An Initial Model of the Computation Viewpoint
for a Spatial Data Infrastructure

Antony Cooper[1], CSIR, acooper@csir.co.za; **Harold Moellering**, Ohio State University, geohal+@osu.edu; **Tatiana Delgado**, National Commission of the SDI of Cuba, internacional@geocuba.co.cu, **Ulrich Düren**, Landesvermessungsamt Nordrhein-Westfalen, ulrich.dueren@lverma.nrw.de, **Jan Hjelmager**, Kort & Matrikelstyrelsen, jnh@kms.dk, **Michel Huet**, International Hydrographic Bureau, pac@ihb.mc, **Petr Rapant**, VSB-Technical University of Ostrava, petr.rapant@vsb.cz, **Abbas Rajabifard**, University of Melbourne, abbas.r@unimelb.edu.au, **Dominique Laurent**, Institut Geographique National, dominique.laurent@ign.fr, **Adam Iwaniak**, Akademii Rolniczej we Wrocławiu, iwaniak@ar.wroc.pl, **Paloma Abad**, Instituto Geográfico Nacional, pabad@mfom.es *and* **Alexander Martynenko**, Russian Academy of Sciences, a_Martynenko@mail.ru.

1. Abstract

The Commission on Spatial Data Standards of the International Cartographic Association (ICA) is working on defining formal models and technical characteristics of Spatial Data Infrastructures (SDI). The Commission has already presented models of the Enterprise and Information Viewpoints from the ISO Reference Model for Open Distributed Processing (RM-ODP) standard (ISO/IEC 10746:1995). The Commission is now taking this further to model the Computation Viewpoint, which describes how the different services of an SDI fit together. The models should be seen as a continuing step towards the overall model of the SDI and its technical characteristics.

The Commission has identified six broad groupings of services: Registry, Data, Processing, Portrayal, Application and Management. The interactions between these high-level services have been modelled using the Unified Modeling Language (UML) Component Diagrams. The detailed services have been modelled using UML Class Diagrams (Object Management Group 2005).

Keywords: Spatial Data Infrastructure, SDI, Spatial Data, Analytical Cartography, Spatial Theory, Reference Model, Computation Viewpoint, Unified Modelling Language, UML.

---

[1] Corresponding author: email: acooper@csir.co.za

2. Introduction and Fundamental Concepts Involved

Over many decades, spatial data scientists in many parts of the world have worked to develop mechanisms to share various kinds of scientific spatial data. More recently, efforts have been organized to design and build Spatial Data Infrastructures (SDI) at the global, regional and national levels. There are two major facets to this effort: the organizational and the scientific/technical. The organizational side is for various groups to develop agreements, protocols and policy strategies for the SDI, see: Groot & McLaughlin (2000), Masser (2005), Rajabifard *et al* (2006), and Delgado (2005). Many organizations, such as the Global Spatial Data Infrastructure (GSDI), have been organized at the Global, Regional, National and Province/State levels.

The second facet of the SDI effort is the scientific and technical: implementing the mechanisms and networks that will bring the SDI to reality. This paper focuses on this facet of the SDI effort. More specifically, it discusses the efforts of the ICA Spatial Data Standards Commission to model the SDI at the scientific level through the prism of the RM-ODP model views, using the UML modeling language.

Earlier research work by the Commission began with a general overview of the SDI (Alders & Moellering, 2001), and then proceeded to examine the analysis using the UML modeling language (Cooper *et al* 2003) as a very useful tool. The Commission then proceeded to create RM-ODP models of the Enterprise and Information Viewpoints (Hjelmager et al., 2005, 2007). Presented here is the work by the Commission to model the RM-ODP Computation Viewpoint of the SDI.

3. RM-ODP Views of Distributed Software and Information Systems

Business and enterprise systems are becoming more and more flexible and increasingly dependant on local and global communication networks. To meet this challenge, the software systems used have to become increasingly modular, and distributed to many places within these networks. The design and installation of such distributed systems is a complex task which needs much conceptual work before the implementation can begin. The International Standard, ISO/IEC 10746, Information technology – Open Distributed

Processing – Reference model, aids in this challenge by providing a framework for the design and description of distributed software and information systems.

For this purpose, the RM-ODP Reference Model describes five different viewpoints:

- Enterprise Viewpoint
- Information Viewpoint
- Computation Viewpoint
- Engineering Viewpoint
- Technology Viewpoint

Referring to Figure 1, the Commission has completed initial formal models of the Enterprise and Information Viewpoints of SDIs (Hjelmager *et al* 2007). It now presents an initial formal model of the Computational Viewpoint for SDIs. One is not certain at this stage if it would be appropriate to develop generic Engineering and Technology Viewpoints of SDIs, as these viewpoints might be too implementation specific.
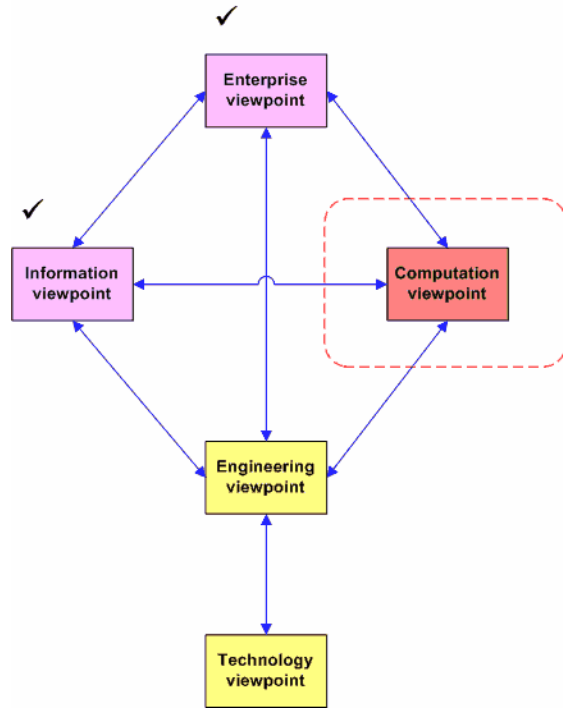

Figure 1: The five RM-ODP Viewpoints and their dependencies

The Enterprise Viewpoint describes the purpose, scope and policies for an Open Distributed Processing System. It describes the relationship of a system to its environment, its role, and the associated policies. The Information Viewpoint describes the semantics of information and information processing incorporated into the system. It could define conceptual schemas (formal descriptions of the model) and methods for defining application schemas. The Computational Viewpoint is a functional decomposition of the system into a set of services that interact through interfaces. This Viewpoint captures the details of these services and interface definitions without regard to their distribution. The Engineering Viewpoint contains the mechanisms and functions required to support distributed interaction between the services and data within the system. This Viewpoint is concerned primarily with the interaction between distinct services and

data. Its chief concerns are: communication, computing systems, software processes, and the clustering of computational functions at physical nodes of a communications network. The Technology Viewpoint contains the specific technologies chosen for the implementation.

As for all distributed software and information systems, this RM-ODP framework can also be used for the description of a Spatial Data Infrastructure (SDI). The first two Viewpoints (Enterprise and Information) have been described in detail by the Commission (Hjelmager *et al* 2005, 2007), using UML. The main results are summarized below.

3.1 Overview of the Enterprise and Information Viewpoints of an SDI

The investigation of the Enterprise Viewpoint of an SDI led to two actions:

A. Identify the stakeholders of an SDI and their tasks – in UML terminology, the Actors and the Use Cases; and

B. Identify the core components of an SDI and their relations.

Six SDI-Actors have been defined: Policy Maker; Producer (of data or services); Provider (of data or services); Broker; Value Added Reseller (VAR) and End User.

For these stakeholders, the Commission identified 24 tasks or activities (Use Cases), where an Actor can have more than one activity, which can be *Active* or *Passive*, eg: *Determine Vision and Mission*; *Set policy*; *Apply policy*; *Build infrastructure*; *Provide Services, Data, and Metadata*; and *Use Service*.

The second effort led to a UML Class Diagram that comprises the identified core components as *Classes* and their interconnections as *Relations*. The core components are:

- Policies
- Connectivity
- Technology
- Processing Tools
- Metadata
- Products

Taking a closer look at the component *Policies*, one finds that it can be subdivided into the concrete parts *Business Model*, *Business Agreements*, *Legal Constraints*, *Standards*, and *Best Practices*.

While the Enterprise Viewpoint deals mainly with the administrative aspects of an SDI, the Information Viewpoint has its focus on the products (data and services), their specification, their description via metadata, and product registries (catalogues). These aspects and their relations were also modelled in a UML class diagram.

The entities of the Information Viewpoint have been combined with the identified Stakeholders (Actors) from the Enterprise Viewpoint and the possible Activities in the SDI field. Hjelmager *et al* (2005) provides a table showing the coherence between all parts.

4. The RM-ODP Computation Viewpoint

The *Computation Viewpoint* is a functional decomposition of the modelled system into a *set of services* that interact through *Interfaces* (as defined by RM-ODP). This Viewpoint captures the details of these services and interface definitions without regard to distribution, which is covered by the *Engineering Viewpoint*. But, the Computational Viewpoint prepares the environment for distribution just by decomposing the system. UML provides a very good tool for modelling the Computation Viewpoint, namely the *Component Diagram*.

4.1 Developing a UML Component Diagram

The main purpose of the UML Component Diagram is to show the structural relationships between the *Components* of a system. Components are considered to be autonomous, encapsulated units within a system or subsystem that provide one or more interfaces and are strictly logical. Component Diagrams offer system architects a natural tool for modelling a solution, allowing them to verify that a system's required functionality is being implemented by components, thus ensuring that the resulted system will be acceptable. In addition, Component Diagrams are useful communication tools for various groups (Bell 2004). The four basic elements of a Component Diagram are:

- *Component* (rectangle with small symbol in upper right corner);
- *Provided Interface* (connector with circlet);
- *Required Interface* (connector with arc); and
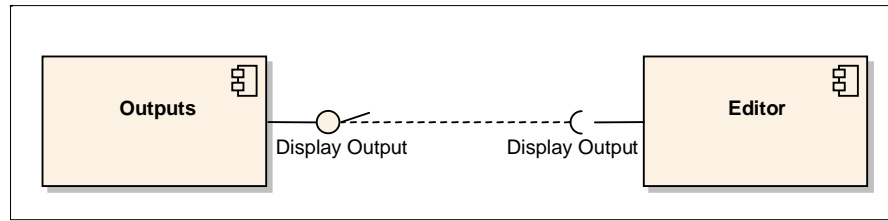- *Dependence* (dashed arrow).

Figure 2. Example of a UML Component Diagram

Figure 2 shows an example of a Component Diagram using the notation of UML 2 (Object Management Group, 2005). This Diagram comprises two components: *Outputs* and *Editor*. The component *Outputs* provides an interface called *Display Output* (a *provided interface*), which provides functionality necessary for displaying outputs on a computer screen. The component *Editor* uses this interface (a *required interface*).

5. UML Component Diagram of SDI

The proposed Component Diagram for modelling an SDI from a Computation Viewpoint is shown in Figure 3. The Commission proposes that an SDI is composed of six service components based largely on RM ODP:

- Registry Services
- Data Services
- Processing Services
- Application Services
- Portrayal Services
- Management Services.

Every component offers some functionality, modelled by the provided interfaces, and uses functionality offered by other Components modelled by the required interfaces. Each Component is discussed in more detail in the following sections.

5.1 SDI Registry Services

The main purpose of the component *Registry Services* is to *Register* other services, to *Publish* them, and later to *Search* through them (by the users). These three functionalities are provided by three interfaces with the Registry Services.

The only interface required by this component, as illustrated in Figure 3, is the *Control* interface, which is provided by the *Management Services* component. Description of these three interfaces is provided below.
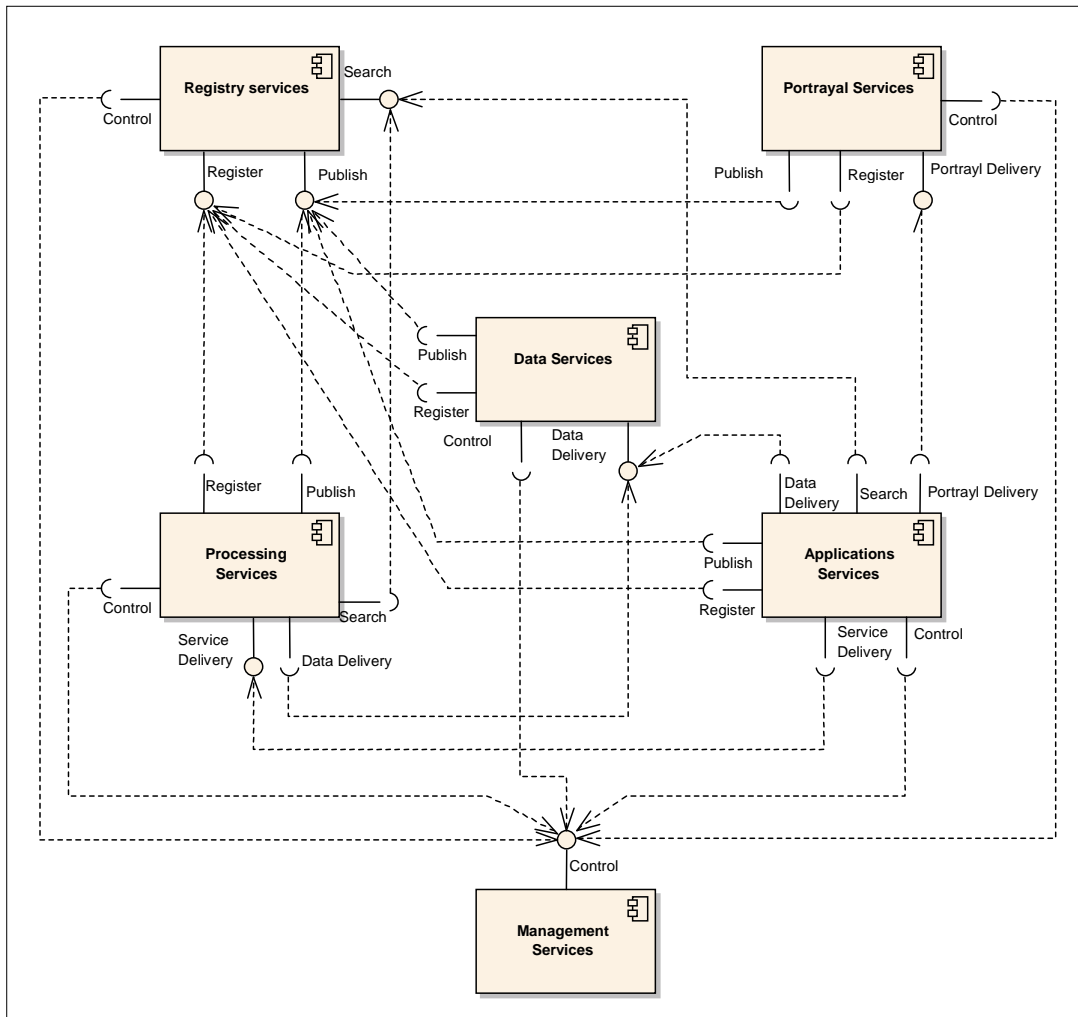
Figure 3. Computational View of Service Component Diagram of SDI

5.1.1 SDI Interface Register

This interface normally provides the necessary functions to *Register* information about resources available on a network, such as:

- Register Product Specification
- Register Product
- Register Metadata
- Register Catalogue

- Re-Register Catalogue
- Register Policy
- Register Business Plan
- Update Register.

5.1.2 SDI Interface Search

The functionality provided by this interface facilitates searching for the required data, service or catalogue. The interface normally includes the following functions:

- Search Register
- Search Product Specification
- Search Product
- Search Metadata
- Search Catalogue
- Search through Catalogue.

5.1.3 SDI Interface Publish

The interface *Publish* provides functionality necessary for publishing the output information from *Registers* through the Internet or other media. For example, the interface usually includes functions such as:

- Publish Product Specification
- Publish Product
- Publish Metadata
- Publish Catalogue

5.2 SDI Data Services

The *Data Services* component in Figure 3 deals with data sets shared and registered on the Internet. For example, *Data Services* provide access to collections of data in repositories and databases. The only interface provided here is *Data Delivery*, which is designed in such a way that it will provide data to users via *Processing Services* or *Application Services*. No other component deals with data sets directly. This component requires a number of interfaces, provided by other components. For example, the following *Service Components :: Interfaces* constructs (where :: means *Interfaces*) are examples of Service Components:

- Application Services:: Access
- Registry Services :: Register
- Registry Services :: Publish
- Management Services :: Control

5.3 SDI Processing Services

The *Processing Services* component, as shown in Figure 3, contains the services for data processing. For example, the processes of coordinate computation or projection system transformation, can be accomplished through this service component. It provides only one interface, *Service Delivery*. However, this component requires a number of interfaces provided by other components, such as:

- Registry Services :: Register
- Registry Services :: Publish
- Registry Services :: Search
- Data Services :: Data Delivery
- Management Services :: Control.

5.4 SDI Application Services

The *Application Services* component is a key part of the architecture, as shown in Figure 3. It includes all processes necessary to meet the needs of users. Hence, it does not provide any interface, but requires a large number of interfaces provided by other components:

- Registry Services :: Register
- Registry Services :: Publish
- Registry Services :: Search
- Data Services :: Data Delivery
- Processing Services :: Service Delivery
- Portrayal Services :: Portrayal Delivery
- Management Services :: Control.

5.5 SDI Portrayal Services

The *Portrayal Services* component in Figure 3 deals with displaying the results of application services. It provides one interface for these purposes, which is *Portrayal Delivery*. It requires the following interfaces from other components:

- Registry Services :: Register
- Registry Services :: Publish
- Management Services :: Control.

and also should provide functionallity which facilitates the output, such as:

- Design Layout
- Editing Function
- Delivery Options
- Formats

5.6 SDI Management Services

The *Management Services* component in the lower part of Figure 3 monitors the overall functionality of the SDI. For this purpose it has one interface, which is *Control*. For example, this component controls the interoperability amongst services or rights of access.

6. Summary & Conclusions

The ICA's Commission on Spatial Data Standards has been developing formal models of SDIs, using the RM-ODP Viewpoints. Presented here are the first models of the RM-ODP Computation Viewpoint of an SDI, modelled using UML.

7. Acknowledgements

9. References

Aalders HJGL & Moellering, H (2001), "Spatial Data infrastructure", Proceedings of the 20th International Cartographic Conference, Beijing, China, Vol. 4, pp. 2234-2244.

Bell D (2004), UML basics: The component diagram, IBM Global Services, IBM. http://www-128.ibm.com/developerworks/rational/library/dec04/bell/. (14.10.2006).

Cooper AK, Hjelmager J, Nielsen A & Rapant P, (2003), "A Description of Spatial Data Infrastructures (SDIs), Using the Unified Modelling Language (UML)", Proceedings of the 21st International Cartographic Conference, Durban S. Africa, CD-ROM.

Delgado T (2005), "Infraestructuras de Datos Espaciales en países de bajo desarrollo económico. Implementación en Cuba", Tesis de doctorado, ITM "José Martí", 143 pp.

Groot R & McLaughlin J, (2000), "Geospatial Data Infrastructure: Concepts, Cases and Good Practices", Oxford: Oxford University Press. ISBN 0198233817, 286pp.

Hjelmager J, Delgado T, Moellering H, Cooper AK, Danko D, Huet M, Aalders HJGL & Martynenko A, (2005), "Developing a Modeling for the Spatial Data Infrastructure", Proceedings of the 22nd International Cartographic Conference, A Coruña, Spain. CD-ROM.

Hjelmager J, Moellering H, Delgado T, Cooper AK, Rajabifard A, Rapant P, Danko D, Huet M, Laurent D, Aalders HJGL, Iwaniak A, Abad P, Düren U & Martynenko A, (2007), "An initial Formal Model for Spatial Data Infrastructures", International Journal of GIS, (In Press).

ISO/IEC 10746 (1995), Information technology – Open Distributed Processing – Reference model.

Masser I, (2005), "GIS Worlds - Creating Spatial Data Infrastructures", ESRI Press, ISBN 1589481224, 312 pp.

Object Management Group (2005), Unified Modeling Language (UML), version 2.0, accessed 2007-01-26 from: http://www.omg.org/technology/documents/formal/uml.htm

Rajabifard A, Binns A, Masser I & Williamson IP, (2006), "The Role of Sub-national Government and the Private Sector in Future SDIs", Vol 20, No 7, International Journal of GIS, 727-741.

Vallecillo A, (2000), "RM-ODP: The ISO Reference Model for Open Distributed Processing", Num especial de la revista de DINTEL sobre normativa en Ingeniería del Software, May (2000). http://www.lcc.uma.es/~av/Publicaciones/00/odpeng.pdf