Advances in Geographic Information Science

Series Editors:

Shivanand Balram, Canada
Suzana Dragicevic, Canada

G. Brent Hall · Michael G. Leahy (Eds.)

Open Source Approaches in Spatial Data Handling

Springer

G. Brent Hall
University of Otago
School of Surveying
Dunedin
New Zealand
brent.hall@otago.ac.nz

Michael G. Leahy
Wilfrid Laurier University
Waterloo, Ontario
Canada N21 3C5
mgleahy@alumni.uwaterloo.ca

ISBN: 978-3-540-74830-4 e-ISBN: 978-3-540-74831-1

Advances in Geographic Information Science ISSN: 1867-2434

Library of Congress Control Number: 2008932589

Cover design: deblik, Berlin

Printed on acid-free paper

9 8 7 6 5 4 3 2 1

springer.com

Preface

During the last several years there has been a significant coalescence of interest in Open Source Geospatial (OSG) or, as it is also known and referred to in this book, Free and Open Source for Geospatial (FOSS4G) software technology. This interest has served to fan embers from pre-existing FOSS4G efforts, that were focused on both standalone desktop geographic information systems (GIS), such as GRASS, libraries of geospatial utilities, such as GDAL, and Web-based mapping applications, such as MapServer. The impetus for the coalescence of disparate and independent project-based efforts was the formal incorporation on February 27th, 2006 of a non-profit organization known as the Open Source Geospatial Foundation (OSGeo). Full details concerning the foundation, including its mission statement, goals, evolving governance structure, approved projects, Board of Directors, journal, and much other useful information are available through the Foundation's website (http://www.osgeo.org).

This book is not about OSGeo, yet it is difficult to produce a text on FOSS4G approaches to spatial data handling without, in some way or another, encountering the activities and personalities of OSGeo. Of the current books published on this topic the majority are written by authors with very close connections to OSGeo. For example, Tyler Mitchell who is the Executive Director of the Foundation, is author of one of the first books on FOSS4G approaches ('Web Mapping Illustrated' (2005)). Another member of the Board of Directors of the Foundation, Markus Neteler, is co-author of the book 'Open Source GIS: a GRASS approach' (2007), which is in its third edition.

Hence, not coincidentally, the current text has contributions from a number of authors with close connections to the Foundation. The importance of OSGeo in all aspects of FOSS4G development is unquestionable and unparalleled in the development of Open Source software within the spatial data domain. As OSGeo has established itself firmly at the centre of what is described by David McIhagga in Chap. 3 of this book as the Open Source Web mapping ecosystem, FOSS4G technologies and concepts have permeated into many diverse areas of application development. One area where recent interest in the spirit and 'openness' of FOSS4G software is apparent is the tertiary education sector. This interest is likely to increase in the future as the tools, which are already of quite an incredible level of maturity, become better known and ever more widely used, and as curricula evolve to explore opportunities other than those that are tightly coupled with the dominant proprietary GIS software vendors.

The idea for this book evolved from working with FOSS4G tools on projects within an academic environment. Other than the two texts referred to above and one or two others, prior to the appearance of OSGeo the dominant reference source for FOSS4G projects was and substantially still is the Internet. This is perhaps the way it should be as, with projects that are in a constant state of evolution, the use of the printed word is inevitably associated with a limited shelf-life. This is especially true for texts that are 'cook book' oriented, containing instructions on how to do this or that with a specific software package. These sorts of texts are essentially only relevant to the versions of software that they relate to, yet there is a huge international market for them. Hence, the approach sought and largely implemented with this book was not to provide readers with information on how to use a specific FOSS4G tool or project, but rather to focus on several projects more from a conceptual rather than a 'how to' point of view. The purpose is to introduce readers new to FOSS4G software to the nature, purpose, evolution and characteristics of a number of projects, while also discussing important issues such as the role of standards in OS software development, the business models that can allow 'free' software development to sustain the developers, and the general need for a spirit of co-operation and partnership building that is often absent from the closed software marketplace.

The book is generally divided into three sections. The first three chapters focus on the topics noted immediately above. New business models have had to be created or have evolved to promote and sustain core FOSS4G projects, and companies, such as DM Solutions Ltd., Refractions Research and others that have grown on the back of FOSS4G inspired ideas. Hence, market niches have been identified that allow these commercial entities to provide FOSS4G services and FOSS4G solutions that remain freely accessible to any and all who are interested, while remaining commercially viable. Equally, issues such as the use of standards, improving documentation, making tools more accessible to end-users who are not programmers, improving FOSS4G interoperability, co-ordinating collaboration within the developer community, and controlling code release frequency through use of best practice management standards are all now substantially more important than beforehand, with the groundswell of support for and involvement in project development. Respectively, these three initial chapters are written by Tom Kralidis on standards, Arnulf Christl on new business models, and David McIhagga on what he aptly describes as the Open Source Geospatial Web 'ecosystem'.

The second section of the text comprises the majority of the chapters. In this section focus is given to a variety of key FOSS4G projects. For the most part the authors of these chapters are individuals who, for the most part, have been either the catalyst of the project or have played a prominent role in its development. Each chapter is generally built around a discussion of the objectives of the project, the architecture of the tool(s), how the project evolved to satisfy its initial objectives, or alternatively how the objectives morphed as the project unfolded into its current state. There is some technical discussion in these chapters, however the intention of the text was not to produce a manuscript shrouded in technical language. While this is inevitable to some extent with technical subject matter, the intention, as noted above, was to make the book accessible to those new to FOSS4G, while also providing information of interest to established members of the FOSS4G community.

The chapters are illustrated to varying extents, some richly and some not at all, with design diagrams and screen captures. Clearly, it was not possible to cover all of the core or key FOSS4G projects that have evolved, but the chapter selection does a fair job at spanning the field. In fact, chapters discuss five of OSGeo's 13 established Web mapping, desktop, or geospatial library projects. Respectively, Chap. 4 is authored by Steve Lime and focuses in MapServer, perhaps the most successful Web-based mapping tool yet developed. Chapter 5, by Frank Warmerdam, discusses the Geospatial Data Abstraction Library (GDAL), which also is possibly the most successful such library to have yet been produced. The 6th Chapter is by Rongguo Chen and Jiong Xie and it deals with Open Source databases and their spatial extensions, most notably PostGIS, produced by Refractions Research. Chapter 7 is by Robert Bray and it is important as it deals with MapGuide Web mapping software, by Autodesk Inc., which was converted to OS in November 2005. The software was contributed to OSGeo in March 2006 as a foundation project. Chapter 8 is by Ian Turton and focuses on GeoTools, which is Java-based and is used as the base for several other well known FOSS4G projects. Chapter 9 is by Markus Neteler and his colleagues from the GRASS development team, and it discusses probably the oldest and most firmly established FOSS4G desktop application available.

The third section of the book comprises the same number of chapters as the first. These chapters discuss applications of some of the tools described in section two, with the addition of the impressive array of FOSS4G libraries and applications packages developed by Gilberto Camara and his colleagues from various institutions in Brazil. Specifically, Chap. 10 discusses one of two University-based FOSS4G projects reviewed in the book, namely GeoVISTA *Studio* developed by Mark Gahegan and his colleagues at Penn State University. Chapter 11, by the Editors, discusses a second University-based tool which utilizes several of the other FOSS4G projects discussed in the text.

Compiling these chapters was an interesting exercise. It became very clear very quickly, and remained clear throughout the project, that the one element that is in very short supply in FOSS4G software development in general is free time. Hence, for several of the contributors finding time to complete their chapter proved to be difficult. In addition, it seems also that outstanding programmers enjoy doing what they do best, but do not necessarily enjoy writing about it! Given this, we would like to thank the contributors for their forbearance in my persistent 'nagging', which was required in order to get the book finished. In the final analysis, the chapters together weave a very useful tapestry of activities within this general field.

It would be remiss of us not to complete this preface without noting thanks to a number of individuals who have helped along the way. First, we would like to thank the series editors, Drs. Shivanand Balram and Suzana Dragicevic from Simon Fraser University, British Columbia, Canada, who liked the initial idea of doing a book on this theme. Chris Bendall, the editor from Springer was also very helpful in moving the idea into a reality. We would like to thank our colleagues at the University of Waterloo, Ontario, Canada, where the bulk of the editorial work for this book was done. In particular, we wish to thank Dr. Rob Feick, as well as the following graduate students who worked on the MapChat project, namely David Findlay, Taylor Nicholls, John Taranu and Brad Noble. Last but certainly not least we thank our wifes Masha and Ally, whom we dedicate this effort to.

Dunedin, New Zealand *G. Brent Hall*
Michael G. Leahy

References

Mitchell T (2005) ‘Web mapping illustrated’, O’Reilly Media Inc., Sebastopol, Ca.
Neteler M, Helena M (2008) ‘Open source GIS: A GRASS GIS approach’, 3rd edn, Springer, New York

Contents

List of Contributors

D.E. Beaudette
Department of Land, Air and Water Resources, University of California, Davis, CA 95616, USA, e-mail: debeaudette@ucdavis.edu

Robert Bray
Autodesk, Inc., 2100, 645 - 7th Ave SW, Calgary, Alberta T2P 4G8 Canada e-mail: robert.bray@autodesk.com

Gilberto Câmara
National Institute for Space Research (INPE), Av dos Astronautas 1758, 12227-010, São José dos Campos, Brazil, e-mail: gilberto@dpi.inpe.br

Marcelo Tílio de Carvalho
Catholic University of Rio de Janeiro (PUC-RIO), Rua Marquês de São Vicente, 22522. 453-900 Rio de Janeiro/RJ, Brazil, e-mail: tilio@tecgraf.puc-rio.br

Marco Antonio Casanova
Catholic University of Rio de Janeiro (PUC-RIO), Rua Marquês de São Vicente, 22522. 453-900 Rio de Janeiro/RJ, Brazil, e-mail: casanova@tecgraf.puc-rio.br

P. Cavallini
Faunalia, Piazza Garibaldi 5, 56025 Pontedera (PI), Italy,
e-mail: cavallini@faunalia.it

J. Cepicky
Help Service - Remote Sensing s.r.o., Cernoleska 1600, 25601 - Benesov, Czech Republic, e-mail: jachym.cepicky@gmail.com

Rongguo Chen
State Key Laboratory of Resources and Environmental Information System, Chinese Academy of Sciences, Beijing, China, e-mail: chenrg@lreis.ac.cn

Arnulf Christl
WhereGroup GmbH & Co. KG, Siemensstr. 8, 53121 Bonn, Germany; Open Source Geospatial Foundation, Beaverton, OR, USA,
e-mail: arnulf.christl@wheregroup.com, arnulf@osgeo.org

Urška Demšar
National Centre for Geocomputation, National University of Ireland, Ireland,
e-mail: urska.demsar@nuim.ie

Karine Reis Ferreira
National Institute for Space Research (INPE), Av dos Astronautas 1758, 12227-010, São José dos Campos, Brazil, e-mail: karine@dpi.inpe.br

Ubirajara Moura de Freitas
Space Research and Applications Foundation (FUNCATE), Av. Dr. João Guilhermino, 429 – 18th floor 12210-131 São José dos Campos, SP, Brazil,
e-mail: bira@geo.funcate.org.br

Mark Gahegan
Department of Geography, University of Auckland, Private Bag, Auckland, New Zealand, e-mail: mark@geog.psu.edu

G. Brent Hall
School of Surveying, University of Otago, Dunedin, New Zealand,
e-mail: brent.hall@otago.ac.nz

Frank Hardisty
GeoVISTA Center, Department of Geography, Penn State University, Pennsylvania, USA, e-mail: hardisty@psu.edu

Athanasios Tom Kralidis
Open Source Geospatial Foundation, Beaverton, OR, USA,
e-mail: tom.kralidis@gmail.com

L. Lami
Faunalia, Piazza Garibaldi 5, 56025 Pontedera (PI), Italy,
e-mail: lami@faunalia.it

Michael G. Leahy
Department of Geography and Environmental Studies, Wilfrid Laurier University, Waterloo, Ontario, Canada, N2L 3C5, e-mail: mgleahy@alumni.uwaterloo.ca

Stephen Lime
Minnesota Department of Natural Resources, St. Paul, MN, USA,
e-mail: steve.lime@dnr.state.mn.us

David McIhagga
President & CEO, DM Solutions Group, Ottawa, ON, Canada,
e-mail: dmcilhagga@dmsolutions.ca

Antônio Miguel Vieira Monteiro
National Institute for Space Research (INPE), Av dos Astronautas 1758, 12227-010, São José dos Campos, Brazil, e-mail: miguel@dpi.inpe.br

M. Neteler
Fondazione Mach - Centre for Alpine Ecology, 38100 Trento (TN), Italy, e-mail: neteler@osgeo.org

Gilberto Ribeiro de Queiroz
National Institute for Space Research (INPE), Av dos Astronautas 1758, 12227-010, São José dos Campos, Brazil, e-mail: gribeiro@dpi.inpe.br

Ricardo Cartaxo Modesto de Souza
National Institute for Space Research (INPE), Av dos Astronautas 1758, 12227-010, São José dos Campos, Brazil, e-mail: cartaxo@dpi.inpe.br

Masa Takatsuka
ViSLAB, Information Technologies, University of Sydney, Australia, e-mail: masa@vislab.net

Ian Turton
GeoVISTA Center, Pennsylvania State University, University Park, PA, 16802, USA, e-mail: ijt1@psu.edu

Lúbia Vinhas
National Institute for Space Research (INPE), Av dos Astronautas 1758, 12227-010, São José dos Campos, Brazil, e-mail: lubia@dpi.inpe.br

Frank Warmerdam
Independent Software Developer, Open Source Geospatial Foundation, Beaverton, OR, USA, e-mail: warmerdam@pobox.com

Jiong Xie
State Key Laboratory of Resources and Environmental Information System, Chinese Academy of Sciences, Beijing, China, e-mail: xiej@lreis.ac.cn

Chapter 1
Geospatial Open Source and Open Standards Convergences

Athanasios Tom Kralidis

Abstract Geospatial information has become a ubiquitous resource. The proliferation of the Internet and information technology has resulted in an enormous volume of information exchange and a growing global geospatial data infrastructure presence. Interoperability is increasingly becoming a focus point for organizations that distribute and share data. Standards are an essential aspect of achieving interoperability. This chapter illustrates the benefits of using open standards for geospatial information processing. It also discusses various free and open source for geospatial software (FOSS4G) packages that support open standards. Finally, the chapter illustrates how open source software and open standards can be easily integrated in a number of scenarios.

1.1 Introduction

Laying the groundwork to establish a framework for the interoperability of spatial data has been an ongoing activity for at least three decades. The 1970s saw the emergence of a growing requirement for national mapping and surveying agencies to create policies, agreements and processes for normalizing access to and applications of spatial data. In Canada, for example, the origins of a spatial data infrastructure emerged in the 1980s as an effective means of facilitating data access (Groot and McLaughlin 2000). Following from this there has been an ongoing effort worldwide to produce standards-based specifications for the discovery, evaluation, access, visualization and exploitation of spatial data resources (Global Spatial Data Infrastructure Association 2001).

This chapter discusses the concept of interoperability, the roles and activities of open standards bodies and organizations, and provides examples of free and open source software for geospatial (FOSS4G) projects which exemplify standards-based approaches to spatial information exchange and processing.

Athanasios Tom Kralidis
Open Source Geospatial Foundation, Beaverton, OR, USA, e-mail: tom.kralidis@gmail.com

G.B. Hall, M.G. Leahy (eds.), *Open Source Approaches in Spatial Data Handling*.

1.1.1 *Geographic Information*

Over the last three decades, governments and industry have invested billions of dollars in the development of geographic information systems (GIS) to serve various information communities including forestry, marine studies, disaster management, natural resources, health, and numerous others (Groot and McLaughlin 2000). The information collected by organizations from within these communities has the potential for multiple uses and sharing between users, activities, systems and applications. Despite significant decreases in the cost of computer hardware and software over time, spatial data are ever more voluminous and an expensive resource to develop and maintain. One means that has become popular to organize spatial data resources is the concept of a spatial data infrastructure.

1.2 Spatial Data Infrastructure

This section discusses the concept of a spatial data infrastructure (SDI) by focusing on digital networks and the Internet as the foundation global data infrastructure and discusses how SDIs leverage the presence of the Internet to establish data sharing and data exchange mechanisms. This discussion also illustrates how the concept of interoperability drives the functionality of an infrastructure and is a core requirement for information exchange of any kind.

1.2.1 *The Internet and the Digital Age*

A data infrastructure can be defined as a transparent, robust computer environment, which enables access to information using common, well-known and accepted specifications, standards and protocols (Global Spatial Data Infrastructure Association 2001). To use a simple analogy, a telephone network can be thought of as an interoperable infrastructure, in that it provides users with connectivity and services to communicate with each other, while the details behind the communications, including the physical telecommunications infrastructure such as networks, wiring, switches, and exchanges are transparent and relatively unimportant from an end user perspective. Such an infrastructure can be seen as an underlying building block to enable communications by products such as specialized applications, as well as the development of sub-networks to be built and deployed for specific purposes. Although a critical aspect of networking and communications, this form of interoperability is also mundane in its ubiquity. However, the very existence of the infrastructure required to facilitate communications makes enabling objects, technologies, and analysis possible (Harvey 2000).

A data infrastructure is also the result of many nodes around which data and services are decentralized. This process of dispersion of data and service points on a densifying network is similar to the process that organizations go through when they are restructuring from single site to multi-site enterprises. For example, in the 1980s and 1990s, globalised forms of enterprise organization began to emerge where companies sought to leverage lower costs. To a large extent, this process was facilitated by the growth of network-based forms of enterprise organization as well as information sharing. Though location is an important consideration in this process, the decentralization of economic activity has shown that location is not as important in the context of doing business as it once was. This principle also applies to infrastructures that gather, process, and disseminate geographic information. Organizations can collect and maintain their own data holdings, and publish them through clearinghouses for use by end users and/or an ever widening variety of data services. In this process data are kept closest to the source of production or collection to facilitate their update and completeness, but equally they may become widely dispersed geographically in their use and potentially also in their enhancement.

A SDI extends the data infrastructure concept by focusing on the transport and transmission of spatial information and in providing the relevant technologies, policies and agreements that assist in the availability of and access to spatial data resources. A SDI provides the architectural underpinnings for the discovery, evaluation and application of spatial information (Global Spatial Data Infrastructure Association 2001). In this context, a concerted effort is currently being made among government agencies, both within and between countries, to enable the discovery, visualization and access of spatial information at all levels by leveraging the Internet as the distributed infrastructural backbone for interaction with spatial data (Canadian Geospatial Data Infrastructure Architecture Working Group 2001; Global Spatial Data Infrastructure Association 2001). Examples of organizations supporting such activities include the United States Government's Federal Geographic Data Committee (FGDC – Federal Geographic Data Committee 2004), the National Aeronautics and Space Administration (NASA – National Aeronautics and Space Administration 2004) and Canada's GeoConnections program (GeoConnections 2007). Other examples include the Australian and New Zealand Land Information Council (ANZLIC – Australian and New Zealand Land Information Council 2007), and the Infrastructure for Spatial Information in Europe (INSPIRE – Inspire 2007).

Using the infrastructure approach to geographic information management, organizations can interact with spatial data over the Internet in a transparent fashion. A SDI reduces the requirements for multiple standards by establishing a unified approach to data syntax, semantics and schemas as well as content management. Moreover, a SDI encompasses networked spatial databases and allows efficient management of complex organizational, technical, human and economic components, all of which interact with one another. Without this centralized and unified approach to management, the cost and feasibility of multiple copies of spatial information quickly become unmanageable, especially if current data are an important requirement for an enterprise. Locally stored copies of data result in large, ongoing data management problems for organizations. In addition they are costly to sustain, and are prone to concurrency issues with multiple versions of the same data set existing within and between the same and different organizations.

Computer networks are vital to a SDI (Groot and McLaughlin 2000). Most distributed network systems are made up of client-server architectures, where one or more central servers provides services and information to client via an information exchange mechanism, which is vital to the infrastructure the network supports. The Internet, which is built upon this model of computing, originated from the United States Department of Defense Advanced Research Projects Agency Network (ARPANET) in the late 1960s (Begg and Connolly 1998). Its current popularity has reinforced and generated numerous diverse information highways in many information communities (Hartman 1997). Computers now leverage network technology to share disk drives, memory, and/or data. The Internet and the TCP/IP reference model, provides a means for transporting information "packets", providing a framework (network addressing, fragmentation, timeouts, and so on) for peer-to-peer communication through TCP, and enabling an application layer for user-level protocols, such as File Transfer Protocol (FTP) and Hypertext Transfer Protocol (HTTP) (Groot and McLaughlin 2000).

The Internet enables information holdings and services to be distributed in terms of their location. Based on open communication standards and protocols, this has enabled organizations to publish information over a distributed network infrastructure, as well as provide a medium for discovering and evaluating educational resources, commercial initiatives and government information among other things. The common standards allow computers to connect to the backbone network and to each other despite differences in hardware, software and other factors that have historically impeded communications (Hartman 1997). This results in a foundation layer of interoperability in network communications.

1.2.2 Interoperability

Interoperability can be defined as the ability of a system or components of a system to provide information sharing and inter-application process control, through a mutual understanding of request and response mechanisms (Groot and McLaughlin 2000). Interoperability is the ability of a system (or a component of a system) to access a variety of heterogeneous resources by means of a single, unchanging operational interface (Canadian Geospatial Data Infrastructure Architecture Working Group 2001). That is, two resources (such as a client and a server) are interoperable if there is a mutually agreed upon and standardized messaging vocabulary which they can understand. While communications may relay different requests and responses, the two resources understand the frameworks in which they are delivered.

The concept and practice of interoperability dovetails with the open systems model, which is an approach to software engineering and system design that enables and encourages the sharing of resources (Gardels 1999; Guerrero 2004). These resources are regarded as objects, meaning that every resource can be seen as a component among other components which coexist under a common framework, thus promoting an operational model as opposed to data standards (Gardels 1999). Such common frameworks benefit from publically available and agreed upon methods and practices for information processing within and between various communities and networks.

1.3 Open Standards

This section provides an overview of open standards to support information processing in the geospatial domain, from what they are, to identifying key formal standards bodies as well as ad hoc and de facto standards groups. Current adoption and organizational benefits of using standards are also discussed to illustrate the significant influence open standards have had and will continue to have in the FOSS4G arena. Though there are numerous standards and standards bodies, those with particular relevance to the FOSS4G community are the focus of interest.

1.3.1 Overview

The ubiquity of geospatial information results in massive information repositories maintained by mapping and surveying organizations that publish content to SDIs. As suggested earlier, the Internet has had an enormous impact in enabling the discovery, access and visualization of spatial data, for both information providers and consumers alike. The Internet has provided the ability to integrate data holdings, and provides a transparent layer to the end user to interact with spatial data resources (Begg and Connolly 1998). With the advances in computer technology and standards, SDI activities increasingly provide an opportunity for the cost-effective collection, sharing and distribution of information with a geographic component within and between user communities (Groot and McLaughlin 2000). With the volume of spatial data being produced and published to the Internet ever increasing, issues emerge with regard to usability and suitability. For example, the following questions, among others, must be answered adequately:

- Are the spatial data posted on the Internet in a format or structured in a way in which those wishing to utilize the data can comprehend and interpret them relative to application and analytic needs?
- Do the data originate from an authoritative and reliable source or provider?
- Are the data representative of the most current updates and maintenance by the authoritative provider?
- Is the consumer looking for an entire data product, or for a specific parcel, region, or band combination of imagery? That is, the consumer may be seeking a subset of a much larger database, but cannot afford to, or may not wish to, acquire the entire data collection.
- Do the data have any security and/or policy issues with regard to their usage?

These questions represent just a few issues with regard to spatial data interoperability within a SDI. The causes of such issues can be due to differing organizational policies and practices, as well as contradictory approaches to information management, technology and data sharing within the spatial data community. In each case it is likely that the lack or ineffective use, of specifications and standards is at the heart of difficulties that are encountered (Groot and McLaughlin 2000).

Harmonizing approaches and standards for spatial data acquisition and exchange lessens the requirement for maintaining multiple versions of the same data, publishing the data, and exchanging data within and between provider and user groups, all of which may become very expensive in terms of resources and operating budgets. The 1970s saw the emergence of a growing requirement for national mapping and surveying agencies to create policies, agreements and processes for normalizing the access to and application of spatial data resources (Groot and McLaughlin 2000). These requirements were initially narrow in scope and have increasingly come to focus on the use of data standards.

A standard can be best defined as a document or collection of documents, usually but not always published, that establish a common language, terminology, accepted practices and levels of performance, as well as technical requirements and specifications, that are used consistently for the development and use of products, services and systems (Yeung and Hall 2007). Standards are multidimensional. That is, they can be defined for data content, values and at various levels of conformance, such as technical specifications, conventions, and guidelines.

Standards initially provide three primary benefits for spatial data and their users, namely portability, which includes use and reuse of information and applications; interoperability, which includes multiple system information exchange; and maintainability, which includes long term updating and effective use of a resource (Groot and McLaughlin 2000). Standards can save time and effort by removing the need for reinventing approaches to discovery, evaluation, access and visualization of spatial data. Standards organizations for SDI are evident at multiple levels, such as government organizations (for example the FGDC), independent bodies, such as the Canadian General Standards Board (CGSB), American National Standards Institute (ANSI), the International Organization for Standardization (ISO), and industry associations, such as the Open Geospatial Consortium (OGC) (Groot and McLaughlin 2000). A SDI supports low-level standards, such as computer hardware, networks and operating systems, as well as high-level standards such as user interfaces, data formats, and presentation views of data (United States National Research Council 1999).

Standards promote interoperability within an infrastructure, and provide significant benefits for information exchange. Standards are designed for broad, long-term use. However, they are not immutable, and may be modified by consensus among users by standards-issuing bodies. This process may occasionally pose difficulties due to the lengthy design and definition process used to create a standard, which initially takes a potentially long process of submission and review iterations before the standard is accepted by the body it relates to. Once the standard is approved, it is up to the relevant organizations or communities of users to utilize its content.

The above mentioned enabling approaches and technologies provide open-ended possibilities for geospatial information. However, they also raise issues of data copyright and intellectual property (IP). Open standards are independent of IP and organizational policies regarding spatial data in that they can be applied to any IP/policy situation for spatial data. The development of a useful legal framework for both private and public activity is vital to the dissemination of spatial data, no matter what standard is used. As noted earlier, geographic information is not cheap to produce and maintain (Aslesen 1998). The capabilities of digital infrastructures and information communities create further concerns over geospatial information and its potential misuse as control over copying data is difficult to implement. In fact, GIS and related technologies can be dangerous in their ability to merge spatial data by identifying details and information that are otherwise not transparent independently.

Hence, it is not surprising that there is a community of interest on matters relating to SDI in general and the role of enabling standards in particular. How does this community communicate in terms of discovering, evaluating, accessing, and visualizing spatial information? How is interoperability prescribed in a SDI, and how does it satisfy the requirements of the community? These questions and other related issues are discussed further below.

1.3.2 Relation to Open Source

In this discussion it is important to distinguish between open standards and open source in order to be aware of differences in the meanings of these terms, as they are sometimes used interchangeably. An open standard (i.e. a standard that is publicly available to use) can be implemented by open source (i.e. the principles and methodologies to promote open access to design and production) software, as well as commercial or proprietary solutions, in much the same manner. However, open standards are implementation agnostic and are not exclusive to open source software.

1.3.3 World Wide Web Consortium (W3C)

Established in 1994, the World Wide Web Consortium (W3C) develops interoperable technologies (specifications, guidelines, software, and tools) to facilitate minimal levels of conformity in Web standards. The Consortium describes itself as a forum for information, commerce, communication, and collective understanding (World Wide Web Consortium 2007). W3C standards are free to obtain and implement. Core specifications include Hypertext Transfer Protocol (HTTP), the Uniform Resource Locator (URL), and Hypertext Markup Language (HTML), which have become building blocks for other W3C specifications such as the Document Object Model (DOM), Extensible Markup Language (XML), Extensible Stylesheet

Language Transformations (XSLT), Scaleable Vector Graphics (SVG), and Cascading Stylesheets (CSS). More information on these specifications is available at http://www.w3.org/.

1.3.4 The International Organization for Standardization (ISO)

The ISO is an international standard-setting body composed of representatives from various national standards bodies. ISO was founded in 1947 to produce world-wide industrial and commercial standards (International Organization for Standardization 2007a). Many ISO standards become nationally endorsed, and are heavily used and implemented for a variety of areas, such as the ISO 9000 series of quality management standards (International Organization for Standardization 2007a).

Within the ISO there are numerous technical committees (TCs). TC211 is the committee responsible for standards pertaining to digital geographic information (International Organization for Standardization 2007b). TC211 produces a number of abstract specifications and reference models. These specifications are typically used as building blocks for other standards to leverage, such as those from the OGC. At the time of writing, current implementation specifications of interest include 19115, which covers digital geospatial metadata (International Organization for Standardization 2007c), and 19138, which is the implementation standard of 19115 (International Organization for Standardization 2007d).

1.3.5 The Open Geospatial Consortium (OGC)

The OGC was founded in 1994 as the OpenGIS Consortium. It is a non-profit, international, voluntary consensus standards organization specializing in geospatial data and Web services. The OGC consists of over 250 organizations from government, academia, industry and other groups. The Consortium was founded on the concept of providing open specifications at no cost to the public to acquire and/or implement, thus providing standards-based interfaces for geospatial discovery, access, visualization and processing. The OGC leverages existing efforts from other standards organizations such as the W3C, ISO, and the Organization for the Advancement of Structured Information Standards (OASIS), and builds upon them in reference to the spatial data domain.

The OGC Abstract Specification provides the reference model for implementation of OGC specifications. Areas covered by the Abstract Specification include:

- Feature Geometry
- Spatial Referencing by Coordinates
- Locational Geometry Structures
- Stored Functions and Interpolation
- Features
- The Coverage Type

- Earth Imagery
- Relationships Between Features
- Feature Collections
- Metadata
- The OpenGIS Service Architecture
- Catalog Services
- Semantics and Information Communities
- Image Exploitation Services
- Image Coordinate Transformation Services
- Location Based Mobile Services
- Geospatial Digital Rights Management Reference Model (GeoDRM RM)

The OGC Abstract Specification represents a carefully engineered process and framework in support of the discovery, access and visualization of spatial data. Specifications, discussion papers and recommendation papers are developed from the vision of the Abstract Specification. For example, all OGC Web Services (OWS) provide models for metadata documentation. The Geography Markup Language (GML) specification models spatial features and topological relationships between them. The widely used Web Feature Service (WFS) describes a service-based supply of vector information as feature collections. All OWS follow the Service Architecture Interoperability approach.

The OGC has a strong and progressive specification development process that requires consensus between specific working group members. OGC specifications are typically developed, tested and revised within the OGC testing environments (or "testbeds"), pilot projects, and working group activities. The major benefit of this approach is the iterative process of standards development in concert with specification and software development. Typical specification development takes place by defining, adopting, and publishing the specification document for vendors and others to implement. It is common practice for specifications not to take into account various aspects which may affect ease of software development, functionality and/or usability.

The result of this is often a revision process which can become resource intensive and time inefficient. Vendors may subsequently add "vendor specific" functionality to software, which is where variations begin to surface across vendor implementations of the same specification. In the OGC environment, because the specification is developed with software implementers, this risk is significantly reduced, allowing for specifications to be tested, analyzed, and updated before they reach public adoption. The result is a stronger, more robust version of a given specification and multi-vendor interoperability of software products. This approach to specification development also introduces the unique concept of competing businesses in the spatial information industry working together in a co-operative manner. A full listing of publicly adopted OGC specifications can be found at http://www.opengeospatial.org/standards.

Since the Web Map Service (WMS) was published in 1999 as the first major OGC specification, the OGC has gained a great deal of momentum and credibility in terms of organizational recognition, resulting in many early adopters of geospatial Web services and interoperability. In fact, a survey in 2004 indicated 166 public OGC WMS instances found via the Google search engine. The survey, though not authoritative or scientific, uses Web development to collect and provide reports on OGC usage over the Internet. While the specific searching and interpretation algorithms of the survey remain subject to further interpretation, it is evident that the number of servers indicate a level of maturity and popularity with the OGC and Web service approaches. OGC instances were found in Canada, the United States, Germany, Netherlands, Australia, Italy, Denmark, Czech Republic and Mexico (Ramsey 2004).

The OGC specifications are also making their presence felt in major GIS vendor software packages. This can also be attributed to industry recognition and in response to organizational requirements based on the underlying benefits of interoperability and the Web services approach. In 2007, 381 vendor products either implement or directly conform to OGC specifications (Open Geospatial Consortium 2007a).

Current OGC activities of interest include Sensor Web Enablement (SWE), which involves defining specifications for sensor-based instruments and platforms, Digital Rights Management, and approaches for more complex geoprocessing (Web Processing Service) and linking (Geospatial Linking) (Open Geospatial Consortium 2007b,c). The OGC liaises with ISO through a cooperative agreement where both organizations can leverage and align with one another's developments while satisfying organizational requirements (International Organization for Standardization 2007e), and recognizing and leveraging other standards groups such as the W3C.

In addition to the above specifications, Keyhole Markup Language (KML) Version 2.2 was adopted as an OGC standard in April 2008 (for more information see http://www.opengeospatial.org/standards/kml). This format allows for visualization in applications such as Google Earth, Google Maps and Google Maps for Mobile (Google 2007). KML is an XML-based grammar which has gained a great deal of popularity and is now used in many applications that run over the Internet supporting data styling and referencing in a single document.

1.3.6 De Facto and Ad Hoc Standards

In addition to formal standards bodies and specification programs, there exist numerous de facto standards, which are illustrative of being developed in a relatively informal setting (mailing lists, wikis, forums, etc.), and mature (somewhat organically) to become so popular that they are followed as if they were formal standards.

Some mainstream Web examples include JavaScript, which was originally developed by Netscape, and subsequently became ECMAScript in DOM 1 and 2 HTML (Flanagan 2006). Perhaps the most popular example of this in the spatial data domain is the shapefile (Environmental Systems Research Institute 1998). Since the inception of this format in ArcView software, most GIS packages have created support to read and write this relatively simple form of spatial data encoding. Other de facto standards include the following developments:

- GeoRSS – This extends Really Simple Syndication (RSS) feeds with location information. GeoRSS has become very popular and is implemented by major applications such as Google Maps, Yahoo Maps, and a variety of other implementations. GeoRSS has since been released as an OGC White Paper (Open Geospatial Consortium 2006)
- GeoJSON – Extends JavaScript Object Notation (JSON) to encode objects with location information expressing a variety of geographic data structures (JSON 2007). JSON is a widely used approach for working with data in rich, Web 2.0 style applications, which can instantiate JavaScript objects directly without transforming them from an intermediary format
- Tiled Maps – This approach replaces arbitrary resolution Web mapping approaches with "tiled" maps, which can be managed in an underlying cache mechanism by software (Open Source Geospatial Foundation 2006a,b,c). This approach has allowed the development of rich Web mapping applications that are very responsive relative to the earlier means of rendering these maps in Web-based mapping. Efforts have resulted in a de facto Tiled Map Service Specification (TMSS), and WMS Tiling Client Recommendation (TCR).

It is evident that there are many standards bodies and options that deal directly or indirectly with various aspects of spatial data. It is important to understand the roles of these bodies (e.g., W3C as pure Web-oriented, OGC as spatial data-oriented) to assess what standard is best suited for a given requirement. It is also encouraging that standards bodies have been and are increasingly working with one another so as not to duplicate effort and maximize leverage of accepted approaches.

1.3.7 Current Adoption and Organizational Benefits

It is evident that open standards are making an impact in the geospatial community. As previously mentioned, OGC specifications are supported across numerous software packages (both desktop and Web-based). De facto geospatial standards such as the shapefile and GeoRSS have also gained much attention and are being adopted by the mainstream IT community.

Extended collaboration and partnerships using open standards provide organizations with the opportunity to create open interfaces and communication mechanisms for distributed computing. In the absence of open standards, application client software packages are "bound" to the interfaces and operations as prescribed by the organization or service provider. The result of this is that whenever some aspect of business logic or process is modified at the service level, clients must align with those changes to ensure the same level of service and information is maintained.

Using open standards also lowers the barrier to integration. That is, well known standards can foster the development and use of common tools and technologies, which can act as building blocks for developers. For example, a Java developer with spatial data requirements in his/her existing application can leverage a package like

GeoTools (GeoTools 2007) instead of implementing something from scratch, allowing more time for including other resources in their business domain requirements.

1.4 Open Source Standards-Based Examples

There already exist many open source tools and technologies which implement and/or conform to open standards. This section outlines several of the FOSS4G packages from various parts of the value chain (e.g., servers, clients, databases), and discusses how the existence of open standards have benefited the development and maintenance of these software packages.

1.4.1 MapServer

MapServer is illustrative of a FOSS4G software package which heavily implements open standards. Originally developed as a C language Web mapping engine with a common gateway interface (CGI), MapServer also provides a scripting environment with bindings to popular scripting languages (perl, php, python, ruby, java, c#, tcl, etc.) for easier integration into scripting environments.

MapServer initially supported OGC WMS version 1.0.0 in 2001 (Regents of the University of Minnesota 2001). At the time of writing, OGC support includes WMS 1.1.0 and 1.1.1, WFS, GML, Web Coverage Service (WCS), Sensor Observation Service (SOS), Web Map Context (WMC), Styled Layer Descriptor (SLD), Filter Encoding Specification (FES), and OWS Common.

As standards emerge and evolve, MapServer continues to respond to requirements for open standards support and implementation, which exemplifies convergence between open standards and open source. As a result, MapServer software can communicate with any other (open source or commercial) software package via standards based interfaces and encodings, as either a publisher or consumer in a client-server scenario. More in depth information on MapServer can be found in Chap. 4 of this book and at the MapServer Web site (Regents of the University of Minnesota 2007).

1.4.2 Community Mapbuilder

Community Mapbuilder is a powerful, Web 2.0 style, standards-compliant geographic mapping client which runs in a Web browser. Mapbuilder is a pure browser-based solution. That is, all code operates on a Web browser client (such as Firefox, Internet Explorer) as HTML and JavaScript. No special plug-ins or browser extensions are required, hence Mapbuilder can be classified as a "thin client" or AJAX (Adaptive Path 2005) solution.

Mapbuilder is a strong proponent of open standards, using XSLT as the core XML processing functionality, and using WMC (hence WMS) at the core of its mapping display capabilities. It also supports WFS, WFS-T, GeoRSS, SLD, GML and more. Heavy use and implementation of standards are a main reason of Mapbuilder's progressive development, which gained momentum in 2004 (Community Mapbuilder 2007).

Despite this promise, and perhaps indicating the general nature of OS development, the Mapbuilder Steering Committee announced at the end of July 2008 that after the release of Version 1.5 of the software it would be retired and there would be no planned enhancements to it. The rationale for this relates to the growth of what is described in Chap. 3 as the Web mapping ecosystem. Specifically, the Project Steering Committee refers to the growth of interest in the Openlayers project (http://openlayers.org) and the resulting diversion of users and developers from Community Mapbuilder to Openlayers. Despite this, and the formal end of life of the project, the beauty of OS means that the code is still alive and that independent developers can continue enhancing it for as long as they like.

1.4.3 PostGIS

PostGIS provides support for geographic objects to extend the PostgreSQL object-relational database software package (see Chap. 6 of this book). PostGIS is a C-based spatial engine which implements the OGC SFS specification (Refractions Research 2005). The benefits of SFS allow for standards-based support of spatial processing functions, such as whether two geometries overlap, are within one another, and so on. Also valuable is the support for input and output of the OGC's Well Known Text (WKT) format, as shown later in this chapter.

As PostGIS can serve as a backend to other FOSS4G packages such as MapServer, GeoServer, uDig, and GRASS among others, the OGC support is further exposed to calling codebases. For example, a C developer connecting to the PostgreSQL C API can make direct SQL calls using PostGIS asgml() functionality to retrieve spatial data encoded as GML.

1.4.4 Others

Numerous other geospatial software packages support open standards. For example, GeoServer is a Java-based geographic server which supports WMS and WFS, as well as WFS-T. uDig is a desktop Internet-enabled GIS which supports WMS, WFS, and other related standards. Geonetwork Opensource supports the Catalogue Services specification as well as ISO 19139 for metadata. Degree is a Java-based geographic server which supports WMS, WFS, WCS as well as other standards.

The existence and availability of open standards has enabled the tools mentioned above to be developed with adherence to internationally accepted approaches for online geospatial information exchange. This eliminates the need for resources in producing custom formats or APIs. As a result, these tools can communicate with any other tools which support standards, whether they are proprietary or open source, in a fairly transparent fashion.

1.5 Integration

There are numerous methods in which the combination of open standards and open source software can communicate. Two examples are shown in the following discussion, first within a given codebase and second as part of a distributed service-oriented architecture (SOA).

1.5.1 With FOSS4G Software

Using MapServer's Python MapScript, a developer can utilize the features built into the MapServer C API. MySQL supports spatial extensions via the OGC SFS specification. This means that spatial data fields in a MySQL database can be tested for various spatial predicates, as well as basic input and output/display. The following simple example shows how, using WKT, MapServer can re-project a coordinate from a MySQL database:

```
#!/usr/bin/python

import MySQLdb
import mapscript

projInObj = mapscript.projectionObj("init=epsg:4326")
projOutObj = mapscript.projectionObj("init=epsg:26918")
db = MySQLdb.connect(host="localhost",user="foo", passwd="bar",
db="mydb")

cursor = db.cursor()
cursor.execute("SELECT AsText(geo) FROM locations")
result = cursor.fetchall()

for record in result:

shape = mapscript.shapeObj.fromWKT(record[1])
shape.project(projInObj, projOutObj)
print shape.toWKT()
```

The example could have easily been implemented using MapScript in Perl or PHP, or any development library which is aware of WKT. Similarly, a PostGIS-enabled PostgreSQL database could have been serving the spatial data. This emphasizes how adherence to standards can make such data manipulations much easier to develop than would be the case otherwise.

1.5.2 As Components

Open source software can also leverage open standards as part of a larger information infrastructure. Consider, for example, a Web mapping application that integrates datasets through Web services. The client's connection code consists of a single approach to interact with any WMS or WFS server. The same approach is used for each layer requested. This results in leaner software codebases, given the abstraction and uniformity that open standards provide. That is, no matter what software is used, standards facilitate the client-server interactions. Figure 1.1 displays, at a very high level, this concept linking various GIS software (open source in shaded boxes, commercial in clear boxes). For example, if the MapServer WMS were changed to an Intergraph WMS server package, this would cause no disruption in the operation of the infrastructure because Intergraph and MapServer both support the WMS specification.

Hence, open standards ensure that open source projects can interoperate with each other, as well as with commercial packages, resulting again in "loosely coupled" infrastructures, based on a service-oriented architecture approach.

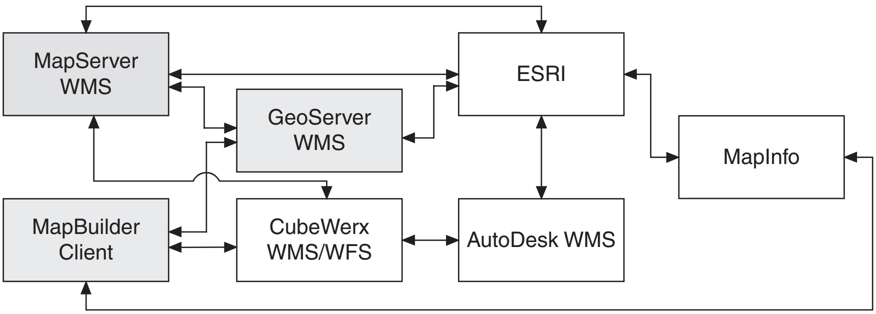

Fig. 1.1 Conceptual intregration of open source software as building blocks for a broader SDI

1.5.3 Example

Consider a scenario that uses MapServer to generate a WMS of earthquake information. The first step in "OGC-enabling" these data is to fetch and reformat the text records into an interoperable, self-describing format. Keep in mind that if this Web service existed, this step would not be required. The initial process is to establish the data format within the data located at http://neic.usgs.gov/neis/finger/quake.asc.

As these data have a geographic location as well as attribute information relative to a series of points on the earth's surface, a GML approach may be used. The primary step in creating a GML document is to create a GML application schema. This application schema defines the data types, structures and objects in W3C XML schema language. Because GML represents an enabling framework, which itself leverages XML schema, a domain expert can easily construct their information

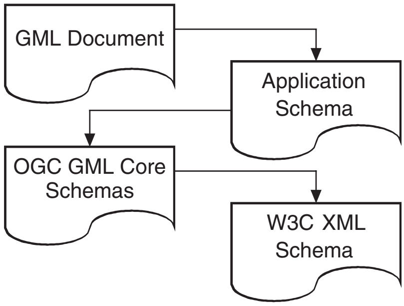

Fig. 1.2 Leveraging standards as information building blocks

model in a standards-based fashion. This reduces the level of effort required to define common nomenclatures and structures where others have already defined them. Figure 1.2 illustrates the building block effect of a GML document.

When creating the application schema, the GML core schemas possess many predefined blocks which can be reused. This saves time and effort by eliminating the requirement for redefining common blocks and structures, as well as (and more importantly) providing an output information model in a form that common tools can process and interpret.

As illustrated in Fig. 1.3, objects defined within the gml:AbstractFeatureType region indicate those inherited from the GML core model. The gml: namespace indicates reuse of an existing definition from the GML schemas. The objects are defined in the local application schema as specific to the National Earthquake Information System (NEIS) data model. A simple scripting process outputs a GML document as input for the WMS. A UMN MapServer installation is then configured to connect to the GML data. Source code, schema and configuration files for this example can be found at http://www.kralidis.ca/gis/eqmapping/.

Once this process is in place, the WMS can run stand alone and unsupervised as a Web service with a self-updating process to gather latest updates from the NEIS data site. As a result, any WMS-aware client application (Web-based or desktop) can interact with the NEIS data source for visualization, data extraction and/or analysis.

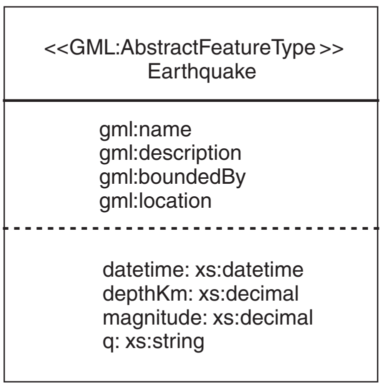

Fig. 1.3 Schema design view of earthquake data GML model

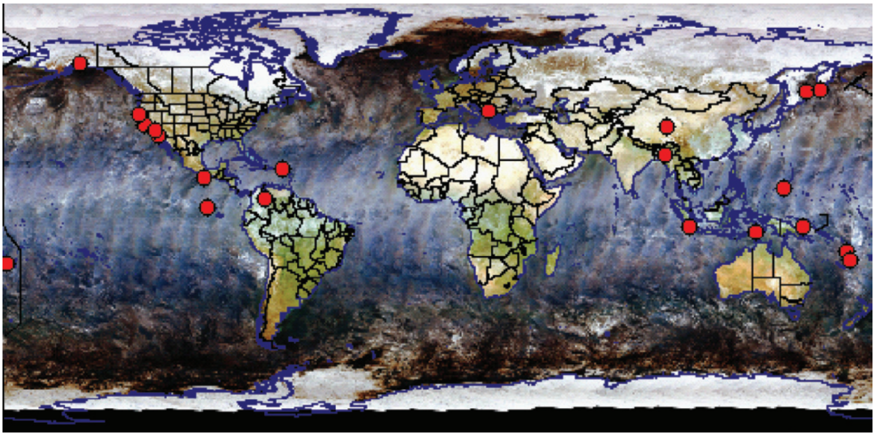

Fig. 1.4 Distributed data visualization: integrating envisat and NEIS from two different servers dynamically via WMS

A webpage could then automatically integrate the NEIS WMS and basemap data, which triggers two requests, producing the virtual Web map as shown in Fig. 1.4.

The WMS GetMap request embedded in the webpage to visualize the NEIS layer is written as follows:

```
http://geo.example.org/wms?
SERVICE=WMS
&VERSION=1.1.1
&REQUEST=GetMap
&SRS=EPSG%3A4326
&BBOX=-180.0000000309986,-112.5000000193741,180.0000000309986,
   112.5000000193741
&WIDTH=560
&HEIGHT=350
&LAYERS=neis
&STYLES=
&FORMAT=image%2Fpng
&BGCOLOR=0xFFFFFF
&TRANSPARENT=TRUE
&EXCEPTIONS=application%2Fvnd.ogc.se_inimage
```

To integrate the Envirsat layer as a backdrop to visualize NEIS, a similar WMS GetMap request would be invoked, changing only the LAYERS parameter and the location of the WMS serving the Envisat data.

This request is a valid WMS GetMap request connection, which means it can apply to any valid WMS server, given the correct URL location and content information. The advantage of using a standards-based API in this case is that it (or