

Information Services for Grid/Web Service Oriented Architecture (SOA) Based Geospatial Applications

Mehmet S. Aktas, Galip Aydin, Geoffrey C. Fox, Harshawardhan Gadgil, Marlon Pierce and Ahmet Sayar

Community Grids Lab, Indiana University, Bloomington, Indiana, 47404, USA
e-mail: {maktas, gaydin, gcf, hgadgil, mpierce, asayar}@cs.indiana.edu

Abstract

Geographical Information Systems (GIS) presents data-intensive environment for acquiring, processing and sharing geo-data among interested parties. In order to serve geographical information to users in such environment, Service Oriented Architecture (SOA) principles have gained great importance. In SOA-based systems, Information Services support the discovery and handling of these geospatial services.

Some options for Information Services in SOA-based GIS systems include a) the Open GIS Consortium (OGC) Web Registry Service (WRS) and b) the Universal Description, Discovery, and Integration (UDDI). WRS is an OGC standard to discover/publish service information of geospatial services. It presents a domain-specific registry capability for geospatial information. UDDI is domain-independent standardized method for publishing/discovering information about Web Services. As it is WS-Interoperability (WS-I) compatible, UDDI has the advantage being interoperable with most existing Grid/Web Service standards.

This study presents an approach combining domain-specific registry capabilities of WRS and WS-I compatible UDDI Specifications. We extend UDDI Information Model to support geospatial services. Our approach supports not only quasi-static, stateless metadata, but also more extensive metadata requirements of rich interacting systems. The implementation of our approach is being used to support a GIS workflow system which is a part of NASA Solid Earth Virtual Observatory (SERVO) Grid project.

1. Introduction

Geographical Information Systems (GIS) presents data-intensive environment for acquiring, processing and sharing geospatial-data among interested parties. In order to serve geographical information to users in such environment, Service Oriented Architecture (SOA) principles have gained great importance. To serve online geo-data and geo-processing capabilities, the OpenGIS Consortium (OGC) has introduced specifications for geospatial web services, such as Web Feature Service and Web Map Service. As the number of geospatial services increased rapidly, an immerging need has also appeared for

methodologies to locate desired services that provide access and data mining capabilities to geospatial data. In SOA-based systems, Information Services support the discovery and handling of these geospatial services.

In this paper, we analyze the state of art in Information Services particularly in GIS domain and discuss our solution as an alternative efficient methodology. Briefly, our approach provides following features.

First, in order to be compatible with existing web/grid service standards, our solution to Information Services is WS-Interoperability (WS-I) compatible.

Second, our approach supports the ability to publish and search prescriptive metadata of services. Here, prescriptive metadata is the descriptive and domain-specific information about the functionality of a service. For instance, prescriptive metadata of a geospatial service could be the capabilities file describing the geospatial data presented by that service.

Third, our approach takes into account the descriptive metadata, i.e. quality of service attributes, into discovery process. The geospatial data being provided by a geospatial service may be fitted with client's request, however, this does not necessarily guarantee whether the service is sufficient for the desired quality of service requirements. By matching Quality of Service attributes of service discovery request and service descriptions, client is able to distinguish geospatial services that match to their requirements.

Fourth, our approach supports stateless web service interactions where services are not responsible storing session and/or state information generated due to service interactions. Geospatial services treat each request independently and Information Services are responsible for management of session metadata. Here, session metadata is considered as distributed shared memory of services that correlates the work of participant geospatial services or their consumers within the same activity.

Fifth, in most service registry systems, service descriptions of a registry may be stale because of the volatile behavior of services. Services may come and go and service metadata can change frequently. To avoid out-of-date information in the registries, our approach provides monitoring schemes to keep the service-metadata up-to-date. We implement leasing concept, which is introduced by Jini Specifications [6], where service providers makes an agreement about how long will the service be alive.

In order to provide these features, our contribution can be summarized as follows. In our approach we provide metadata-oriented service discovery. We utilize existing UDDI Specifications [4] and design an extension to UDDI Data Structure and UDDI XML API to be able to associate both prescriptive and descriptive metadata with service entries. We also cover session metadata requirements of services, by integrating such metadata into service registries. We implement extended version of WS-Context Specifications to provide interface for publishing and accessing session metadata.

This paper is organized as follows. First, we discuss the relevant work explaining different approaches for service discovery in GIS domain. Then, we discuss our contribution in GIS compatible Information Services. Next, we describe our architectural design followed by an application case scenario. Finally, we conclude the paper with conclusions and future work.

2. Relevant Work

Open GIS Consortium Inc. (OGC) [1], an international organization providing specifications to integrate geospatial data and geo-processing resources into mainstream computing, leads an effort to provide a) standardized protocols for accessing geospatial information and services and b) standardized service metadata such as “capabilities.xml” file. The OGC introduces a) Catalog Specifications (v.2) which defines discovery and retrieval of metadata that describes geospatial data and geo-processing services and b) Web Registry Service (WRS) Specifications [7] as implementation specification of the OGC Catalog Specifications. WRS Specifications define a standard to discover/publish service information of geospatial services and presents a domain-specific registry capability for geospatial information. The WRS Specification adopts the OGC Registry Information Model which is based on the ebXML registry information model (eBRIM) [5] (<http://www.ebxml.org>). WRS uses eBRIM to support/integrate service entries with metadata. So, the eBRIM provides facilities for metadata management to implement catalogue services for geospatial domain.

An example implementation of WRS based on Metadata Catalog Service (MCS) [23] is done by LAITS group in George Mason University [24]. In their work, OGC Registry Information Model is mapped into Metadata Catalog Service data model.

Our approach utilizes an alternative Registry Information Model; the Universal Description, Discovery, and Integration (UDDI). UDDI is domain-independent standardized method for publishing/discovering information about Web Services. As it is WS-Interoperability (WS-I) compatible, UDDI has the advantage being interoperable with most existing Grid/Web Service standards.

We observe that the adoption of UDDI Specification in various domains such as GIS is slow, since existing UDDI specification has following limitations. First, UDDI introduces keyword-based retrieval mechanism. It does not allow advanced metadata-oriented query capabilities on the registry. Second, UDDI does not take into account the volatile behavior of services and supports only quasi-static service registries. Since Web Services may come and go and information associated with services might be dynamically changing, there may be stale data in registry entries. Third, UDDI does not support extensive metadata requirements of rich interacting systems. For instance, services may require an Information Service to publish and discover session metadata generated by one or more services as a result of their interactions. Fourth, since UDDI is domain-independent, it does not provide domain-specific query capabilities in particular for GIS domain such as spatial queries. There is a need for integration between the OGC and UDDI Service information model. In

order to provide solutions to these limitations, various solutions have been introduced. First, we will discuss OGC approach to UDDI limitations for spatial service discovery.

2.1. OGC use of UDDI Registries

OGC has introduced design principles, requirements, spatial discovery methodologies for discovery of OGC services through UDDI interface [11]. The proposed methodologies have been implemented by various organizations such as Sycline and Galdos Inc. The Syncline experiment focuses on implementing a UDDI discovery interface on an existing OGC Service Registry so that services registered through OGC Registries can be discovered by UDDI users. The Galdos experiment focuses on turning OGC Service Registry into a UDDI node by utilizing JAXR API to map UDDI inquiry interface to the OGC Registry Information Model. Briefly, these methodologies showed that it is possible to do spatial discovery and content discovery through out-of-box UDDI Specifications. Here, the discovery of services is based on general taxonomic or custom classification criteria. We note that these discovery methodologies extend the UDDI interface, however, they do not introduce an extension to existing UDDI information model.

Existing UDDI approaches by OGC community are designed in particular for geospatial specific usage. Services such as the Web Map and Web Feature service, because they are generic, must provide additional, descriptive metadata, such as Quality of Service attributes, in order to be useful. As the number of geospatial services has increased rapidly, it is still an open problem how to make these geospatial services distinguishable from others. We need to provide extensive metadata-oriented query capabilities in addition to spatial query capabilities. Here, we investigate methodologies for discovering services based on both general and domain-specific search criteria.

2.2. Extensions to UDDI Information Model

There have been some solutions introduced to provide better retrieval mechanism by extending existing UDDI Specifications. UDDIe [8] project introduces the idea of associating metadata and lifetime with UDDI Registry service descriptions where retrieval relies on the matches of attribute name-value pairs between service description and service requests. UDDI-M^T [10] improves the metadata representation from attribute name-value pairs into RDF triples. A similar approach to leverage UDDI Specifications was introduced by METEOR-S [9] project which identifies different semantics when describing a service, such as data, functional, quality of service and executions.

In our design, we also extend UDDI Information Model, by providing an extension where we associate metadata with service descriptions similar to existing solutions. We use name-value pairs to describe characteristics of services. However, apart from the existing methodologies, we provide both general and domain-specific query capabilities. An example for domain-specific query capability could be XPATH/RDQL queries on the auxiliary and domain-specific metadata files stored in the UDDI Registry. Another distinguishing aspect of our design is the support for session metadata. Our design supports not only quasi-static, stateless metadata, but also more extensive metadata requirements of interacting systems.

We explain our methodology in extending UDDI to provide GIS compatible Information Services in the next section in great details.

3. GIS compatible Information Services

Services such as the Web Map and Web Feature service, because they are generic, must provide additional, descriptive metadata in order to be useful. A client may interact with two different Web Feature Services in exactly the same way (the WSDL is the same), however the Web Feature Services may hold different data. One, for example, may contain GPS data for the Western United States while the other has GPS data for Northern Japan. Clients must be able to query information services that encode (in standard formats) all the necessary information, or metadata, that enables the client to connect to the desired service. This is an example of the very general problem of managing information about Web Services. An approach to solve the problem of locating services of interests is the Universal Description, Discovery, and Integration (UDDI) specifications. The UDDI is WS-I compatible and offers users a unified and systematic way to find service providers through a centralized registry of services.

Here we design Information Services that extends existing UDDI Specifications in order to implement geospatial-domain-specific, dynamic and metadata-oriented service registries.

3.1. Supporting/integrating quasi-static, stateless metadata

First, we classify service metadata as session metadata and quasi-static, interaction-independent metadata. Session metadata is the dynamically generated metadata as result of interactions of Web Services. Static and interaction-independent metadata is the metadata describing Web Service characteristics such as its usage cost, availability, bandwidth, computing power, storage capability, etc. These characteristics can be broadly classified as a) prescriptive (functional) and b) descriptive (non-functional). a) The prescriptive characteristics are directly related with functionality of the service. Some services maybe considered as data services where the coverage of the data is defined by an auxiliary file. As an example, The Open Geographical Information Systems (GIS) Consortium (OGC) (<http://www.opengis.org>) defines standard for auxiliary capability file defining the data coverage of geospatial services. The prescriptive metadata may be defining the operations, inputs, outputs, data being provided, pre-conditions and post-conditions of a service. b) The descriptive characteristics consist of the service attributes that affect the quality of a service, such as performance or capacity. Both prescriptive and descriptive attributes form the static, interaction-independent metadata which characterizes a web service.

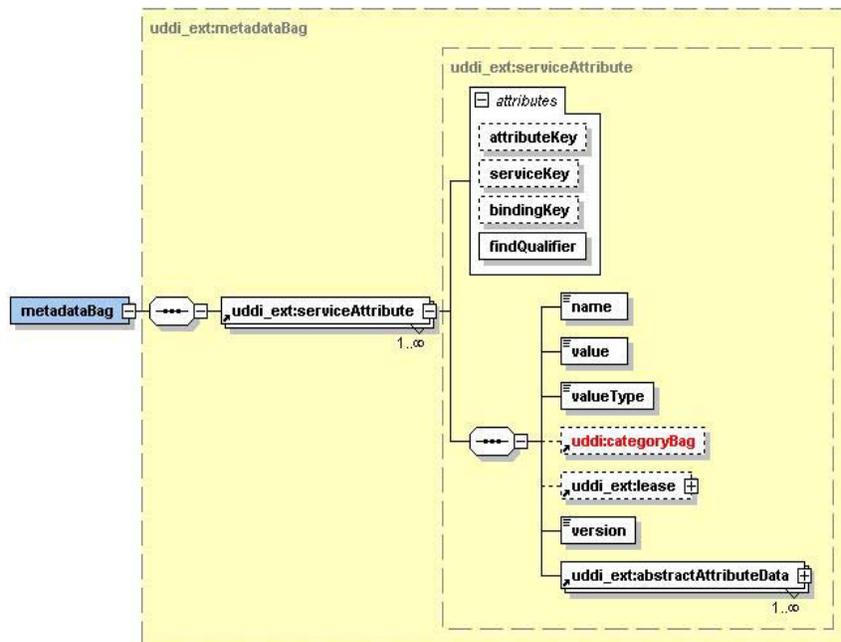


Figure 1: XML Schema for service metadata

Second, we extend UDDI Data Structure Schema. We design a data model for metadata describing services as illustrated in Figure 1. Based on this model, each service entry is associated with a “metadataBag”, which consists of one or more “serviceAttribute” elements where each “serviceAttribute” corresponds to a piece of metadata. Here, a metadata can be associated with a lifetime and can have a version number and can be categorized based on custom classification schemes. Each metadata describes either prescriptive or descriptive (QoS) characteristics of a service. Say, if the metadata is describing QoS characteristics, then we use (name, value) pairs (as illustrated in Figure 2) that are sub-elements of a “serviceAttribute” element for representation. This approach was first used in UDDIe project [8] originated in Cardiff University. Service metadata could be directly related with functionality of the service as well. Some services may provide auxiliary files providing catalog information about the data or properties of a service. For instance; Open GIS Services provide capabilities.xml file providing the data coverage of geospatial services. We design our model to dynamically retrieve, store and maintain auxiliary files (such as capabilities/properties files) of those services that present their metadata. We use an “abstractAttributeData” element to represent such metadata where we store and maintain these auxiliary files as-is. This model has simplicity in its nature and it is expandable since it can address both prescriptive and descriptive requirements of services.

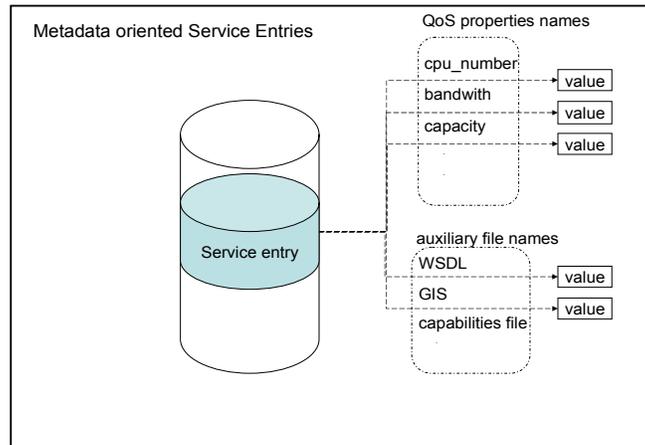


Figure 2: Metadata integrated service entries

Third, we extend UDDI XML API Schema to introduce metadata-oriented service publishing/discovery capabilities. We extend existing UDDI API such as “save_service”, “find_service” and “get_serviceDetail”. Our implementation is compatible with existing out-of-box UDDI Registry clients as well. One can publish/query service entries on the UDDI Registry w/o metadata-oriented capabilities introduced here. We also introduce additional API to provide publishing/query capabilities on the service metadata catalog. We implement “get_serviceAttributeDetail”, “save_serviceAttribute”, “find_serviceAttribute” and “delete_serviceAttribute” functionalities. Here, one can publish additional metadata to an already stored service entry as well as search for metadata associated with that entry.

Our approach for populating registry with metadata is as following. Say, a user publishes a new service into UDDI Registry. In this case, the user constructs a “metadataBag” filled with “serviceAttributes” where each “serviceAttribute” has (name, value) pairs. Each pair describes one generic descriptive characteristics of the service. The user also specifies the category of the service based on a classification, when the service is published. Here, the UDDI Registry detects whether the published service is an OGC Web Service by checking the classification of the service. If the published service is an OGC service, UDDI Registry dynamically invokes “getCapabilities” function of that OGC service to retrieve prescriptive metadata, i.e. “capabilities.xml” file. Each domain-specific metadata file is stored associated with service registries as-is.

Our approach for service discovery can be summarized as following. Simply, querying user constructs a query “metadataBag” consisting of a list of “serviceAttribute”s. Each “serviceAttribute” forms a search criterion. The constructed “metadataBag” is passed to UDDI Registry as an argument of the “find_service” function. The user has option to choose logical operations such as “and/or” between the lists of “serviceAttribute” criteria’s. We extend “find_service” functionality in a way to provide XPATH/RDQL query capabilities on the UDDI Registry. Say, in given a “serviceAttribute” element, one could indicate a) XPATH/RDQL query statement b) name of the auxiliary file and c) an

indication whether there is an XPATH query statement. In this case, if the search criterion is a XPATH/RDQL query, then the query is applied on the corresponding auxiliary files stored in the metadata catalog. The result will be a list of services that have the indicated auxiliary file and that satisfy the XPATH/RDQL query. This feature allows us to do spatial queries on the UDDI Service Registry. We use (name, value) pairs to indicate the name of an auxiliary files and the XPATH query statement in a search criterion. Service matchmaking process has been definitely investigated as in [10], [18], and [19] and so not covered in our design.

3.2. Supporting/integrating session metadata

Existing information services focus on the problems of management of quasi-static, stateless metadata related services and do not hold information to facilitate discovery based on dynamically generated session metadata. Here, session metadata is the generated metadata by the participants of an activity as result of their interactions. Such metadata describes the context of the session and has a lifetime.

There are different approaches specifying session metadata. For instance, WS-Context [12] provides an abstract context defining such metadata. WS_Context specification defines a context service that maintains a repository of context information.

Each session is started by the coordinator of an activity. The coordinator service publishes the session metadata to Information Service and gets a unique identifier in return. The uniqueness of the session-id is ensured by the Information Service. Sessions can obviously be composed from other “sub” sessions hierarchically. Each session is associated with the participant services of that session. This enables various rich query capabilities of interacting systems such as discovery of other entities within a session, discovery of any state that might need to be associated with a failed entity. As the session data is shared by the participants of an activity, we use space-based asynchronous communication model to ensure mutual-exclusive access to session data.

We find context information very valuable for discovering and managing services. Here, the context metadata is not only used for discovering services, but also used by services after they have been discovered. However, such context metadata is either not available to consumer of services or it does not capture dynamic behavior. We design Information Services that would support dynamically generated session metadata. We extend existing WS_Context Specifications and implement as part of our Information Services to support dynamically generated session metadata [2], [3]. The WS-Context part of an information service keeps track of context information shared between multiple participants in Web Service interactions. The context here has information such as unique ID and shared data. It allows a collection of action to take place for a common outcome. We utilize WS_Context Specification to maintain user profiles and preferences, application specific metadata, information regarding sessions and state of entities in these sessions.

4. Architectural Design

In SOA-based systems, Information Services support the discovery and handling of these services. Our design for Information Services provides a solution to the very general problem of managing information about Web Services, yet it can also support domain-specific metadata requirements of geospatial domain. Here, an information service supports not only quasi-static, stateless metadata, but also more extensive metadata requirements of rich interacting systems. An information service combines a) WS-Context and b) extended UDDI Specifications in one service as illustrated Figure 3 below. We extend existing UDDI Specifications to annotate service descriptions with metadata describing characteristics of services. We also extend WS-Context Specifications to manage session metadata between multiple participants in Web Service interactions.

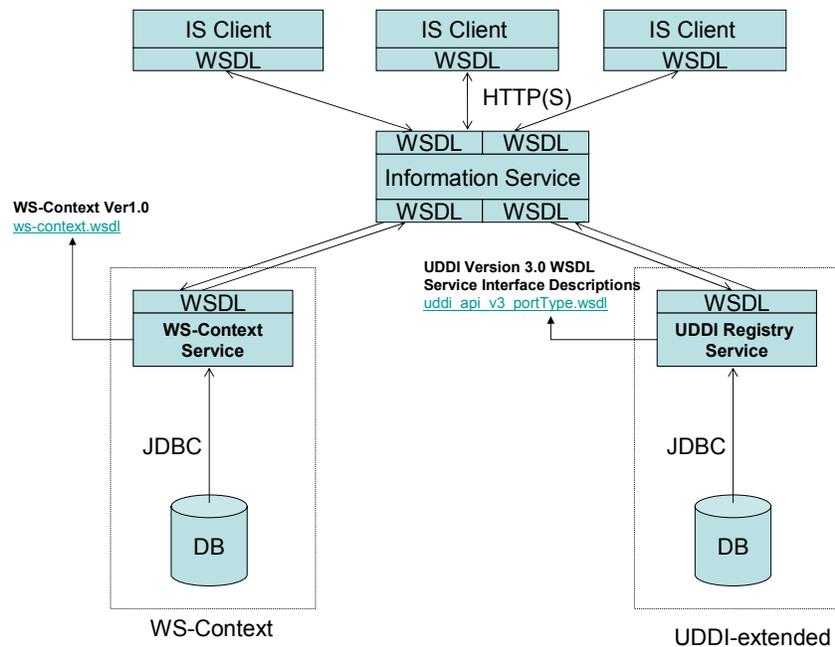


Figure 3: Architectural Design of an Information Service

We use jUDDI (version 0.9r3) as an open source Java-based implementation of a UDDI registry and a toolkit for developers to build access to UDDI registries. jUDDI has been architected to act as the UDDI front-end on top of existing directories and databases. (More at <http://www.juddi.org>)

In our design, we only use a portion of the jUDDI library as UDDI-front end. We extend the jUDDI library to implement extended version of UDDI Data Structure and UDDI XML API. We discard jUDDI Servlet architecture and implement the access to UDDI registries via Web Services by implementing UDDI (v.3.0) Service Descriptions.

5. Application Case Scenario

SERVOGrid project [22] integrates historical, measured, and calculated earthquake data with simulation codes. SERVOGrid resources are located at various institutions across the country. The SERVOGrid Complexity and Computational Environment (CCE) [20] is an environment to build and integrate different domains of Grid and Web Services into a single cooperating system [13]. As a part of SERVOGrid CCE environment, we chose the Pattern Informatics (PI) application [21] which is used to produce the well-publicized “hot spot” maps published by SERVO team member Prof. John Rundle and his group at the University of California-Davis.

We illustrate the PI motivating scenario where Information Services interacts with Web Map Services and Web Feature Services as service registry in Figure 4. In this scenario, Information Services are also used for storing transitory metadata needed to describe distributed session state information. In the current test system, it is used to store information needed by a workflow engine (HPSearch) to orchestrate system interactions. We can describe the components of this integrated application as following. (More info available at [16]).

Web Feature Service: The Web Feature Service (WFS) [14] is an Open Geospatial Consortium (OGC) based data service that manages “feature” data: abstract representations of map features and associated metadata. WFS servers are capable of storing map entities such as political boundaries, geographic features such as rivers, roads, etc.

Web Map Service: The Web Map Service (WMS) [15] is an OGC specification for generating interactive, online maps. WMS can generate maps in several formats (JPEG, SVG) by acting as client to both WFS and other WMS instances.

HPSearch: The HPSearch system provides a scripting environment for managing distributed Web Services. We use HPSearch as an engine for managing distributed workflows on SERVOGrid [17].

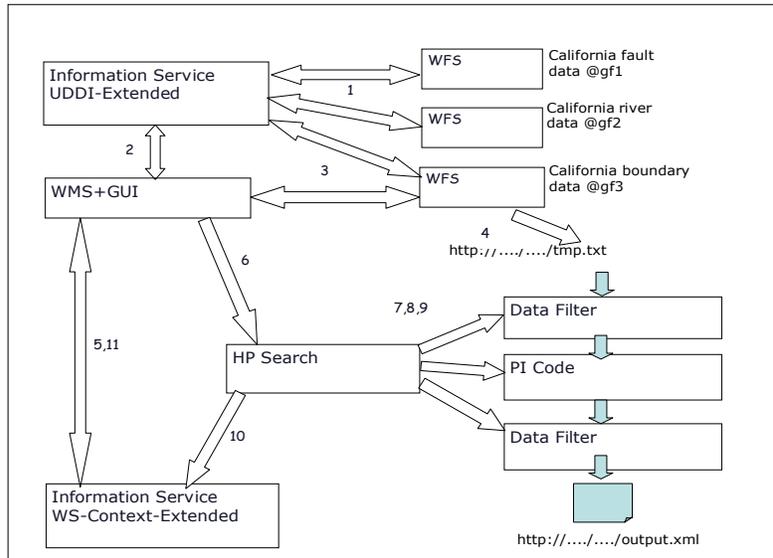


Figure 4: Application Case Scenario for GIS Compatible Information Services

In this scenario, Web Feature Services are published into the UDDI-Registry. Each Web Feature Service provides data layers corresponding to geographic entities. An important challenge is that UDDI does not natively support registry of services with a bounding box corresponding to a data layer and representing a location of interest. To overcome this problem, we provide XPATH query capabilities on the auxiliary metadata files such as capabilities.xml file. This methodology allows us to make coordinate based spatial queries on the UDDI-Registry.

We can summarize the steps of the scenario illustrated in Figure 4 as follows.

- 1: All WFS services are expected to register with an Information Service. The Information Service dynamically interacts with WFSs to retrieve and store auxiliary capabilities files associated to services. Each service makes a lease with the Information Service provided that the service will be up and running during that time period.
- 2: Web Map Services interact with the Information Service to find out available WFS (data services) satisfying the data requirements of a map. As the Information Service responds a query of WMS with metadata of services satisfying the query, WMS can then start interacting with corresponding WFS to acquire the data layers needed to create maps.
- 3: WMS can query WFS for a given bounding box and time interval.
- 4: WFS dumps the results into a web accessible .txt file
- 5, 6: WMS starts a session, invokes HPSearch to run workflow script for PI Code with a session id
- 7, 8, 9: HPSearch runs the workflow script and generates output file in GML format as result
- 10: HPSearch writes the URI of the of the output file into Context
- 11: WMS polls the information from Context Service

12: WMS retrieves the generated output file's location from the Context Service, downloads the data, and generates a map.

6. Conclusions and Future Work

In order to provide methodologies for discovering services based on both general and domain-specific search criteria, UDDI Specifications seems promising, yet limited. An extended version of UDDI Information Model can be used to provide domain-specific discovery capabilities such as spatial queries. Such information model could also provide metadata augmented service registries, where one can support/integrate quasi-static, stateless metadata with service entries. Information Services should support not only static metadata, but also dynamically generated session metadata. This way, Information Services can satisfy more extensive metadata requirements of rich interacting systems. In this study, we designed and implemented Information Services providing such capabilities. We tested our implementation in SERVOGrid project, which is a SOA based computational environment.

Work remains further develop implementation of Information Services and provide a novel approach for building P2P/Grid Fault Tolerant and High Performance Information Systems and dynamic and decentralized context management in P2P/Grid Environment.

Acknowledgement

This work is supported by the Advanced Information Systems Technology Program of NASA's Earth-Sun System Technology Office.

References

1. The Open Geospatial Consortium, Inc. (OGC) Web Site: <http://www.open-geospatial.org/>
2. Mehmet S. Aktas, Geoffrey C. Fox and Marlon Pierce, "Managing Dynamic Metadata as Context", Submitted to The 2005 Istanbul International Computational Science and Engineering Conference (ICCSE2005), Istanbul, Turkey.
3. Fault Tolerant High Performance Information Services Web Site: <http://grids.ucs.indiana.edu/~maktas/fthpis>
4. Bellwood, T., Clement, L., and von Riegen, C. (eds) (2003), *UDDI Version 3.0.1: UDDI Spec Technical Committee Specification*. Available from <http://uddi.org/pubs/uddi-v3.0.1-20031014.htm>.
5. OASIS/ebXML Registry Technical Committee. OASIS/ebXML Registry Information Model v2.0 Approved Committee Specification. <http://www.oasis-open.org/committees/regrep/documents/2.0/specs/ebRIM.pdf>

6. The JINI Specification, Ken Arnold, Ann Wollrath, Byran O'Sullivan, Robert Scheifler, and Jim Waldo, Addison-Wesley, Reading, MA, 1999
7. Open GIS Consortium Inc. OWS-1 Registry Service. 2002-07-26
8. UDDIe: An Extended Registry for Web Services
Ali ShaikhAli, Omer Rana, Rashid Al-Ali and David W. Walker. Proceedings of the Service Oriented Computing: Models, Architectures and Applications, SAINT-2003 IEEE Computer Society Press. Orlando Florida, USA, January 2003
9. Verma, K., Sivashanmugam, K. , Sheth, A., Patil, A., Oundhakar, S. and Miller, J. "METEOR-S WSDI: A Scalable P2P Infrastructure of Registries for Semantic Publication and Discovery of Web Services" , Journal of Information Technology and Management. TM03-006.pdf
10. Personalized Grid Service Discovery. Miles, S., Papay, J., Dialani, V., Luck, M., Decker, K., Payne, T., and Moreau, L. Nineteenth Annual UK Performance Engineering Workshop (UKPEW'03), University of Warwick, Coventry, England, 2003.
11. OWS1.2 UDDI Experiment. OpenGIS Interoperability Program Report OGC 03-028, Open GIS Consortium, Inc., January 17 2003.
<http://www.opengeospatial.org/docs/03-028.pdf>
12. Bunting, B., Chapman, M., Hurlery, O., Little M., Mischinkinky, J., Newcomer, E., Webber J, and Swenson, K., Web Services Context (WS-Context), available from http://www.arjuna.com/library/specs/ws_caf_1-0/WS-CTX.pdf
13. Chen, A., Donnellan, A., McLeod, D., Fox, G., Parker, J., Rundle, J., Grant, L., Pierce, M., Gould, M., Chung, S., and Gao, S., "Interoperability and Semantics for Heterogeneous Earthquake Science Data", International Workshop on Semantic Web Technologies for Searching and Retrieving Scientific Data, Sanibel Island, FL, October 2003.
14. SERVOGrid WFS implementation web page:
<http://www.crisisgrid.org/html/wfs.html>
15. SERVOGrid WMS implementation home page:
<http://complexity.ucs.indiana.edu/~asayar/wms>
16. Crisis Grid Web Site, Geographic Information Systems Research at Community Grids Lab, <http://www.crisisgrid.org>
17. Harshawardhan Gadgil, Geoffrey Fox, Shrideep Pallickara, Marlon Pierce, Robert Granat, Proceedings of the IEEE/ACM Cluster Computing and Grid 2005 Conference, CCGrid 2005, Cardiff, UK.
See also HPSearch Web Site, <http://www.hpsearch.org>
18. D. Chakraborty, F. Perich, S. Avancha, and A. Joshi, DReggie: A Smart ServiceDiscovery Technique for E-CommerceApplications, In Workshop in conjunctionwith20thSymposiumonReliableDistributed Systems, October 2001.
19. Paolucci, M., Kawamura, T., Payne, T. R. and Sycara, K., Importing the Semantic Webin UDDI. (2002). In Proceedings of WebServices, E-Business and Semantic WebWorkshop, CAiSE 2002. , pages pp. 225-236, Toronto, Canada.
20. Geoffrey Fox, et al, "Complexity Computational Environments (CCE) Architecture." Technical report available from
<http://grids.ucs.indiana.edu/ptliupages/publications/CCE%20Architecture.doc>.
21. Tiampo, K. F., Rundle, J. B., McGinnis, S. A., & Klein, W. Pattern dynamics and forecast methods in seismically active regions. Pure Ap. Geophys. 159, 2429-2467 (2002).

22. Marlon Pierce, Choonhan Youn and Geoffrey Fox: Interacting Data Services for Distributed Earthquake Modeling. ACES Workshop at ICCS June 2003 Australia
23. Singh, G., Bharathi, S., Chervenak, A., Deelman, E., Kesselman, C., Manohar, M., Patil, S. and Pearlman, L. A Metadata Catalog Service for Data Intensive Applications. SC'03, November 15-21, 2003, Phoenix, Arizona, USA
24. Peisheng Zhao , Aijun Chen , Yang Liu , Liping Di , Wenli Yang , Peichuan Li, Grid metadata catalog service-based OGC web registry service, Proceedings of the 12th annual ACM international workshop on Geographic information systems, November 12-13, 2004, Washington DC, USA