Preface

When a thing is new, people say: 'It is not true.' Later, when its truth becomes obvious, they say: 'It is not important.' Finally, when its importance cannot be denied, they say: 'Anyway, it is not new.

~ William James, 1896

Several emerging phenomena and technologies, such as the increasing availability of mature open source software and the continuing evolution of distributed computing, are introducing a new dynamic into information system development. Specifically, these phenomena and technologies are enabling the development of a variety of innovative spatial information systems. This book contains several chapters that present innovative spatial information systems that have been developed for a specific problem or decision-making situation. Also included are several chapters that discuss key concepts and theories underlying current spatial information systems as well as technology trends and emerging concepts that may impact spatial information system development and applications in the future. Chapters are typically presented as case studies, are grounded in information science research, and have a specific practical application.

Spatial Informatics and Emerging Technologies

Spatial informatics can be described as the sciences concerned with the collection, manipulation, classification, analysis, storage, and retrieval of recorded spatial data, information, and knowledge. These sciences are utilized in the development, management, and use of spatial information systems. Of particular interest to researchers and practitioners alike, is the impact that emerging technologies may have on these systems. Open source software and distributed computing (a brief overview of each is presented below) are two of the many emerging phenomena and technologies that are impacting the development of spatial information systems.

Open Source Software

Free and open source software is software that gives users the right to run, copy, distribute, study, change, and improve it as they see fit, without the need to receive permission from or make additional payments to any external group or person (Bollinger, 2003). Perhaps two of the most well-known examples of open source software are the operating system Linux and the Apache Web server. Linux is the second most commonly-used operating system (Windows is number one) (Netcraft, 2001; The Economist, 2001) while Apache is the principal Web server in use today (72% vs. Windows 22%) (Security Space, 2006).

Mature open source software solutions have expanded the number of acquisition choices available to the organization beyond traditional closed source (commercial or proprietary) software solutions. As a result, there now exists the possibility of mixing open source and closed source solutions to best meet the information system needs of the organization. There are four strategies (Bollinger, 2003) for mixing open source and closed source software:

1. **Distribution mixing:** Open source and closed source software can be stored and transmitted together.
2. **Execution mixing:** Open source and closed source software can run at the same time on the same computer or network.
3. **Application mixing:** Open source can rely on closed source software to provide it with services and visa versa.
4. **Service mixing:** Open source can provide generic services to closed source software and visa versa.

Table 1. Comparison of open, mixed, and closed source software

	Open Source Software	Mixed Source Software	Closed Source Software
Operating System	Linux	Windows	Windows
Web Server	Apache	IIS	IIS
Database	MySQL	MySQL	MSSQL
Scripting Language	PHP	PHP	ASP

Table 1 illustrates the concept of open source/closed source software mixing in relation to the typical Internet-based information system.

As illustrated in Table 1, this specific collection of open source software represents the technologies that represent the open source Internet platform: LAMP = Linux + Apache + MySQL + (PHP | Perl | Python). The mixed source solution consists of an operating system and Web server supplied by Microsoft, Inc.; the database and scripting language are open source. The closed source solution is completely proprietary and is based on additional products from Microsoft – MS SQL database server and the Active Server Page scripting language. This is one example that illustrates the impact of open source software on information system development; in the chapters that follow, several more are presented.

Distributed Computing

A distributed computing system consists of multiple software components on multiple computers running as a single system (International Business Machines Corporation, 2004). Furthermore, the computers in a distributed system can be physically close together and connected by a local network, or they can be geographically distant and connected by a wide area network. A distributed system can comprise any number of possible configurations: mainframes, personal computers, workstations, minicomputers, and so forth. The goal of distributed computing is to construct a system such that it appears and acts as a single computer. Numerous emerging technologies are supporting the development of distributed computing and applications. These include:

- **Web services** have emerged as a popular standards-based framework for accessing network applications. Web services consist of a set of messaging protocols, programming standards, and network registration and discovery facilities that expose business functions to authorized parties over the Internet from any Web-connected device (Oracle Corporation, 2001). Basically, Web services allow specific application logic to be exposed and used between independent applications with a minimum knowledge of the Web service and/or underlying application. The advent of Web services promises to let organizations connect their applications to any number of other organizations relatively inexpensively and easily (Hagel, 2002).

- **Grid computing** has emerged as an important new field, distinguished from conventional distributed computing by its focus on large-scale resource sharing, innovative applications, and, in some cases, high-performance orientation (Foster, Kesselman et al., 2001). Grid computing seeks to address many of the problems associated with the continuing evolution of distributed computing, in particular, controlled and coordinated resource sharing and use for problem-solving in dynamic, scalable, multi-institutional virtual organizations (a set of individuals and/or institutions linked through resource sharing rules). In this instance, resource sharing is not primarily file exchange, but direct access to computers, software, data, and other resources, that is, those resources required by the range of collaborative, problem-solving, and resource brokering strategies that are emerging in industry, science, and engineering. Resource sharing in this situation is highly controlled and coordinated, with resource providers and consumers clearly defining the rules regarding what is shared, who is allowed to share it, and the conditions under which sharing occurs.
- **The portable Internet** is a platform for high-speed data access using Internet protocol (IP) and includes advanced short-range wireless technologies (within 30 meters) such as Bluetooth, medium-range wireless technologies (at least 150 meters) such as WiFi, and long-range wireless technologies (up to 50 kilometers) such as WiMAX as well as several advanced techniques that make more efficient use of the available spectrum, including spread spectrum, smart antennae, agile radios, and mesh networks. A wireless IP platform can be used to carry not only high-speed services, such as video entertainment and data transfer, but also medium-speed services, such as Web-browsing, and low-speed services such as voice or e-mail. As such, it is potentially substitutable over a wide-range of existing networks and services, and could impact a large number of current business models (International Telecommunications Union, 2004).

The emerging spatial information systems and applications that appear in the following chapters, encompass, in some manner, aspects of the open source software and distributed computing technologies described above.

Book Organization

Section I: Introduction

This section presents information regarding key concepts and theories underlying current spatial information systems. For instance, **Chapter I**, by Zhao, Yu, and Di, introduces all aspects of geospatial Web services from service-oriented architecture to service implementation. It covers the life cycle of geospatial Web services in terms of geospatial interoperable standards, including publish, discovery, invocation, and orchestration. Semantic issues regarding geospatial data and services are discussed, and the applications of standard-compliant geospatial Web services are reviewed.

Chapter II, by Hilton, Burkhard, and Abhichandani, presents an approach to an ontology-based information system design theory for spatial information system development. This approach addresses the dynamic nature of information system development at the beginning of the 21st century and addresses the question of how to establish relationships between the various design components of a spatial information system. An example of this approach is presented, along with examples of the various ontologies utilized in the design of this particular spatial information system.

Section II: Challenges for Spatial Information Systems

This section presents those innovative spatial information systems that have been developed for a specific problem. As seen in **Chapter III**, Judith Woodhall discusses how the need for geospatially-enabled data messaging among emergency response agencies can be enabled with the emergency provider access directory (EPAD). She describes the directory, how it enables message routing, and its fit into a boarder E-Safety network. She also discusses the architectural components of the EPAD, specifically the geographic information system module, and how Web services and open source products were used in the design to enhance the EPAD service offering.

Chapter IV, by Gunjan Kalra, discusses the process of providing information in its most accurate, complete form to its users, and the difficulties faced by the users of the current information systems. She describes the impact of prevalent technologies such as the multi-agent systems and the Semantic Web in the area of information supply via an example implementation and a model use case. She also offers a potentially more efficient and robust approach to information integration and supply process.

Ku and Zimmermann, in **Chapter V**, present an information architecture using Web services for exchanging and utilizing geotechnical information, which is of critical interest to a large number of municipal, state, and federal agencies as well as private enterprises involved with civil infrastructures. They propose an infrastructure of Web services, which handles geotechnical data via an XML format, report on its design, and share some initial experiences.

Chapter VI, by June K. Hilton and David E. Drew, discusses ScienceMaps, an online resource portal for standards-based science instruction using GIS technology. ScienceMaps is unique in that it concentrates on using GIS to teach, not on teaching GIS. Using an Internet-based GIS, ScienceMaps provides access to GIS technology and data to anyone, anywhere, with access to an Internet browser.

Section III: Decision-Making Environments

This section examines those spatial information systems that have been developed for a specific decision-making situation or environment. Gao and Sundaram, in **Chapter VII**, draw from several relevant disciplines to overcome the problems identified in various areas of spatial decision support and propose a generic spatial decision-making process and a domain-independent spatial decision support system (SDSS) framework and architecture to support this process. They develop a flexible SDSS to demonstrate an environment in which decision-makers can utilize various tools and explore different scenarios to derive a decision.

As seen in **Chapter VIII**, Sugumaran, Ilavajhala, and Sugumaran discuss the development of an intelligent Web-based spatial decision support system and demonstrate it with a case study for planning snow removal operations. They illustrate how traditional decision support system (DSS) and Web-based spatial DSS can be further improved by integrating expert knowledge and utilizing intelligent software components (such as expert systems and intelligent agents) to emulate the human intelligence and decision-making.

In **Chapter IX**, Todd G. Olson and Brian N. Hilton discuss Conservation Studio, a spatial information system that automates the entire process of conservation modeling, simulation, and planning. Conservation Studio consists of four software modules: Data Acquisition Interface, Habitat Suitability Analyst, Conservation Criteria Developer, and Implementation Modeler, the latter of which models the outcome of using tradable conservation credits to conserve habitat resources in a specified geographical plan area.

Chapter X, by Sugumaran and Bakker, discusses the need for the development of a decision support system to assist in the selection of an appropriate location for the development of future confined animal feeding operations (CAFO) structures. Furthermore, it presents the development of a decision support tool to aid CAFO managers and producers in selecting appropriate locations for animal confinements using geographic information system technology and CAFO regulations in Iowa.

Section IV: Future Trends and Technologies

This section highlights technology trends and emerging concepts and considers how they may impact spatial information system development and/or applications in the future. In **Chapter XI**, Lars Brodersen and Anders Nielsen, present the relationships and impacts between the various components of the spatial data infrastructure (SDI) and geo-communication. They also discuss a model for the organization of the passive components of the infrastructure, that is, legislation, collaboration, standards, models, specifications, Web services, and information.

Fengxian Fan, in **Chapter XII**, explores an implementation to process and interpret the data gathered by wireless sensor networks deployed to monitor rare plants and other endangered species. The system she presents in this chapter combines database management technology, geographic information system, and Web development technology to visualize the data gathered by these wireless sensor networks.

Chapter XIII, by Ibach, Malek, and Tamm, reviews the enabling technologies that drive system development and also discusses market factors, security and privacy concerns, and standardization processes that need to be taken into account concerning the "global real-time enterprise". The SEMALON (SEMAntic LOcation Network) approach is proposed as a basic infrastructure for discovery and composition of location-based services. A case study implementation for NOMADS Campus, a distributed spatial information system on the campus at Humboldt University Berlin, is presented.

Tolone, Xiang, Raja, Wilson, Tang, McWilliams, and McNally, in **Chapter IV**, propose a knowledge-driven methodology that facilitates the extraction of critical infrastructure (CI) information from public domain, that is, open source, municipal data sets. The proposed methodology was tested successfully on a municipality in the Southeastern United States and is considered to be a viable choice for CIP professionals in their efforts to gather CI information for scenario composition and vulnerability assessment.

In **Chapter XV**, Lyn Kathlene describes and analyzes the effectiveness of two methodological techniques, cognitive mapping and geographical information systems (GIS), for identifying social service resources. She also examines the processes used to integrate hand-drawn map information into geo-coded data points and provides recommendations for improving efficiency and precision.

In the final chapter, **Chapter XVI**, by Bruemmer, Few, and Nielsen, research to study and improve an operator's ability to navigate or tele-operate a robot that is distant from the operator through the use of a robot intelligence architecture and a virtual 3D interface is presented. Their results suggest that performance is improved when the robot assumes some of the navigational responsibilities or the interface presents spatial information as it relates to the pose of the robot in the remote environment.

References

Bollinger, T. (2003). *Use of Free and Open Source Software (FOSS) in the U.S. Department of Defense.* The MITRE Corporation.

Foster, I., Kesselman, C., et al. (2001). The anatomy of the grid: Enabling scalable virtual organizations. *International Journal of Supercomputer Applications, 15*(3).

Hagel, J. (2002). *The strategic value of Web services*. New York: McKinsey and Company.

International Business Machines Corporation. (2004). *TXSeries™ for MultiplatformsConcepts and Planning Version 5.1*. Armonk, NY: International Business Machines Corporation.

International Telecommunications Union. (2004). *ITU Internet reports: The portable Internet.* Geneva, Switzerland: International Telecommunications Union.

Netcraft (2001). Netcraft Web Server Survey — June 2001.

Oracle Corporation (2001). *Oracle 9i Application Server: Web Services Technical White Paper*. Redwood Shores, CA: Oracle Corporation.

Security Space. (2006, January 1). Web Server Survey.

The Economist (2001). *Survey of software: Out in the open*. The Economist.