Processing and Managing Scientific Data in SOA Environment

BOGDAN SHISHEDJIEV, MARIANA GORANOVA, JULIANA GEORGIEVA, VESKA GANCHEVA
Faculty of Computer Systems and Control
Technical University of Sofia
1000 Sofia, 8 Kliment Ohridski
BULGARIA
bogi@tu-sofia.bg, mgor@tu-sofia.bg, july@tu-sofia.bg, vgan@tu-sofia.bg

Abstract: - Increased complexity of scientific research poses new challenges to scientific data management. Meanwhile, scientific collaboration is becoming increasing important, which relies on integrating and sharing data from distributed institutions. Scientific experiments require effective and efficient management of data. In this paper we present an integrated, extensible architecture based on service-oriented approach that addresses large collections of heterogeneous scientific data. This architecture provides capabilities to scientists to model their experiments, enabling complex interconnections between computational simulations, data transformations applications, and analysis and visualisation tools.

Key-Words: - service-oriented architecture (SOA), scientific data, meta data, data management.

1 Introduction

SOA [1] is an architectural style for building software applications that use services available in a network such as the web. It promotes loose coupling between software components so that they can be reused. Applications in SOA are built based on services. A service is an implementation of a well-defined business functionality, and such services can then be consumed by clients in different applications or business processes. SOA allows for the reuse of existing assets where new services can be created from an existing IT infrastructure of systems. In other words, it enables businesses to leverage existing investments by allowing them to reuse existing applications, and promises interoperability between heterogeneous applications and technologies. SOA provides a level of flexibility that wasn't possible before in the sense that:

- Services are software components with well-defined interfaces that are implementation-independent. An important aspect of SOA is the separation of the service interface (the what) from its implementation (the how). Such services are consumed by clients that are not concerned with how these services will execute their requests.
- Services are self-contained (perform predetermined tasks) and loosely coupled (for independence).
- Services can be dynamically discovered.
- Composite services can be built from aggregates of other services.

Scientists have been dealing with a variety of heterogeneous data sources: text files, ASCII files, XML documents, relational databases. Scientists are producing data that has to be managed from the creation of the raw data from sensors and instruments. The raw data has to be transformed into standard datasets, open and selected for scientific use, processed and experimental analysed, and published and archived. The strategy is to apply reusable principles – turning data into service that is available as logical modules, each with a standards-based interface. This allows scientists to access and to use the service more easily, improves data visibility and promotes greater reuse.

To address the challenges of scientific non-standard data formats, that do not have transformation, parsing, querying, analysis and visualisation tools readily available, researchers have begun to develop systems for modelling and processing scientific data. For instance, modelling generic oceanographic data [2] is based on commonalities across many data types used in ocean environmental research. Trident is an ocean scientists' workbench [3] that converts raw sensor data into useful data products, in particular visualisations. Trident allows scientists to explore and visualise oceanographic data in real-time and provides an environment to compose, run and catalogue workflows. The Global Geodynamics Project [4] allows Earth scientists the ability to leverage a network of globally distributed instruments for operational and research activities into Earth tides. The PANIC system [5] is an integrated, extensible architecture based on preservation metadata, automatic notification services, software and format registries and semantic grid services. The astronomers use the PANIC system to specify the parameters they require in the conversion service.

The PADS system [6] allows data analysts to describe both the physical layout of ad hoc data sources and partially the semantic properties of that data. The researches at AT&T examine call detail data, web server logs, net flows capturing internet traffic, log files characterizing IP backbone resource utilisation, wire formats for legacy telecommunication billing systems.

The XML based system [7] describes and accesses fusion and plasma physics simulation data of various formats from major data analysis and visualisation tools. The system enhances the interoperability between various data formats and analysis tools used in the fusion and plasma simulation community. The ServOSims SOA Framework [8] is using for the composition and execution of Multidisciplinary Simulation Models (e.g. Meteorological, Hydrological, Pollution, Fire Propagation) based on Application and File WS-Resources. Iterative workflows for numerical simulations in subsurface sciences [9] are built to assess risks of radiologically and chemically hazardous "legacy waste" during the development and manufacture of nuclear weapons and nuclear reactors. The workflow techniques and technologies enable both engineering and scientific research uses of complex models of subsurface flow and contaminant transport.

The X-SIGMA [10] is a Grid-based integrated system developed as a scientific management system for the cyber infrastructure for six civil engineering testing centres in Korea.

2 Scientific Background

The goal of this research is the intelligent management and data visualisation including experiments, numerical simulations, data organisation, analysis and publication. The service-oriented architecture in combination with Web services is the best scalable solution for application architecture. Web services are software systems designed to support interoperable machine-to-machine interaction over a network. This interoperability is gained through a set of XML-based open standards, such as WSDL, SOAP, and UDDI. These standards provide a common approach for defining, publishing, and using web services. Sun's Java Web Services Developer Pack 1.5 (Java WSDP 1.5) and Java 2 Platform, Enterprise Edition (J2EE) 1.4 can be used to develop state-of-the-art web services to implement SOA. The J2EE 1.4 platform enables you to build and deploy web services in your IT infrastructure on the application server platform. It provides the tools you need to quickly build, test, and deploy web services and clients that interoperate with other web services and clients running on Java-based or non-Java-based platforms. In addition, it enables businesses to expose their existing J2EE applications as web services. Servlets and Enterprise JavaBeans components (EJBs) can be exposed as web services that can be accessed by Java-based or non-Java-based web service clients. J2EE applications can act as web service clients themselves, and they can communicate with other web services, regardless of how they are implemented.
The advantages in data management, analysis, knowledge discovery and visualisation empower the scientists to achieve new scientific breakthroughs. As a result the research work directed towards developing architecture for solving problems in different scientific fields.

We are specifically gathering data and information on the investigations of Bulgarian Academy of Science, Solar-Terrestrial Influences Laboratory.

As first attempt we gathered experimental and simulation data from different sources, especially from radiation and spectral measurements made in the Institutes of the Bulgarian Academy of Science and simulation data of magnetic fields in human tissues.
The particle telescope Liulin-5 [11] is an adherent part of the international project MATROSHKA-R on the International Space Station for investigating the space radiation doses distribution in the human body using human models – tissue-equivalent phantoms. The radiation environment measurements are very important to predict the effects of radiation on humans during a long-term space mission and requires: accurate knowledge and modelling of the space radiation environment, calculation of primary and secondary particle transport through the shielding materials and through the human body, and assessment of the biological effect of cosmic particles, especially that of the high energy particles in the heavy ion component particles. Our approach will allow scientists to receive more accurate estimation on the organism hazard, caused by the space ionising radiation and to determine more accurately the radiation risk values of space flights.

The multi-satellite INTERBALL [12] project was devoted to investigation of the energy transfer process from Sun to Earth via the magnetosphere. The Low Energy Ion Composition experiment aboard INTERBALL was performed by means of the Bulgarian-Russian energy mass spectrograph AMEI-2. Measurement data for the energy-angular spectra of H^+, He^{++} and O^+ in Earth magnetosphere provide information contributing to solve one of the major problems in magnetospheric physics – the processes of magnetosphere plasma supply.

The monitoring of the slant column content of atmospheric minor gases such as O_3 and NO_2 is performed by the spectrometer GASCOD-BG [13] (Gas Analyzer Spectrometer Correlating Optical absorption Differences). The objectives of the data analysis are to find out the variability of the NO_2 and O_3 and its reasons,

which are of chemical and dynamical nature and are in close connection to the climate change problems and to the environmental protection. The establishment of long time trends is very important with regards to the climate change.

Within our project we use a flexible, dynamic, automated approach which provides access to a range of tools and services through a service-oriented architecture. Our solution is composed of services, which access scientific data from different data source types with different formats, transform raw data into standard data sets that can be analysed, processed and visualised. Examples that we face include the low energy ion composition experiment; the charged particles spectra and radiation dose distribution inside the tissue equivalent phantom of human body aboard the International Space Station, the slant column content of atmospheric minor gases, simulation of a low-frequency magnetic field generated of electromagnetic devices with different construction placed directly on the surface of the human body, experimental data measured on the surface of the electromagnetic devices with therapeutic application.

3 Architecture

The architectural approach for processing, managing and visualisation scientific data is a Service Oriented Architecture. Figure 1 shows the proposed architecture. This architecture provides interoperability between different application and sources via standards-based interfaces. It allows distribution of services over heterogeneous systems and computing environments. The architecture integrates .NET applications using C# clients, J2EE applications using Java message service, IBM WebSphere MQ applications and different data sources using services.

The architecture on Fig.1 implements a technology which allows an inexperienced user to prepare, to manipulate and to visualise its own or any other data available. It serves as a middle tier between various data formats and analysis tools. This includes an advisor that helps the user to make a XML description of the dataset(s). The advisor can be a mere Web application or applet that can use Web services.

A compiler is proposed to work out modules for converting the raw data layout in a canonical form based only on the data semantics. Another compiler is provided to work out manipulation and visualisation routines. The generated APIs allow user to parse, query and retrieve data from data visualisation and analysis tools. All routines take form of Web services and can be used throughout the Web. The purpose of the canonical layout is to standardise the input for the manipulation and visualisation routines and in that way to minimize their number. That uses the well known approach to use an intermediate format (or language) when there are a lot of input formats and a lot of output formats.

A query engine enables creation of data services to abstract the complexity of underlying data sources.

Services can be programmed using application development tools like Microsoft Visual Studio .NET, NetBeans or BEA WebLogic Workshop, which allow distributed applications to be exposed as Web services.

The proposed architecture illustrates how applications running on different platforms are abstractly decoupled from each other and can be connected together with services.

4 Meta Description of Scientific Data

The common XML Schema and uniform XML metadata are designed to describe semantically and structurally scientific data. The data descriptions are stored in a database in order queries to be made. The raw data description contains three main sections (Fig.2). The “general” section is about the ownership, the purpose and other general dataset proprieties. The “semantics” section provides information about the data nature and existing relationships in the dataset. The “layout” section describes the physical storing scheme of the dataset. Data description details are shown on Fig. 3. The main sections are as follows:

Section 2, named “general” which describes: a) the data identification; b) the domain and annotation; c) the source ant its description; d) the procedures; e) the version information; f) the access rights; g) the information about contact person(s).

Section 3 with the name “semantics” is about data relationships. It includes: a) information about data parameters; b) all necessary information about the independent and dependant parts of data.

Section 4 gives an information about the data structures.

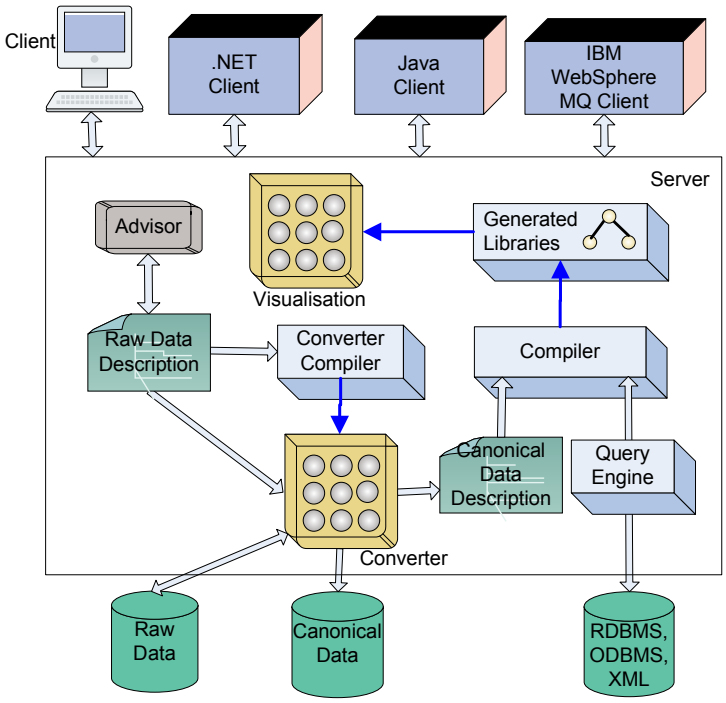

Fig. 1 System architecture

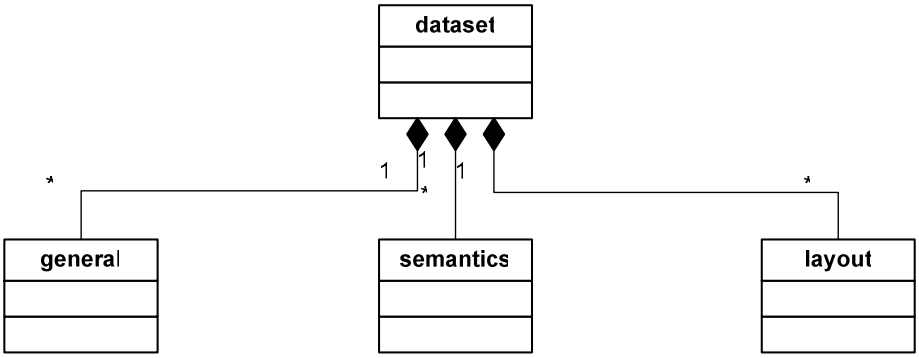

Fig. 2 XML Schema data description

1. dataset
2. general
   2.1. set_ID
   2.2. about
      2.2.1. domain
      2.2.2. keywords {
         2.2.2.1. keyword}*
      2.2.3. annotation
   2.3. source
      2.3.1. timestamp
      2.3.2. method
         2.3.2.1. [instrument]
         2.3.2.2. [procedure]
         2.3.3. [parents {
            2.3.3.1. parent_name (URI)
            2.3.3.2. pre_procedure (Name, URI, description)}+]
         2.3.4. [children {
            2.3.4.1. child_name (URI)
            2.3.4.2. post_procedure (Name, URI, description}+]
   2.4. [procedures /*native procedures availables*/ {
      2.4.1. Procedure name
      2.4.2. Procedure description
      2.4.3. Procedure distribution /* URI, WSDL description */}+]

2.5. version_info
2.5.1. version_no
2.5.2. ver_timestamp
2.5.3. [modification_info]
2.6. access_rights
2.6.1. ownership
2.6.2. distribution_rights
2.6.3. read_rights
2.7. [contact Information
2.7.1. contact Person
2.7.1.1. contact Position (Optional)
2.7.1.2. contact Address (Mandatory)
2.7.1.3. {contact Voice Telephone }+
2.7.1.4. [contact Facsimile Telephone]
2.7.1.5. {Contact Electronic Mail Address}+
2.7.1.6. [Hours of Service]
2.7.1.7. [Contact Instructions]
2.7.2. [Contact Organization (Optional)
2.7.2.1. Address
2.7.2.2. 10.4.3 City
2.7.2.3. 10.4.4 State or Province
2.7.2.4. 10.4.5 Postal Code
2.7.2.5. 10.4.6 Country]
3. semantics {
3.1. data parameters
3.1.1. {parameter
3.1.1.1. id
3.1.1.1.1. (type|type_ref)
3.1.1.1.2. name
3.1.1.1.3. layout_ref
3.1.1.1.4. [constraint]}*
3.1.1.1.5. [description]
3.1.1.2. (type|type_ref)
3.1.1.3. name
3.1.1.4. {component
3.1.1.4.1. (type|type_ref)
3.1.1.4.2. name
3.1.1.4.3. [unit]
3.1.1.4.4. {component
3.1.1.4.4.1. (type|type_ref)
3.1.1.4.4.2. name
3.1.1.4.4.3. [unit]
3.1.1.4.4.4. layout_ref
3.1.1.4.4.5. [description]
3.1.1.4.4.6. [constraint]}*
3.1.1.4.5. layout_ref
3.1.1.4.6. [description]
3.1.1.4.7. [constraint]}*}*
3.2. independent
3.2.1. time
3.2.1.1. id
3.2.1.1.1. (type|type_ref)
3.2.1.1.2. name
3.2.1.1.3. layout_ref
3.2.1.1.4. [constraint]}*
3.2.1.1.5. [description]
3.2.1.1.6. {coordinate
3.2.1.1.6.1. (type|type_ref)
3.2.1.1.6.2. name
3.2.1.1.6.3. [unit]
3.2.1.1.6.4. layout_ref
3.2.1.1.6.5. [constraint]
3.2.1.1.6.6. [description]
3.2.1.1.6.7. [case]*
3.2.1.1.6.8. [parameter
3.2.1.1.6.8.1. (type|type_ref)
3.2.1.1.6.8.2. name
3.2.1.1.6.8.3. [unit]
3.2.1.1.6.8.4. layout_ref
3.2.1.1.6.8.5. [constraint]
3.2.1.1.6.8.6. [description]]*}*
3.2.2. space
3.2.2.1. dimension
3.2.2.2. coordinate_system
3.2.2.3. point
3.2.2.3.1. id
3.2.2.3.1.1. (type|type_ref)
3.2.2.3.1.2. name
3.2.2.3.1.3. layout_ref
3.2.2.3.1.4. [constraint]}*
3.2.2.3.1.5. [description]
3.2.2.3.2. {coordinate
3.2.2.3.2.1. (type|type_ref)
3.2.2.3.2.2. name
3.2.2.3.2.3. [unit]
3.2.2.3.2.4. layout_ref
3.2.2.3.2.5. [constraint]
3.2.2.3.2.6. [description]
3.2.2.3.2.7. [case]*
3.2.2.3.2.8. [parameter
3.2.2.3.2.8.1. (type|type_ref)
3.2.2.3.2.8.2. name
3.2.2.3.2.8.3. [unit]
3.2.2.3.2.8.4. layout_ref
3.2.2.3.2.8.5. [constraint]
3.2.2.3.2.8.6. [description]]*}*
3.2.3. {other
3.2.4. id
3.2.4.1. (type|type_ref)
3.2.4.2. name
3.2.4.3. layout_ref
3.2.4.4. [constraint]}*
3.2.4.5. [description]
3.2.4.6. {coordinate
3.2.4.6.1. (type|type_ref)
3.2.4.6.2. name
3.2.4.6.3. [unit]
3.2.4.6.4. layout_ref
3.2.4.6.5. [constraint]
3.2.4.6.6. [description]

3.2.4.6.7. [case]*
3.2.4.6.8. [parameter
3.2.4.6.8.1. (typeltype_ref)
3.2.4.6.8.2. name
3.2.4.6.8.3. [unit]
3.2.4.6.8.4. layout_ref
3.2.4.6.8.5. [constraint]
3.2.4.6.8.6. [description]]*}*
3.3. dependant
3.3.1. field_value
4. layout{
4.1. [typedef]*
4.2. cluster
4.2.1. [clust_name]
4.2.2. clust_URI(URI)
4.2.3. Struct_description}*

[] – means an optional element
{}* – 0 or more element of this type
{}+ – 1 or more element of this type

Fig. 3 XML Schema data description (details)

5 Conclusion

The advent of web services and SOA offers potential for lower integration costs and greater flexibility. An important aspect of SOA is the separation of the service interface (the what) from its implementation (the how). Web services are the next step in the Web's evolution, since they promise the infrastructure and tools for automation of business-to-business relationships over the Internet.

In this paper we propose a SOA architecture model for scientific data processing. The proposed architecture describes the content of scientific data, gives scientists the opportunity to process their data easier and faster, which is of critical importance to make service-oriented computing paradigm operational in a business context.
The detailed structure for scientific data description using XML Schema is created.

Acknowledgments
This work was supported by Ministry of Education and Science in Bulgaria, National Scientific Fund contract DO 02-175/2008.

References:

[1] T. Erl, *SOA: Concepts, Technology, and Design*, Prentice Hall PTR, ISBN: 0-13-185858-0, 2005.
[2] A. Isenor, J. Keeley, "Modelling Generic Oceanographic Data Objects in XML", *Computing in Science & Engineering*, pp. 58-66.
[3] R. Barga, J. Jackson, N. Araujo, D. Guo, N. Gautam, K. Grochow, E. Lazowska, Trident: Scientific Workflow Workbench for Oceanography, *IEEE Congress on Services*, Part I, 2008, pp.465-466.
[4] L. Lumb, K. Aldridge, Grid-Enabling the Global Geodynamics Project: The Introduction of an XML-Based Data Model, *Proceedings of the 19th International Symposium on High Performance Computing Systems and Applications (HPCS'05),* 2005.
[5] J. Hunter, S. Choudhury, Semi-Automated Preservation and Archival of Scientific Data using Semantic Grid Services, *IEEE International Symposium on Cluster Computing and the Grid*, 2005, pp. 160-167.
[6] K. Fisher, R. Gruber, PADS: A Domain-Specific Language for Processing Ad Hoc Data, *PLDI'05,* 2005, pp. 295-304.
[7] S. Shasharina, Ch. Li, FSML: Fusion Simulation Markup Language for Interoperability of Data and Analysis Tools, *Proceedings of the Challenges of Large Applications in Distributed Environments, CLADE,* 2005.
[8] E. Floros, Y. Cotronis, ServOSims: A Service Oriented Framework for Composing and Executing Multidisciplinary Simulations, *Proceedings of the Second IEEE International Conference on e-Science and Grid Computing (e-Science'06),* 2006.
[9] J. Chase, K. Schuchardt, G. Chin, J. Daily, T. Scheibe, Iterative Workflows for Numerical Simulations in Subsurface Sciences, *IEEE Congress on Services,* Part I, 2008, pp. 461-464.
[10] D. Kim, K. Jeong, S. Hwang, K. Won Cho, X-SIGMA: XML based Simple data Integration system for Gathering, Managing, and Accessing Scientific Experimental Data in Grid Environments, *Proceedings of the Second IEEE International Conference on e-Science and Grid Computing (e-Science'06)*, 2006.
[11] J. Semkova, R .Koleva V. Shurshakov, V. Benghin, St. Maltchev, N. Kanchev, V. Petrov, E. Yarmanova, I. Chhernykh, Calibration results of Liulin-5 charged particle telescope obtained in ICCHIBAN-7 experiment, New instrumentation for radiation monitoring on interplanetary missions, *11th WRMISS*, 2006.
[12] R. Koleva., Case study of plasma in the near magnetospheric lobes, *Compt. Rend. Acad. Bulg. Sci.* v.60, No11, 1221, 2007.
[13] R. Werner, The latitudinal ozone variability study using wavelet analysis, *Journal of Atmospheric and Solar-Terrestrial Physics,* Volume 70, Issue 2-4, 2008, p. 261-267.