4

Constructing geo-information sharing GRID architecture

Qiang Liu[1] and Boyan Cheng[1,2]
[1]*Institute of Geo-Spatial Information Science and Technology*
University of Electronic Science and Technology of China
China
[2]*No.95007, Guangzhou, Guangdong*
China

1. Introduction

Along with the development of Internet, Geo-information Sharing and Open GIS are of increasing importance for GIS application fields. Spatial Information Grid (SIG) is the fundamental application of Grid technology in spatial information application service domain. This chapter presents a pilot platform for Resource and Environment Geo-information Sharing for Southwestern China based on Web Services, .NET, OGC, Web GIS, SIG, and Mobile Agent is constructed. The architecture in the pilot platform consists of 3 tiers: application layer, service layer and resource layer. Via the pilot platform, distributed heterogeneous geo-information, software and hardware resource from four provinces and one municipality in Southwestern China is integrated.

Geospatial data is the major type of data that human beings have collected. Geospatial data and information are significantly different from those in other disciplines. How to effectively, wisely, and easily use the geospatial data is the key information technology issue that we have to solve.

Along with the development of Internet, Geo-information Sharing and Open GIS are of increasing importance. Grid technology is developed for general sharing of computational resources and not aware of the specialty of geospatial data. Spatial Information Grid (SIG) is the fundamental application of Grid technology in spatial information application service domain. This paper presents a pilot platform for Resource and Environment Geo-information Sharing Architecture for the Southwestern China based on Web Services, Open GIS, Spatial Information Grid and OGSI.Net.

1.1. Open Geographical Information Systems

In (Panagiotis A. Vretanos. 2005), Open GIS Consortium (OGC) thinks that Interoperability is the "capability to communicate, execute programs, or transfer data among various functional units in a manner that requires the user to have little or no knowledge of the unique characteristics of those units." There are many methods of information communications, such as CGI (Common Gateway Interface), Server API (Application Programming Interface), Plug-in GIS, Java Applet, ActiveX and COM (Component Object Model).

The distributed geographic information sharing technology includes three models, which are Microsoft's DCOM (Distributed Component Object Model)/ActiveX, Sun's Java/RMI and OMG's (object management group) CORBA (common object request broker architecture). The ActiveX/DCOM model only runs on the Microsoft Windows platform, and Java/RMI technology localizes on the Java language. In despite of the independency of languages and Operate Systems, CORBA adapts to construct the distributed systems of compact coupling.

In a sense, Web Services extend the distributed Objects/Component technology. By contrast with the former Interoperable methods, Web Services accomplish the accesses to objects in the loose coupling circumstances. Web components can implement and publish in any computer which provides corresponding functions and services, and also integrate the heterogeneous applications. Therefore, the technology of Web Services adapts to share the Geo-information of heterogeneous GISs.

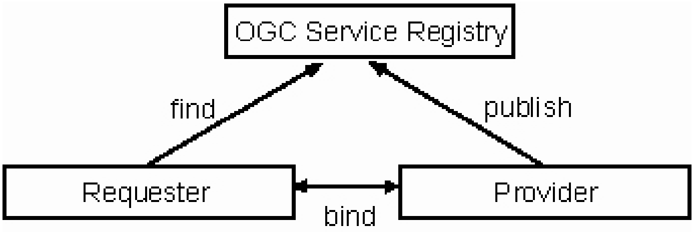


Fig. 1. Three roles in Web Service

Web Services, which are the product of the Internet, publish the standard interfaces of functions or services. The protocols such as Simple Object Access Protocol (SOAP), Web Services Description Language (WSDL) and universal Description, Discovery and Identification (UDDI) provide the architecture for simple, yet effective GIS interoperability independent of concrete platforms and softwares. UDDI's aim is to develop a standard for an online registry, and to enable the publishing and dynamic discovery of Web services offered by businesses. Web Service Architecture includes three roles (as illustrated in Fig. 1.): Service Provider, Service Registry (or Service Broker), and Service Requester. Service Provider is the provider of the web service. The provider implements the service, and makes it available on the Internet. Service Requester is any consumer of the web service. The requestor utilizes an existing web service by opening a network connection and sending an XML request. Service Registry is a logically centralized directory of services. The registry provides a central place where developers can publish new services or find existing ones. It therefore serves as a centralized clearinghouse for companies and their services. The roles communicate through three behaviors: Publish, Find, and Bind.

Provider entity may descript a service with WDSL, and then may register the service in Registration Center (such as OGC Service Registry). Request may find the description of a service in Registration Center, and then bind with the Provider that publishes the service. Request may communicate with Provider each other, and may invoke the service.

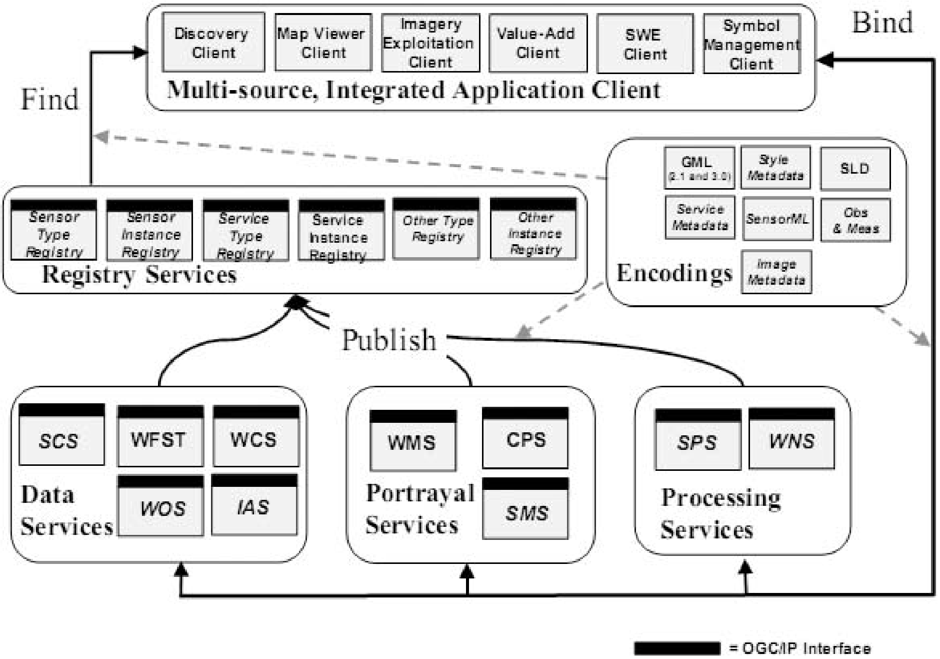


Fig. 2. Architecture of GIS based on Web Service

In (Joshua Lieberman. 2003), the OpenGIS Services Framework (as illustrated in Fig. 2.) provides the common set of interfaces that spans these functional parts of the enterprise and provides enterprise-wide interoperability.

The elements of the architecture are listed as follows:

Client Services - the client-side components of client applications that interact with users, and on the server-side interact with Server-side Client Applications, Application Servers and Data Servers.

Registry Services – provides a common mechanism to classify, register, describe, search, maintain and access information about network resources (data and services). Registry Services include Web Registry Service (WRS).

Processing-Workflow Services – the foundational application-building-block services that operate on geospatial data and metadata, providing value-add service. Processing-Workflow Services include Sensor Planning Service (SPS) and Web Notification Service (WNS).

Portrayal Services – Portrayal Services provide specialized capabilities supporting visualization of geospatial information. Portrayal Services are components that, given one or more inputs, produce rendered outputs such as cartographically portrayed maps, perspective views of terrain, annotated images, views of dynamically changing features in space and time, etc.). Portrayal Services include Web Map Service (WMS), Coverage Portrayal Service (CPS) and Style Management Service (SMS).

Data Services – The foundational service building blocks that serve data, specifically geospatial data. Data Services include Web Object Service (WOS), Web Feature Service

(WFS), Sensor Collection Service (SCS), Image Archive Service (IAS) and Web Coverage Service (WCS).

The OGC (Open GIS Consortium) constitutes three principal interoperable specifications to access Web Services, which are Web Map Server (WMS) as well as the Web Coverage Server (WCS) and the Web Feature Server (WFS). A Web Map Service produces maps of geo-referenced data. This WMS specification (Praveen Ummadi) defines three WMS operations: GetCapabilities (required) returns service-level metadata, which is a machine-readable (and human-readable) description of the WMS service's information content and acceptable request parameters; GetMap (required) returns a map image whose geospatial and dimensional parameters are well-defined; GetFeatureInfo (optional) returns information about particular features shown on a map. A "map" defined as a visual representation of geo-data is not the data itself. The WFS operations support INSERT, UPDATE, DELETE, QUERY and DISCOVERY operations on geographic features using HTTP as the distributed computing platform. In (Boucelma et al, 2002), this WFS specification (Boucelma et al, 2002) defines two kinds of WFS operations: Basic WFS and Transaction WFS. A basic WFS would implement three WFS operations: GetCapabilities (required) describe the capabilities of a web feature service such as which feature types it can service and what operation are supported on each feature type; DescribeFeatureType (required), upon request, describes the structure of any feature type a web feature service can service; GetFeature (required) services a request to retrieve feature instances, when the client specifies which feature properties to fetch and constrains the query spatially and non-spatially. The Basic WFS is considered a READ-ONLY web feature service. A transaction WFS supports all the operations of a Basic WFS and in addition it implement the Transaction operation. A transaction request is composed of operations that modify features; that is CREATE, UPDATE, and DELETE operations on geographic features. Optionally, a transaction WFS could implement the LockFeature operation that process a lock request on one or more instances of a feature type for the duration of a transaction. This ensures that serial transactions are supported.

At present, many GIS applications are independent of each other, and are not interoperable. ArcIMS, ESRI's Web mapping server technology has its own XML protocol, ArcXML utilized to communicate between ArcIMS client and ArcIMS server. ESRI's MapObjectsIMS, which provides clients with images of geo-data, is a tool to develop Web GIS. AutoDesk's Mapguide deals with requests from Mapguide Viewer, and returns the geo-data. Intergraph's GeoMedia transmits ActiveCGM to clients and makes vector maps. These GIS softwares receive respective requests from their own clients. ArcIMS and MapObjectsIMS servers return images of geo-data, Mapguide ans GeoMedia servers provide vector maps. It is difficult to interoperate the applications and the geo-data of these GISs.

1.2. Spatial Information Grid

In (S. Tuecke et al., 2003), (Ian Foster & Carl Kesselman, 1999) and (Ian Foster et al, 2002), the term "Grid" was coined in the mid-1990s to denote a proposed distributed computing infrastructure for advanced science and engineering. "Grid" computing has emerged as an important new field, distinguished from conventional distributed computing by its focus on large-scale resource sharing, innovative applications. The concept of the "SIG" (spatial information grid) which was put forward based on the grid computing, is the fundamental plan and useful tool to construct the Geo-information sharing. SIG is an intelligent information platform which is built based on the current spatial information infrastructure and spatial information network protocol specification and from which the user can obtain the common spatial information services conveniently. The procession of the spatial information in this platform is distributing collaborated and intelligent in this platform, and user can visit all spatial information via the common protocol and do not need to know the information he needed from million website from internet.

In (I. Foster et al, 2005), Global Grid Forum think that grid systems and applications aim to integrate, virtualize, and manage resources and services within distributed, heterogeneous, dynamic "virtual organizations". The realization of this goal requires the disintegration of the numerous barriers that normally separate different computing systems within and across organizations, so that computers, application services, data, and other resources can be accessed as and when required, regardless of physical location. The Open Grid Services Architecture (OGSA) is a service-oriented architecture. It is intended to facilitate the seamless use and management of distributed, heterogeneous resources. In this architecture, the terms "distributed," "heterogeneous" and "resources" are used in their broad sense. For example: "distributed" could refer to a spectrum from geographically-contiguous resources linked to each other by some connection fabric to global, multi-domain, loosely- and intermittently-connected resources. "Resources" refers to any artifact, entity or knowledge required to complete an operation in or on the system.

Building on both Grid and Web Services technologies, the Open Grid Services Infrastructure (OGSI) defines mechanisms for creating, managing, and exchanging information among entities called Grid services. In a word, a Grid Service is a Web Service that conforms to a set of conventions (interfaces and behaviors) that define how a client interacts with a Grid Service.

Web Services address the discovery and invocation of persistent services, and offer a powerful application-to-application integration framework. Web Services miss state management, global service naming, reference resolution and common behaviors and semantics defined for services by some interface description language. Web Services are stateless. In contrast, Grid Services are stateful Web Services. Grid Services are the glue that transforms a collection of distributed, independent resources into a coherent computation, storage, and collaboration fabric. They provide uniform mechanisms for user authentication and authorization, accounting, resource access, data transfer, system monitoring, and resource management. Grid Services make it possible for users, applications, and tools to discover and use disparate resources in coordinated ways.

In (Glenn Wasson et al, 2004), the basic design of OGSI.NET is to have a container entity that "holds" all the service instances running on a host. As illustrated in Fig. 3, the container process consists of a collection of Application Domains (or AppDomains), Microsoft's mechanism for intra-process memory protection. Each service instance executes in its own AppDomain and there is one additional domain for the container's logic (some dispatching and message processing functionality).

A client makes a request on the OGSI.NET architecture by sending a message to the IIS web server. In order to support arbitrary names for grid services, OGSI.NET uses an ISAPI filter to intercept requests at an early stage in the IIS request chain. This filter re-writes the request so that IIS will dispatch it to OGSI.NET's ASP.NET HttpHandler. This HttpHandler dispatches the request to the OGSI.NET container. The container process has a thread pool and each IIS request causes one of the container process' threads to execute the dispatcher.

The dispatcher determines which service instance should get the request and transfers execution of that thread to an object in the appropriate AppDomain.

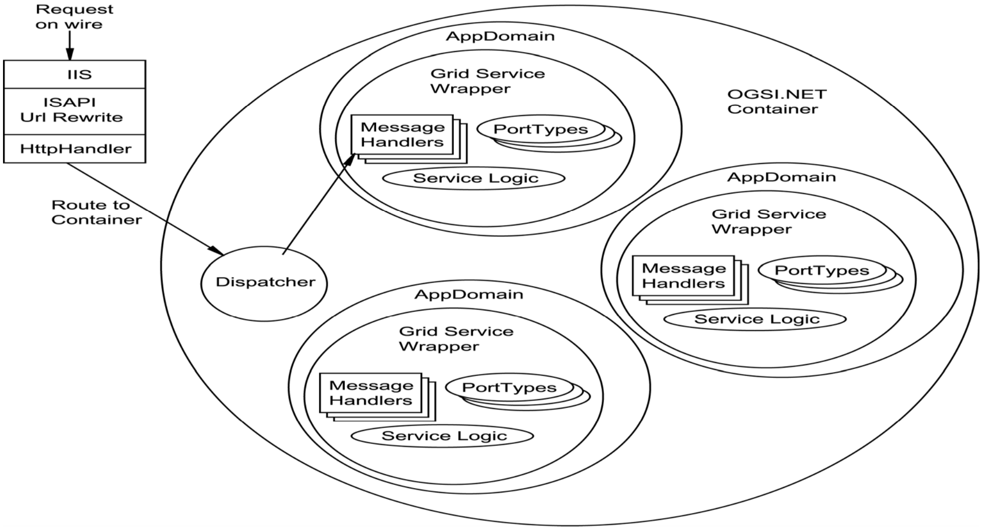


Fig. 3. OGSI Container on the .Net Platform

1.3. Mobile Agent Technology

In (D. Chess et al, 1994), mobile agent is considered as an enabling technology for automated, flexible and customized service provision in a highly distributed way as network nodes become active and take part in the computation of applications and provision of customized services. Besides the clear separation among the key functionality and the aspects of deployment on the functional side, Mobile Agent can offer potential technical advantages. Among them are reduced communication cost, reduced bandwidth usage, the possibility of using remote interfaces and the support for off-line computation. Mobile agents enable both temporal and spatial distribution of the service logic, which makes another technical advantage, namely scalability, while at the same time bottlenecks of centralized approaches such as reduced network availability and malfunctioning are avoided (A. Puliafito et al, 2001). With the capabilities mentioned above, Mobile agents can provide a suitable framework for spatial data resource and service discovery in grid platforms, and can support optimal access and interaction through heterogeneous terminals, differing in terms of memory capacity, computational resources, display characteristics, allowed connection mode, etc (A. L. Corte et al, 1999).

In (Jian Ma et al, 2006), the architecture with the convergency of Mobile Agent and Spatial Data Grid is constructed. A middleware layer allowing users to access distributed resources in a transparent, secure and effective way is proposed. The Grid middleware is integrated with functions to make data mining and data transferring easier. And the mobile agent that manages the user and the issues related with the heterogeneity of the devices is applied.

The agents provided by the system are classed into two types: user agent, carrying users' requests for spatial data; information agent, representing information of distributed spatial data resources and providing spatial data services for other agents. Otherwise, Agency Agent is proposed to settle distributed transparency of system. Agents could communicate with Agency Agent and obtain other agents' information (attribute, location, and resources, etc), and then move to the resource node located and complete corresponding work.

The Mobile Agent system consists of a set of agent subsystems and the agency agent system whose task is to manage the information of all agent subsystems. The agent subsystem which includes User Agent, Native Query Agent and Collaboration Query Agent is interactive and distributed. The system architecture is shown in Fig.4.

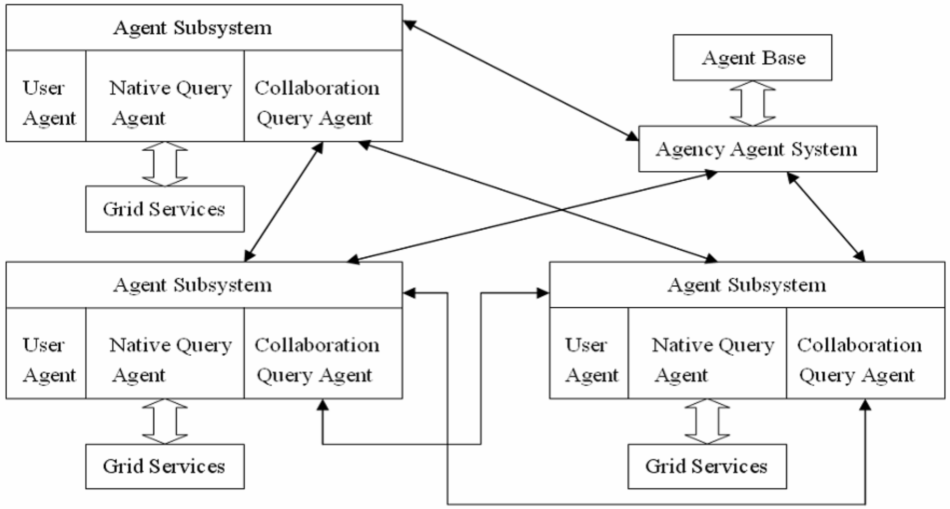


Fig. 4. the Geo-information sharing Grid architecture based on Mobile Agent

Agency Agent (AA), playing the most important role in the whole system, is the key technology to solve the requirement of distributed transparency. The task of Agency Agent is to collect, manage, query all agents' information, and at one time act as a dependable secure authentication center to ensure communication among those agent subsystems. The information Agency Agent needs is provided by information base of agents.

User Agent (UA) has the main task of providing the user with the services he wishes. User Agent which has friendly user interface can accept spatial data requests from users and return them results. User Agent is able to transform users' requests into commands which could be identified by other agents, and search correlative information agent automatically. The process between putting forward requests and returning results is transparent for users.

Native Query Agent (NQA) accepts request from User Agent, and communicates with native spatial data services which are built up in the grid environment to obtain information needed. Native Query Agent is founded by User Agent and runs in background. Native Query Agent is also transparent for users.

Collaboration Query Agent is a mobile agent. It accepts collaboration information query requests from User Agent, gets the address of target agent subsystem from Agency Agent, then moves to target host computer and completes the information query task.

The whole process for information query is described as follows:

1) A request for spatial data is sent to User Agent via web explorer.
2) A request for native information query is sent to Native Query Agent by User Agent.
3) When the native information query is accomplished, the collaboration information query is provided. First, Collaboration Query Agent asks Agency Agent for other agent subsystems' profile information.
4) When gets other agent subsystems' context information, Collaboration Query Agent dispatches a mobile agent which carries corresponding request to the spatial information node located, then the mobile agent asks for native information query in the target agent subsystem's context and returns the result.

Java is adopted in the whole system's implementation to meet platform-independence. Grid environment is built up with Globus Toolkit 4, which is based on Java. Agents' mobility and interoperability is met by Aglets which is based on Java. Dynamic web page and function of User Agent is implemented by Servlet which is based on Java. The communication among agents is actualized by Aglets' message system which is also based on Java.

2. Framework of the resource and environment Geo-information sharing architecture based on Web Services

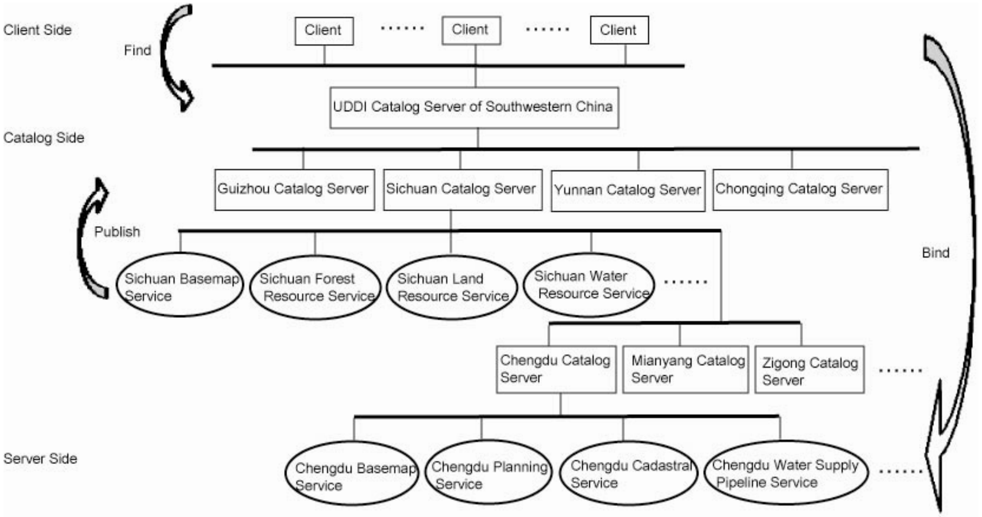


Fig. 5. The resource and environment Geo-information sharing architecture for the Southwestern China

Web service is a stateless service. The Resource and Environment Geo-information Sharing Architecture for the Southwestern China presented in (LIU Qiang & CHENG Boyan, 2006) is based on Web service. It integrates resource and environment geo-information from four provinces and one municipality in the Southwestern China. The framework is illustrated in Fig. 5.

This architecture in the pilot platform consists of 3 tiers (as illustrated in Fig. 4): Client side, Catalog side and Server side. Catalog side is a multi-level tree structure. The top node is a

UDDI Catalog Server of Southwestern China, which owns several children nodes, Guizhou Catalog Server, Sichuan Catalog Server, Yunnan Catalog Server and Chongqing Catalog Server. These children nodes also own several their own children nodes, respectively. For example, Sichuan Catalog Server's children nodes are Chengdu Catalog Server, Mianyang Catalog Server, and Zigong Catalog Server, etc. All Services in Southwestern China are separated into several cases corresponding to UDDI Catalog Servers. For instance, Provincial Services such as Sichuan Basemap Service, Sichuan Forest Resource Service, Sichuan Land Resource Service, and Sichuan Water Resource Service as well as the children Catalog Servers are registered into Sichuan Catalog Server. Municipal Services such as Chengdu Basemap Service, Chengdu Planning Service, Chengdu Cadastral Service and Chengdu Water Supply Pipeline Service as well as the children Catalog Servers are registered into Chengdu Catalog Server. Thus, users can access all services via the UDDI catalog servers tree conveniently.

2.1. System Structure Platform Architecture

The stateless architecture in the pilot platform consists of 3 tiers (as illustrated in Fig. 6): client side, catalog side and server side.

The server side as service provider publishes and registers services to the catalog side. It includes multiple web sites which provide services of geo-data (base map database, forest, land-use, mineral, disaster and water resources, etc.) and mapping functions (Qiang Liu et al, 2005).

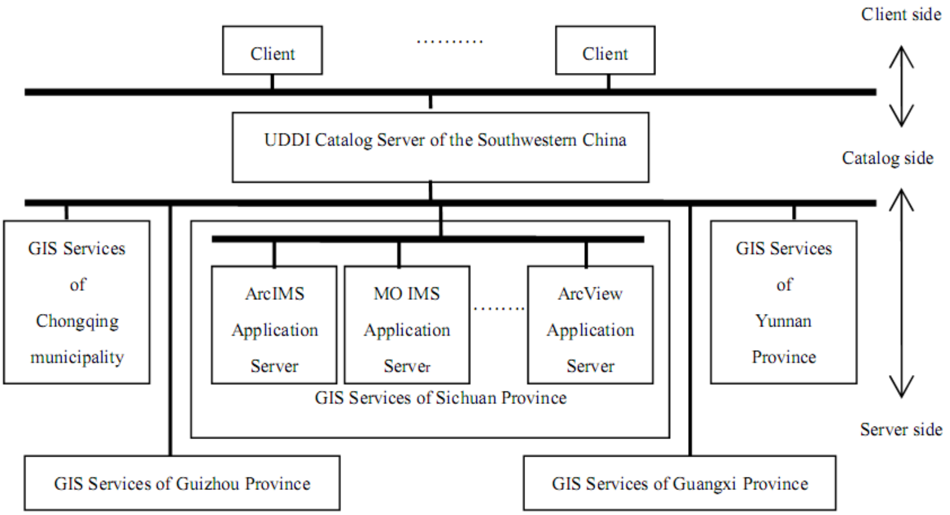

Client side
Client
....
Client
UDDI Catalog Server of the Southwestern China
Catalog side
GIS Services of Chongqing municipality
ArcIMS Application Server
MO IMS Application Server
....
ArcView Application Server
GIS Services of Sichuan Province
GIS Services of Yunnan Province
Server side
GIS Services of Guizhou Province
GIS Services of Guangxi Province

Fig. 6. The 3tiers architecture in the pilot platform

As a service requester, the client side makes the OGC WMS-compliant command to inquire geo-data and services. It finds the service description in the catalog side, then binds the service provider and invokes the service. At last, the client side displays the result and the image. The client side communicates with the server side via SOAP.

2.2. System Function

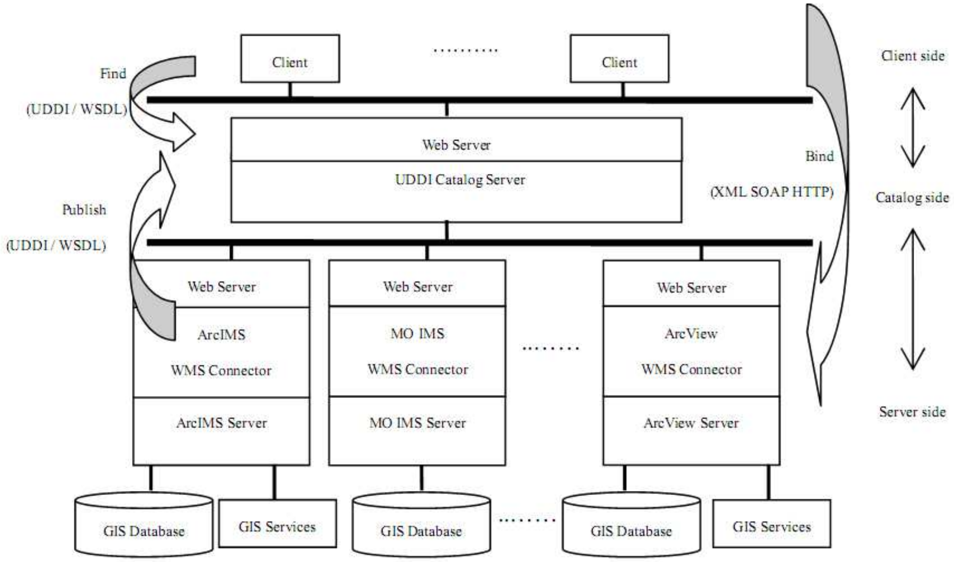


Fig. 7. the Geo-information Sharing Architecture Based on WMS

In the Resource and Environment Geo-information Sharing Architecture based on WMS (as illustrated in Fig. 7), the server side that includes WMS connectors publishes and registers services to the catalog side. Firstly, the server side describes services in WSDL, organizes metadata, and publishes the documents to the catalog side via UDDI. In the client side, a user browses uniform graphics interface and chooses service scopes such as districts and layers. The client side makes a WMS-compliant search request (or a series of searches), and sends it to the catalog side. The request is first handled by the Web server (such as Microsoft IIS), and then submitted to the catalog server in the catalog side. According to the request, the catalog server searches from the index tree of service metadata, returns the description of the specific services. According to the description, the client side makes the WMS-compliant image request, and then sends the image request to the server side. The web server of the server side parses the request, and then invokes the service provided by GIS server through the WMS connector. The service invoked by the web server handles the geo-data and produces an image. Then the image is sent to the web server through the WMS connector, transferred to the client side in succession.

In the Resource and Environment Geo-information Sharing Architecture based on WFS (as illustrated in Fig. 8), the server side that includes WFS connectors publishes and registers services to the catalog side. The client side makes a WFS-compliant search request (or a series of searches), and sends it to the catalog side. According to the description returned from the catalog side, the client side makes the WFS-compliant geographic features request, and then sends the geographic features request to the server side. The web server of the server side parses the request, and then invokes the service provided by GIS server through the WFS connector. The service invoked by the web server handles the geo-data and produces a shape file and a feature properties file include geographic features requested.

Then the files are sent to the web server through the WFS connector, transferred to the client side in succession, and then displayed the map of the requested geographic features (as illustrated in Figure 4)。

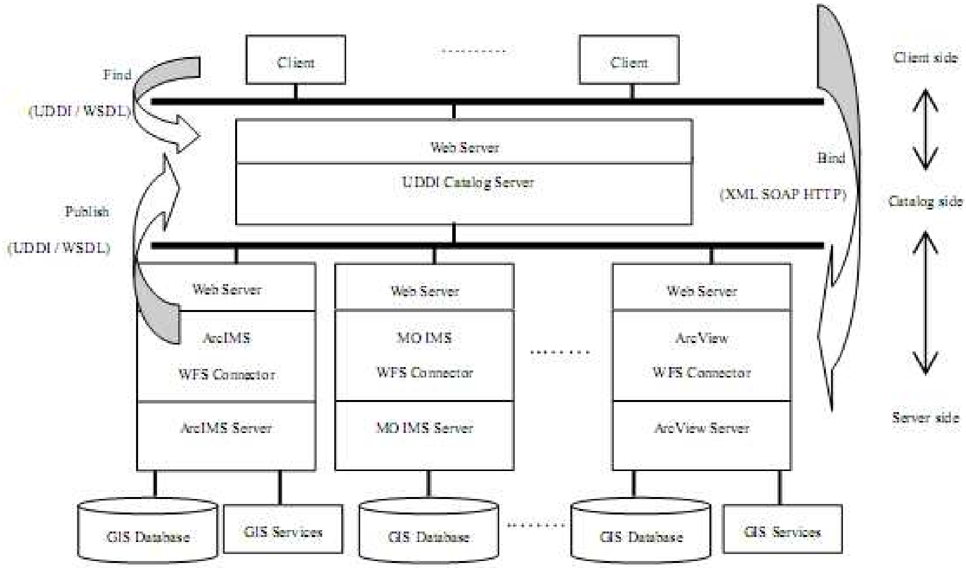


Fig. 8. the Geo-information Sharing Architecture Based on WFS

2.3. Key Technologies

The service metadata in the sharing platform is published in the catalog side. Along with the increase of service metadata, it is important to design a method to organize and inquire the metadata. The service metadata is stored in a structure of an index tree. A node of the index tree stores services that handle geo-data in the same geographical coordinate scope. According to the spatial scope of requests, the catalog server recursively searches for the corresponding service from the root node to leaf nodes of the metadata index tree.

Making WMS connectors is one key of constructing the sharing platform. For each type of Web GIS software used in the architecture, a respective WMS connector is needed. In the circumstance of Microsoft .NET, ISAPI program is a DLL file that separately runs in a server. In this platform, we have built three WMS connectors: ArcIMS WMS connector, ArcView WMS connector and MO-IMS WMS connector. The ArcIMS WMS connector developed as ISAPI is used to transmit WMS-compliant requests to the ArcIMS server side. The ArcIMS WMS connector receives the WMS-compliant requests from web server, as followed.

http://serverIP/Scripts/GetMap.dll?SERVICENAME=servicename&REQUEST=GetMap&LAYERS=layerlist&STYLES=stylelist&SRS=namespaceidentifier&BBOX=minx,miny,maxx,maxy&WIDTH=outputwidth&HEIGHT=outputheight&FORMAT=outputformat&TRANSPARENT=0&BGCOLOR=0xFFFFFF&EXCEPTIONS=SE_XML&&VERSION=1.1.0

Then, the ArcIMS WMS connector transfers them to the ArcIMS-compliant requests that consist of the requests URL and the ArcXML file. The requests URL is:

http://ArcIMSserverIP/servlet/com.esri.esrimap.Esrimap?
ServiceName=servicename&ClientVersion=4.0

The ArcXML file is:

```
<?xml version='1.0'   encoding='UTF-8' ?>
 <ARCXML version='1.1'>
  <REQUEST>
   <GET_IMAGE show=”layerlist”>
    <PROPERTIES>
     <ENVELOPE   minx=”minx” miny=”miny” maxx=”maxx” maxy=”maxy” />
    </PROPERTIES>
   </GET_IMAGE>
  </REQUEST>
 </ARCXML>
```

At last, the ArcIMS WMS connector submits them to ArcIMS server. With such specific WMS connectors, a united WMS-compliant client interface and a catalog side used to serve for both the WMS-compliant client side and the server side can be built. Then, the Resource and Environment Geo-information Sharing Architecture in the Southwestern China with a 3-tier WMS-compliant Web Service can be implemented.

Making WFS connectors is the other key of constructing the sharing platform. For each type of Web GIS software used in the architecture, a respective WFS connector is needed. In the circumstance of Microsoft .NET, ISAPI program is a DLL file that separately runs in a server. In this platform, we have built three WFS connectors: ArcIMS WFS connector, ArcView WFS connector and MO-IMS WFS connector. The ArcIMS WFS connector developed as ISAPI is used to transmit WFS-compliant requests to the ArcIMS server side. The ArcIMS WFS connector receives the WFS-compliant requests from web server, as followed.

http://serverIP/Scripts/GetFeature.dll?SERVICENAME=servicename&REQUEST=GetFeature&LAYERS=layerlist&STYLES=stylelist&SRS=namespaceidentifier&BBOX=minx,miny,maxx,maxy&WIDTH=outputwidth&HEIGHT=outputheight&FORMAT=outputformat&TRANSPARENT=0&BGCOLOR=0xFFFFFF&EXCEPTIONS=SE_XML&&VERSION=1.1.0

Then, the ArcIMS WFS connector transfers them to the ArcIMS-compliant requests that consist of the requests URL and the ArcXML file. The requests URL is:

http://ArcIMSserverIP/servlet/com.esri.esrimap.Esrimap?ClientVersion=3.1&ServiceName=servicename&CustomService=Extract

The ArcXML file is:

```
<?xml version='1.0'   encoding='UTF-8' ?>
 <ARCXML version='1.1'>
  <REQUEST>
   <GET_EXTRACT>
    <PROPERTIES>
     <ENVELOPE minx=”minx” miny=”miny” maxx=”maxx”     maxy=”maxy” />
    </PROPERTIES>
   </GET_EXTRACT>
  </REQUEST>
 </ARCXML>
```

At last, the ArcIMS WFS connector submits them to ArcIMS server. With such specific WFS connectors, a united WFS-compliant client interface and a catalog side used to serve for both the WFS-compliant client side and the server side can be built. Then, the Resource and Environment Geo-information Sharing Architecture in the Southwestern China with a 3-tier WFS-compliant Web Service can be implemented.

3. Framework of the resource and environment Geo-information sharing architecture based on Spatial Information Grid

The Resource and Environment Geo-information Sharing Architecture for the Southwestern China based on GRID presented in this section integrates distributed heterogeneous geo-information, software and hardware resource from four provinces and one municipality in the Southwestern China (Qiang Liu & Boyan Cheng, 2009).

3.1. System platform architecture

The architecture in the pilot platform consists of 3 tiers (as illustrated in Fig. 9): application layer, service layer and resource layer.

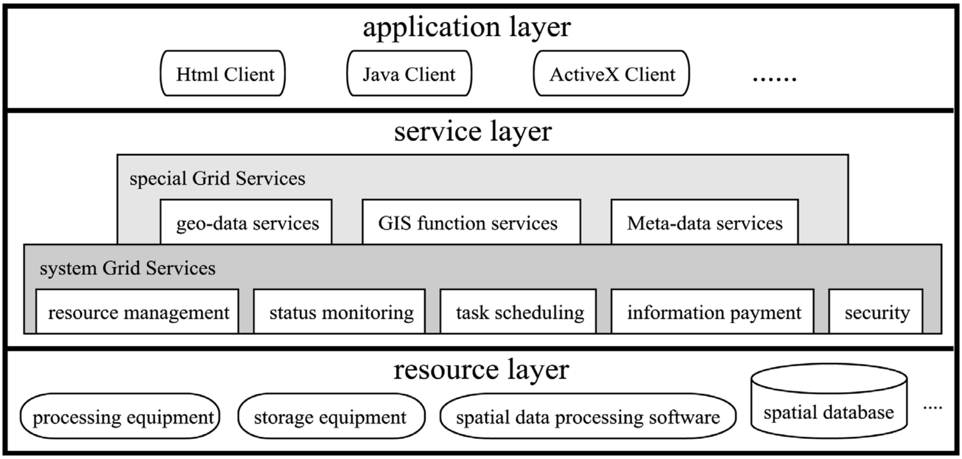


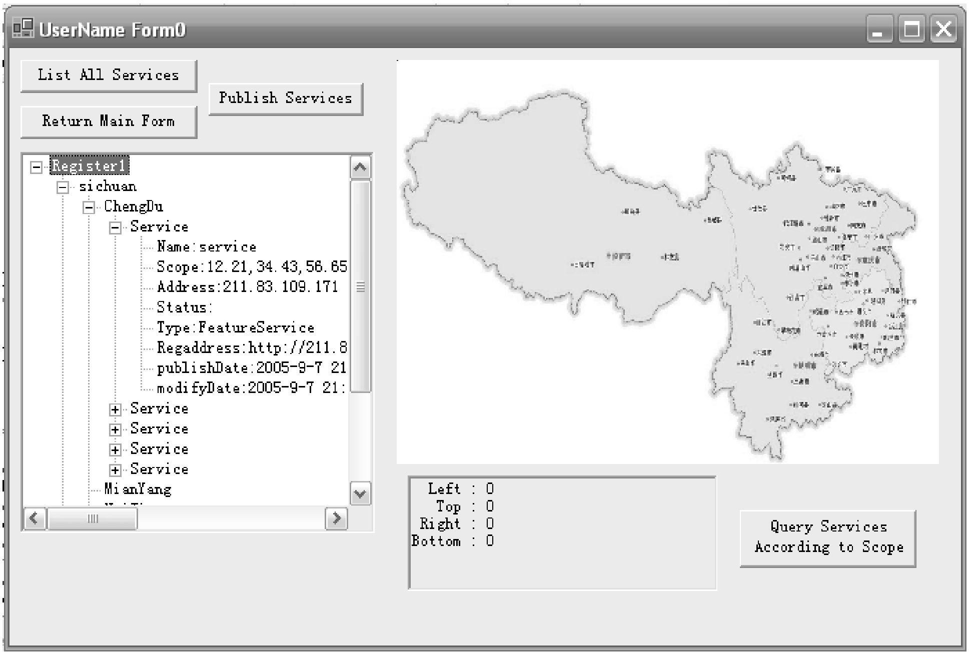
Fig. 9. Resource and Environment Geo-information Sharing Architecture

The resource layer includes storage resource-multiple spatial databases which provide geodata (base map database, special map database, etc.), various GIS softwares (ArcIMS, ArcMap, etc.), and disposal equipment (such as computers).

The resources are connected via facilities of Internet or wireless communication. The service layer, which builds on the resource layer, provides a management platform of integrative spatial information, and comprises system services and special services.

The application layer can request Grid Services of geodata or functions and browse maps via uniform user interface.

The service layer is the core layer of the Resource and Environment Geo-information Sharing Architecture based on GRID. The system Grid Services in the service layer, manage and maintenance the sharing platform. They are composed of resource management service, security service, task scheduling service, monitoring service and payment service. The special Grid Services include geodata services and GIS processing services.



```
Register1
  sichuan
    ChengDu
      Service
        Name:service
        Scope:12.21,34.43,56.65
        Address:211.83.109.171
        Status:
        Type:FeatureService
        Regaddress:http://211.8
        publishDate:2005-9-7 21
        modifyDate:2005-9-7 21:
      Service
      Service
      Service
      Service
    MianYang
```

```
  Left : 0
   Top : 0
 Right : 0
Bottom : 0
```

Fig. 10. The interface of the register service

3.2. The system Grid Services.

The resource management service is in charge of registering spatial information services and spatial data services to the registry center, and managing the services. The registry center is divided into three levels that constitute a structure of an index tree. The structure of an index tree facilitates to register, discover, update, and dispose the register information. The root node is the main register center, which is the first level register center. Resources are registered in the leaf nodes, and the junior register centers are registered in other nodes. The user interface of the register service is showed in Fig. 10. A user can list all services registered in the register center, and then select the service.

The monitoring service monitors the status of Grid nodes and GIS processing services. If the status of the registered service is changed, the monitoring service notifies the registry service to update the status. The method can ensure that the services in registry centers are exact and real-time.

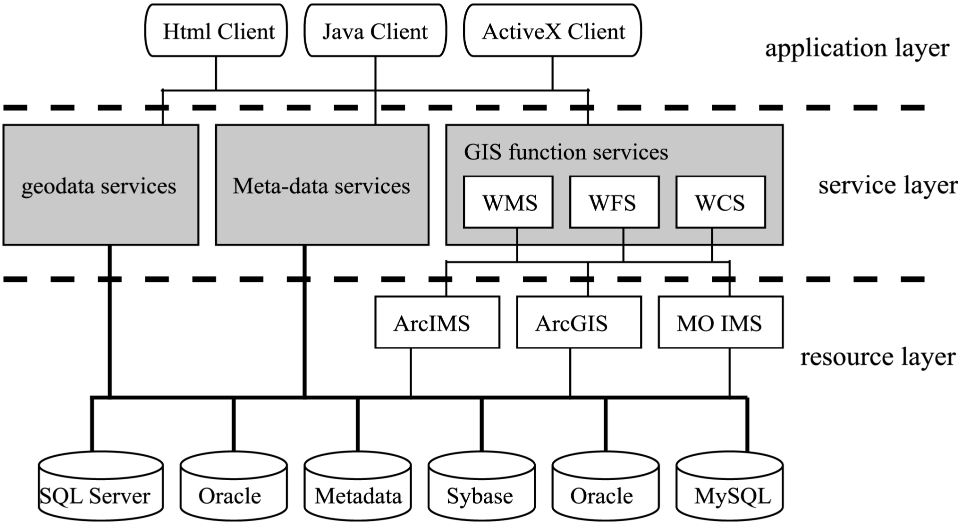


Fig. 11. The special Grid Services architecture

The security service is in charge of the security of the sharing platform. It provides two aspects security: one is access control service, which ensures that unlawful users can not access the resources in the sharing platform; the other is communication security service, which encrypts and decrypts transmitted data and implements digital signature.

3.3. The special Grid services.

The special Grid Services include geodata services, meta-data services and GIS processing services (as illustrated in Fig. 11). The geodata services access heterogeneous distributed database, and implement the geodata and meta-data sharing. The GIS function services invoke OGC-compliant services, such as Web Map Service, Web Feature Service, and Web Coverage Service, and share the Internet Map Services that run in distributed grid node. The meta-data services, as an assistance of geodata services, publish, find and manage meta-data.

4. Framework of the resource and environment Geo-information sharing Grid architecture based on Mobile Agent

According to the system principle, an application flow whose purpose is to query spatial data information in grid environment is put forth, as illustrated in Fig. 12.
1) A spatial data request is sent by the user who visits any site in the system via web explorer.
2) Web explorer and Aglets system are equipped on the web server. When a request for spatial data is accepted by a web explorer, a User Agent's agent is started by servlet engine to judge whether it is a native work. The reason that a User Agent is not started directly is that User Agent doesn't allow the outside to access and read its information directly but allows that via an agent which offers corresponding interface.

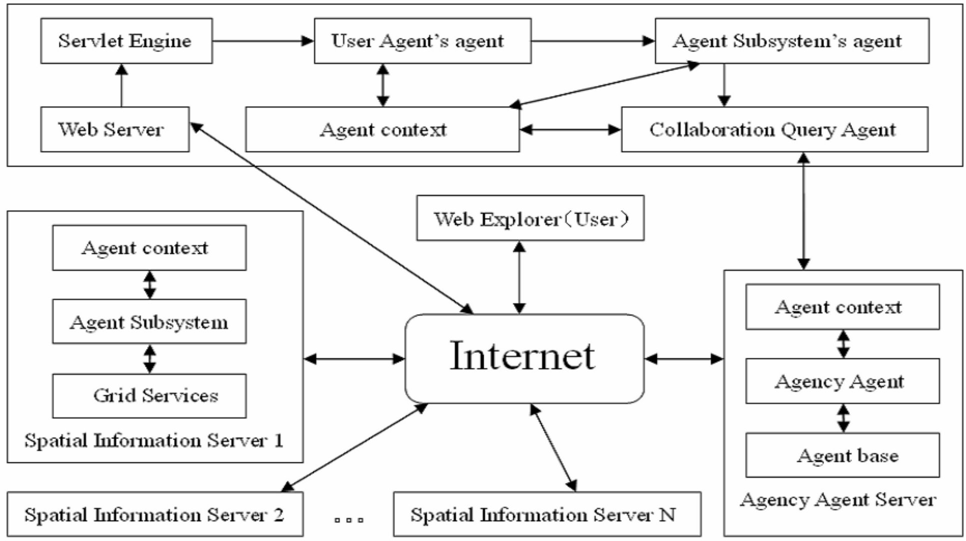


Fig. 12. The framework of Geo-information sharing Grid based on Mobile Agent

3) If it is a native task, a native agent subsystem's agent is started directly and dispatched to native spatial information server. When arrived, the native agent subsystem's agent sends a service request to spatial data service which is built up in Globus Toolkits 4 and returns the result to web server.
4) If it isn't a native task, a Collaboration Query Agent is created and dispatched to Agency Agent server, and then asks for querying spatial information servers. When obtains target spatial information server's address and port, web server starts proper agent subsystem's agent and dispatches it to target spatial information server. The agent communicates with spatial data service which is built up in Globus Toolkits 4 and returns the result to web server.
5) While getting all the information needed, the web server returns it to user via web explorer.

5. Conclusions

This Geo-information sharing platform provides integrated spatial information and applications for users with the technology of Spatial Information Grid, the Grid platform of OGSI.NET, and mobile agent. In this platform, a feasible method for spatial data sharing and interoperability in grid environment is provided. It makes data accessing easier and shields spatial data's heterogeneity. Users can access spatial information resource through uniform interface.

The interoperation of heterogeneous GIS is implemented in the Resource and Environment Geo-information Sharing Architecture for the Southwestern China. Via uniform user interface, web users can take advantage of geo-data and function provided by various Web GISs. However, there are some problems that ought to be further solved, such as the security of the access to spatial databases, the management of the Geo-information service lifecycle, and etc.

6. Acknowledgments

This research was funded partly by the National Basic Research Program of China (also called the 973 program, contract 2007CB714400), by National Key Technology Support Program (contract 2006BAJ09B09), and by Open Research Fund Program (contract GCWD200706) of Key Laboratory of Digital Mapping and Land Information Application Engineering, State Bureau of Surveying and Mapping ″ . The authors would also like to thank everyone who has supported this effort through his thoughtful discussions of issues raised in this paper.

7. References

A. L. Corte, A. Puliafito, and O. Tomarchio. (1999). An agent-based framework for mobile users. in ERSADS'99, Madeira, Portugal,1999.

A. Puliafito, S. Riccobene, and M. Scarpa. (2001). Which paradigm should I use? An analytical comparison of the client-server, remote evaluation and mobile agent paradigms. Lecture Notes in Computer Science, vol.2150, 2001.

Boucelma, O., Esid, M., and Lacroiz, Z. (2002). A WFS-based Mediation System for GIS Interoperability. Tenth ACM International Symposium on Advances in GIS, 2002, pp. 23-28.

D. Chess, C. Harrison, and A. Kershenbaum.(1994). Mobile Agents: Are They a Good Idea?, Yorktown Herghts, New York, Tech. Rep. RC 19887, (December 21, 1994-Declassified March 16, 1995), 1994

Glenn Wasson, Norm Beekwilder, Mark Morgan, etal. (2004). OGSI.NET: OGSI-compliance on the .NET Framework. http://www.cs.virginia.edu/~gsw2c/OGSIdotNet/ccGrid04.pdf

Ian Foster, Carl Kesselman. (1999). The Grid: Blueprint for a Future Computing Infrastructure, Morgan Kaufmann Publishers, USA, 1999.

Ian Foster, Carl Kesselman, Jeffrey Nick, and Steve Tuecke. (2002). The Physiology of the Grid: An Open Grid Services Architecture for Distributed Systems Integration, January 2002.

I. Foster, H. Kishimoto, A. Savva. (2005). The Open Grid Services Architecture, Version 1.0. http://www.gridforum.org/documents/GWD-I-E/GFD-I.030.pdf

Joshua Lieberman. (2003). OpenGIS Web Services Architecture. Version: 0.3(2003-01-18). http://www.opengeospatial.org/docs/03-025.pdf

LIU Qiang, CHENG Boyan. (2006). Constructing Geo-information Sharing Architecture. GEO-SPATIAL INFORMATION SCIENCE [QUARTERLY], 2006, Vol. 9, ISSUE 3, pp. 206~209

Jian Ma, Qiang Liu, Boyan Cheng, Yuancheng Sun. (2006). A Mobile Agent Based Spatial Data Grid, 26th Anniversary IGARSS 2006 IEEE, pp. 3271~3274

Panagiotis A. Vretanos. (2005). Web Feature Service Implementation Specification. Open Geospatial Consortium, 2005, Inc. OGC 04-094. https://portal.opengeospatial.org/files/?artifact_id=8339

Praveen Ummadi. A white paper on Standards & Interoperability in GIS. http://www.msu.edu/~ummadipr/Standards & Interoperability in GIS.pdf

Qiang Liu, Boyan Cheng. (2009). Constructing Geo-information Sharing GRID Architecture. International Conference on management and Service Science 2009, Vol. 6: 022

Qiang Liu, Boyan Cheng, Xingfa Gu. (2005). Constructing Geo-information Sharing Architecture for the Southwestern China Based on WMS, 25th Anniversary IGARSS 2005 IEEE, Vol 2, pp. 971~974.

S. Tuecke, I. Foster, S. Graham, et al. (2003). Open Grid Services Infrastructure (OGSI). http://xml.coverpages.org/OGSISpecificationV110.pdf

Xue, Y., Wang, J., Sheng, X., & Guo, H. (2003). Building Digital Earth with GRID Computing–The Preliminary Results, Proceedings of Digital Earth 2003, 2003, pp. 804-812

Yuan Xiang-ru, Gong Jian-ya, Lin Hui, Chen Li-li. (1999). Internet GIS Approach to Interoperability of Heterogeneous Geographic Information Processing Environment. Journal of Wuhan Technical University of Surveying and Mapping, Sept.1999, Vol.24 No.3, pp. 194-198

Management and Services
Edited by Mamun Habib

ISBN 978-953-307-118-3
Hard cover, 112 pages
Publisher Sciyo
Published online 06, October, 2010
Published in print edition October, 2010

Management in all business areas and organisational activities are the acts of getting people together to accomplish desired goals and objectives. Service is intangible, therefore, it is not too easy to define the theory application in varieties of service industries. Service Management usually incorporates automated systems along with skilled labour; it also provides service development. Due to enormous demand of service industries and management development, the book under the title "Management and Services" would create a milestone in management arena for all categories of readers including Business Administration, Engineering and Architecture. This book covers educational service development, service-oriented-architecture and case research analysis, including theory application in network security, GRID technology, integrated circuit application. The book is comprised of five chapters and has been divided into two parts. Part A contains chapters on service development in educational institutions and it depicts the application of supply chain management concept in service industries like tertiary educational institutions and multiple ways of web 2.0 applications transforming learning patterns and pathways. To understand the subject in a practical manner, Part B of this book consists of noteworthy case studies and research papers on management and services and represents theory application of Data mining, Fuzzy Cluster, Game theory, GRID Technology, simulation of Operational Amplifier and Current Controlled Conveyor II in network security, architecture, and integrated circuit application.

How to reference

In order to correctly reference this scholarly work, feel free to copy and paste the following:

Qiang Liu and Boyan Cheng (2010). Constructing Geo-Information Sharing GRID Architecture, Management and Services, Mamun Habib (Ed.), ISBN: 978-953-307-118-3, InTech, Available from: http://www.intechopen.com/books/management-and-services/constructing-geo-information-sharing-grid-architecture

INTECH
open science | open minds

InTech Europe
University Campus STeP Ri
Slavka Krautzeka 83/A
51000 Rijeka, Croatia
Phone: +385 (51) 770 447

InTech China
Unit 405, Office Block, Hotel Equatorial Shanghai
No.65, Yan An Road (West), Shanghai, 200040, China
中国上海市延安西路65号上海国际贵都大饭店办公楼405单元
Phone: +86-21-62489820

www.intechopen.com

Fax: +385 (51) 686 166
www.intechopen.com

Fax: +86-21-62489821