accessibility of the underlying grid infrastructure, usability and PSE expressiveness in terms of number of different problems that can be handled by our environment. The testbed is currently composed by several workstations and clusters and many software applications, as detailed in tables I and II.

All of the applications are currently installed on all of the resources belonging to the testbed. These applications can be used to produce Synthetic Aperture Radar (SAR) interferograms. SAR interferometry [22] has emergeg as a very promising mapping technique that has the potential to routinely provide quantitative information on height, deformation and change of Earth surface, allowing a full geo-coding of SAR imagery [23]. This technique involves several processing steps: two SAR images (acquired form a satellite or spacecraft from two different positions) are first focused, then co-registered (in order to achieve a perfect overlapping) and finally are processed through an interferometric software to achieve a semi-finished product that contains information about the topology and the dynamic of an Earth surface region.

Through our PSE, users can compose the full interferometric processing chain in a few and simple steps by means of a tool for composing direct acyclic graphs. In particular we tried to use our PSE to define an interferometric chain, steering a 100 km by 100 km SAR frame; we have modified resource status during job execution to simulate a resource failure. The system promptly detected the fault and activated task rescheduling. From the usability and reliability points of view, the users can simply drag

TABLE II. – *Software resources installed on hardware resources.*

| Executable | Description |
|---|---|
| Mainflyn | Phase unwrapping by Flynn's min discontinuity algorithm |
| Maindiff | Surface difference and RMS measure |
| RangeFilter | Sar range filter |
| AzimuthFilter | Sar azimuth filter |
| Coregistrer | Images coregistration ruotine |
| Interfero | Sar interferometric processor |
| InterferoFilter | Sar interferometric filter |

and drop the available services in their work area and link them, specifying the needed processing parameters in automatically generated windows. Our PSE is also able to notify the about for a possible misleading configuration or mistake when linking with an incompatible data set, it also suggests, whenever possible, the proper way to compose the services or tries to automatically convert the data in compatible data format.

In this paper, the I-GEO project, a Grid-based Problem Solving Environment for remote-sensing data processing has been presented. In order to build an efficient environment, several aspects have been considered: the analysis and characterization of resources, their efficient usage, the easy and efficient composition of the user's processing applications, and so on. Each aspect has been investigated and the architecture of the modules that have been developed has been described. Finally, we plan to migrate towards the emerging WSRF standard [24].

* * *

This I-GEO project has been developed within the GRID.IT, FIRB 2001 project, aimed at defining, implementing and applying innovative solutions for network computing enabling platforms, oriented towards scalable VOs and based on the grid computing paradigm. The GRID.IT project was funded by the Italian Ministry of Education, University and Research (MIUR). For further information, readers should refer to the Web page `http://www.grid.it`.

REFERENCES

[1] Walker D., Rana O. F., Li M., Shields M. S. and Huang Y., *Int. J. Concurrency: Practice and Experience*, **12**, no. 15 (2000) 1455-1480.

[2] Gallopoulos E., Houstis E. N. and Rice J., *IEEE Comput. Sci. Eng.*, **1**, no. 2 (1994).

[3] Foster I., Kesselman C. and Tuecke S., *Int. J. High Performance Computing Appl.*, **15** (2001) 200.

[4] Foster I. and Kesselman C., *The Grid: Blueprint for a New Computing Infrastructure* (Morgan Kaufmann) 1999.

[5] Italian Grid for Earth Observation (I-GEO) Project. `http://leonardo.unile.it/igeo`.

[6] GridLab Project. `http://www.gridlab.org`.

[7] Aloisio G., Cafaro M., Epicoco I., Lezzi D., Mirto M. and Mocavero S., *The Design and Implementation of the GridLab Information Service*, in *Proceedings of Gird and Collaborative Computing, Shanghai, 2003* (GCC2003).

[8] The GRelC Project: *An Easy Way to Manage Relational Data Sources in the Globus Community*. `http://gandalf.unile.it/`.

[9] Czajkowski K., Fitzgerald S., Foster I. and Kesselman C., *Grid Information Services for Distributed Resource Sharing, Proceedings of the Tenth IEEE International Symposium on High-Performance Distributed Computing (HPDC-10)* (IEEE Press), August 2001.

[10] Aloisio G., Cafaro M., Epicoco I. and Quarta G., *Information Management for Grid-Based Remote Sensing Problem Solving Environment*, in *Proceedings of the International Conference on Parallel and Distributed Processing Techniques and Applications (PDPTA 2004), June 21 to 24, 2004, Las Vegas (Nevada) USA*, Vol. II (CSREA Press), ISBN 1-932415-24-6, pp. 887-893.

[11] ISO TC 211 standard of the International Organization for Standardization. `http://www.isotc211.org`.

[12] CEOS format specifications of the Committee on Earth Observation Satellites: `http://www.ceos.org`.

[13] Aloisio G., Cafaro M., Fiore S. and Quarta G., *A Grid-Based Architecture for Earth Observation Data Access, Proceedings of the 2005 ACM Symposium on Applied Computing (SAC 2005), March 13-17, 2005, Santa Fe, USA* (ACM Press), ISBN 1-58133-964-0, Vol. I, pp. 701-705.

[14] Aloisio G., Cafaro M., Fiore S. and Mirto M., *The GRelC Project: Towards GRID-DBMS. Proceedings of Parallel and Distributed Computing and Networks (PDCN), IASTED, February 17 to 19, 2004, Innsbruck, Austria.*

[15] The Globus Project, `http://www.globus.org/`.

[16] Van Engelen R. A. and Gallivan K. A., *The gSOAP Toolkit for Web Services and Peer-To-Peer Computing Networks. Proceedings of IEEE CCGrid Conference, May 2002, Berlin*, pp. 128-135.

[17] Tuecke S., Grid Security Infrastructure (GSI) Roadmap. Internet Draft 2001 (2001). www.gridforum.org/security/ggf1-200103/drafts/draft-ggf-gsi-roadmap-02.pdf].

[18] Aloisio G., Cafaro M., Lezzi D. and Van Engelen R., *Secure Web Services with Globus GSI and gSOAP, Proceedings of Euro-Par 2003, 26th-29th August 2003, Klagenfurt, Austria*, *Lecture Notes in Computer Science*, No. 2790 (Springer-Verlag) 2003, pp. 421-426,

[19] Aloisio G., Cafaro A., Epicoco I., Fiore S., Lezzi D., Mirto M. and Mocavero S., iGrid, a Novel Grid Information Service, submitted to the first European Grid Conference (EGC 2005).

[20] Wolski R., Spring N. and Hayes J., *J. Future Generation Computing Systems*, **15** (1999) 757.

[21] The Southern Partnership for Advanced Computational Infrastructures Project. `http://www.spaci.it`.

[22] Massonnet D., *Scientific American*, **276** (1997) 46.

[23] Elachi C., *Spaceborne Radar Remote Sensing: Applications and Techniques* (IEEE Press, New York, NY) 1988.

[24] WS - Resource Framework (WSRF). `http://www.globus.org/wsrf/`.