GI_Forum 2016, Vol.2
Page: 46-56
Full Paper
Corresponding Author:
christoph.schaller@bfh.ch
DOI: 10.1553/giscience2016_02_s46

GeocentraleApps - Practical Approaches to Data Integration for Spatially Enabled Apps

Christoph Schaller[1] and Simon Hofer[2]

[1] Bern University of Applied Sciences (BFH), Switzerland
[2] Geocloud AG, Switzerland

Abstract

With modern mobile and internet technologies, spatial data is becoming ubiquitous. In order to realize the vision of a spatially enabled society, however, decision makers in government, public administration, business and society need to understand and actively take into account location as a driver in their decisions. This necessitates decision-support tools that integrate the required spatial and textual data from a variety of sources. GeocentraleApps is a platform for building such modern spatially enabled applications. These applications and the underlying spatial data infrastructures face challenges such as data discovery, matching between disparate data sources, issues with data and service quality, as well as the need for appropriate visualization and presentation. GeocentraleApps meets these challenges by flexibly combining a number of different mechanisms for data integration. This paper presents lessons learned from an analysis of the platform's data integration approaches with regard to their individual architectures. It points out the advantages and disadvantages of the current solutions and gives an outlook on future developments.

Keywords:

service-oriented architecture, data integration, spatial data infrastructures, distributed data

1 Indroduction

The demand for and use of spatial information are growing steadily. In the International Federation of Surveyors' (FIG) FIG Report 58, the Spatially Enabled Society (SES) was presented as an answer to that demand (Steudler et al., 2012). The report identified a sound data integration concept that defines common principles and standards as one of the required elements for enabling an integral view of spatial and textual data. According to the report, Spatial Data Infrastructures (SDI) can enable data sharing and reduce duplication. Additionally, SDIs serve as an enabling platform linking data producers, providers and value-adders to data users in order to reach the common goal of data sharing. The report sketches a common concept for data integration and points out the difficulties that have to be overcome when implementing such a concept (Kaufmann & Steudler, 2012). To achieve data

integration on a wide scale, a degree of interoperability has to be reached that includes not only the technical level, but also the underlying social and legal levels, and those of policy and institutional interoperability (Mohammadi et al., 2006). On the technical level, the current SDIs are only partially fit for the task and will need to adapt modern web-based technologies and address various issues, including security concerns, in order to serve as the basis for data integration in the SES (Lüthy & Kaul, 2015).

The GeocentraleApps platform is an integration platform designed to address these issues and to meet the requirements of the SES (Lüthy, 2014). Its design principles encompass: (a) the reuse of existing SDIs and information infrastructures to avoid data duplication; (b) the (re)use of proven open source components (e.g. OpenLayers or the GDAL library); (c) real-time access to the underlying data sources; (d) the use of modern web technologies wherever possible to enable distributed infrastructures and easy user access; and (e) flexibility and scalability to support the integration of a variety of data sources. Building upon a service-based architecture, the platform provides a number of components that serve as a basis for developing spatially enabled applications. These include components to support different data integration approaches, for example a rule-based engine enabling the matching of spatial and textual data (Lüthy, 2014); a data abstraction layer with pluggable data adapters supporting the declarative definition and combination of data from different sources; and an inter-application communication facility that allows for the integration of specialized apps (i.e. easy-to-use applications focused on a specific task such as displaying a web map) into more complex applications (Hofer et al., 2015). Since its inception, several applications have been developed based on the GeocentraleApps platform, thus validating its architecture and components on a practical level. Among these applications are spatially enabled management cockpits for infrastructure management in Swiss municipalities (Hofer et al., 2015), which were developed by an interdisciplinary team from university, industry and municipalities within an R&D project that was partially funded by the Federal Commission for Technology and Innovation CTI.

Following this project, an evaluation of the data integration approaches of the GeocentraleApps platformwas performed based on the lessons learned from their practical applications as well as additional research. Unlike most research contributions that focus on a single specific data integration approach, the aim was to gain a comprehensive overview of the most important data integration approaches of the platform and of how they complement each other. Therefore, the data integration mechanisms employed within GeocentraleApps were analysed with regard to their individual architectures as well as their advantages and disadvantages, in order to identify their suitability for certain applications and any needs for further development. This paper presents the most important lessons learned from this analysis. It first gives an overview of data integration approaches in general and their categorization found in the literature, before looking into findings from the analysis of the practical implementations within GeocentraleApps. The advantages and disadvantages of the different approaches are pointed out alongside examples of their usage. Finally, we present an outlook on future development.

2 Literature Review

Data integration is the process of combining data residing in different sources, and providing the user with a unified view of these data (Lenzerini, 2002). Spatial data integration, by extension, may be defined as the process of making different sets of data within a GIS compatible with each other so that they can be displayed and analysed together (Flowerdew, 1991). Data integration has been identified as one of the key elements of the SES (Steudler et al., 2012). In that context, Kaufmann & Steudler (2012) introduce a data integration concept with common principles and standards that aim to ensure interoperability. The interoperability components necessary to achieve data integration (social, legal, policy, institutional and technical) and connected issues are laid out in Mohammadi et al. (2006). Along with other aspects of spatial data integration, they are explored in more detail in Mohammadi (2008). Numerous other works on specific approaches and problems concerning spatial data integration can be found in the literature (e.g. Janowicz et al., 2010; Gitis et al., 2015; Bédard et al., 2001), a full review of which, however, is beyond the scope of this paper. Rather, this review concentrates on the literature that served as the conceptual basis for the analysis of the GeocentraleApps platform.

Different categorizations for data integration approaches can be found in the literature, for example the layered classification from the architectural viewpoint in Ziegler & Dittrich (2007). From the perspective of persistence, on the other hand, data integration approaches can be broadly divided into materialized data integration and virtual data integration. The former stores the integrated results, while the latter performs the integration on the fly and does not store the results (Kutsche et al., 1999).

Materialized Data Integration

Materialized data integration is characterized by storing data from different sources in a single integrated data model of a central repository (usually a relational database management system). Such repositories are often known as data warehouses. The integration is achieved through a process that (a) extracts the data from the source systems, (b) transforms it into the desired format, and (c) loads it into the common data model, a process known as the ETL (Extract, Transform and Load) process. The data warehouse approach was popularized by the methodologies proposed in Kimball & Ross (2002) and Inmon (2005). Other variants of the approach, including ones specifically pertaining to spatial data (e.g. Bédard et al., 2001), can be found.

Data warehouses play a major role in Business Intelligence and decision support systems. Their data models often use a multi-dimensional structure well suited for data analysis. Due to the fact that ETL processes are only executed periodically, the up-to-dateness of the data can be of concern. For applications with near real-time requirements, there are event-driven approaches to data integration (e.g. Naeem et al., 2008) that mitigate these problems. Periodically integrating the current state also allows the recording and analysing of changes over time. The duplication that accompanies materialized data integration, however, can also lead to a deviation from the source. This goes against some of the demands for modern SDI and is less desirable in a web service-oriented architecture (Lüthy & Kaul, 2015).

Virtual Data Integration

Virtual data integration is characterized by the fact that the data from the sources is not permanently duplicated and stored. Rather it is integrated on the fly by the integration system and only materialized (if at all) temporarily. While access times may be of concern, this approach completely avoids problems with the up-to-dateness of the integrated data. This makes virtual integration well suited for federated environments like SDI and for web service-oriented architectures. Two major approaches for virtual data integration can be identified in the form of federated schema and mediator/wrapper-based architectures.

The federated schema approach was developed out of federated databases (Sheth & Larson, 1990). Systems based on this approach provide functions akin to classical databases, including a uniform query language as well as read and write access to data (Kutsche et al., 1999). In this approach, the schemas from the data sources are exported and integrated into a single federated schema. Maintaining this schema can be a complex task and may limit the flexibility when integrating new data sources. The approach also introduces a certain degree of coupling, and therefore dependency between the systems involved. If the federation and the integration components are maintained by the user, one can speak of a loosely coupled system, while a federation managed by the administrator in control of the underlying data sources would constitute a tightly coupled system (Sheth & Larson, 1990). In both cases, the system will need components that map the external schemas of the data sources to the integrated federated schema, as well as components for translating user queries and distributing them to the data sources (Sheth & Larson, 1990).

More flexibility and a lower level of coupling can be achieved using mediator/wrapper-based approaches, which were first introduced in Wiederhold (1992). On the one hand, wrappers provide mediators with a uniform mechanism for data access. They encapsulate a data source and hide the technical implementation details as well as the complexity of the underlying source. On the other hand, mediators are components that offer users the possibility to query data. They take into account data from their own data sources as well as inputs from other mediators (Kutsche et al., 1999). This approach lends itself to service-based architectures where the functions of mediators are made available, for example through web services where each mediator handles the access to and integration of a limited set of data. The design also provides a high degree of flexibility for adding new data sources through additional mediators and/or wrappers.

3 Approaches to Virtual Data Integration

Mediator/Wrapper-based Architecture

The GeocentraleApps platform uses a mediator/wrapper-based approach for applications that require the reuse of existing data sources and complex matching logic during data integration. The components of the solution can be seen in Figure 1. Pluggable data adapters within an extensible data abstraction and transformation layer serve as wrappers that encapsulate the underlying data sources and translate them into an internal data model

(Lüthy, 2014; Hofer et al., 2015). The core of this approach is the so-called Rule Engine, which serves as a mediator that is capable of handling advanced integration scenarios. Using a set of configurable rules, it can perform either direct (e.g. using shared keys) or indirect (e.g. using spatial proximity) linking between data sources. The integration functions of the Rule Engine are supported by the Localizer and by the Reporting Engine. The Localizer facilitates the identification of the required data sources and resources using catalogues containing information about the SDI, as well as textual data that are accessible to the system; the Reporting Engine enables the output of the Rule Engine's results in different formats.

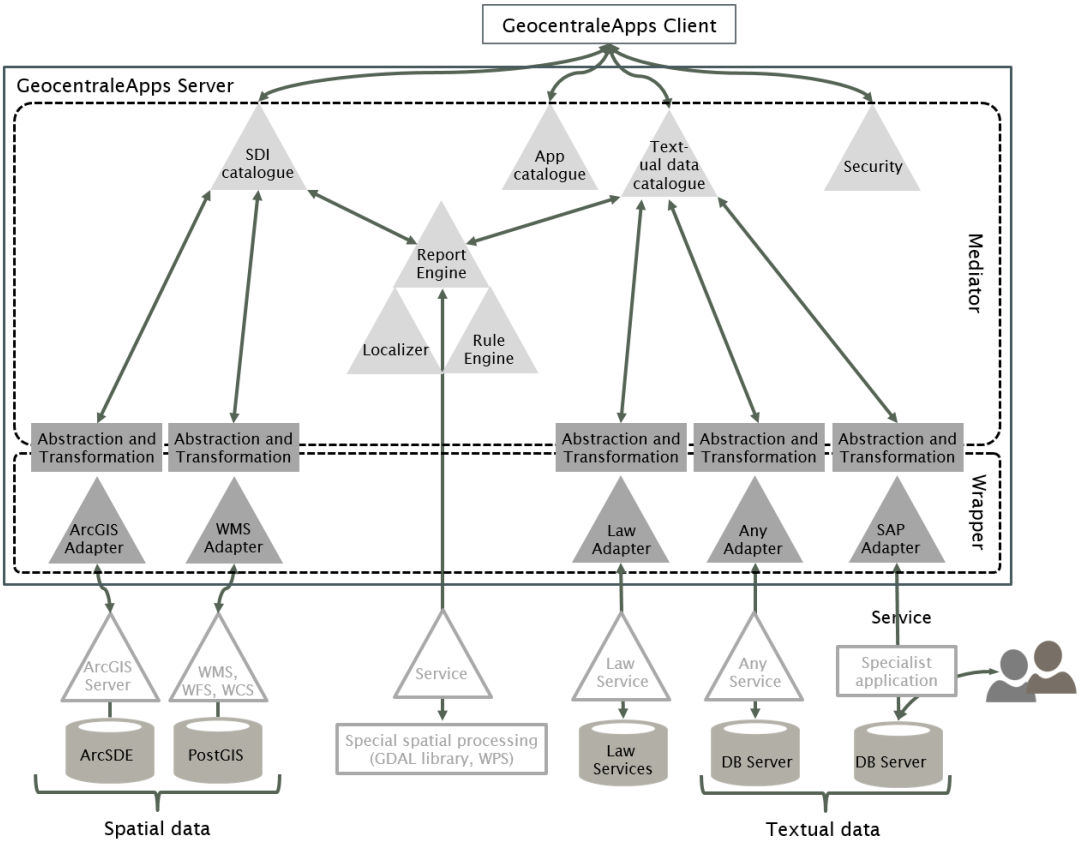

Figure 1: Architecture of GeocentraleApps

To illustrate the function of the Rule Engine, consider the example of integrating data about land plots with additional data based on a set of configured rules. The Rule Engine evaluates all rules against the inputs and, where applicable, integrates the respective data from the various sources. This process also involves localizing and evaluating additional data sets required as inputs for the rules. Table 1 shows two examples for matching rules, represented in a simplified syntax. The example for indirect matching is based on the application of an article of forest law (e.g. "Article 25") to objects within 50 metres of a forest boundary.

Taking the land plots as Input A, the Rule Engine would resolve the set of forest objects for input B and evaluate the rule for each of the input objects. The example for direct linking represents the equivalent of a LEFT OUTER JOIN in a relational database where building objects (input C) are linked to the corresponding land plots (input A) using a common plot identifier.

Table 1: Examples of matching rules, represented in simplified syntax

Description	**Rule**
Indirect linking using proximity	*IF A.Within(Buffer(B,50m)) OR A.Intersects(Buffer(b,50m)) THEN A.Attach("Article 25")*
Direct linking using common keys	*IF A.PlotId==C.PlotId THEN A.Attach(C)*

The mediator/wrapper-based approach allows for integrating data from a variety of sources without the need to duplicate the data. It guarantees the use of up-to-date data since the data adapters access the sources at query time. At the same time, if a data source is not available, this also means that a query may fail. While a mediator/wrapper-based architecture and its components are harder to build and maintain than other solutions, they allow for the implementation of very complex as well as application-specific data integration logic. If a specific data source is not available, a mediator may also divert part of a query to an alternative source (provided such a source exists), thereby partially offsetting the disadvantage stemming from the requirement that the data source needs to be accessible at query time. By using other mediators as data sources, the integration of other previously unknown data sources is also possible. While the approach guarantees up-to-date data, it has drawbacks when the analysis of historical data is necessary but the sources do not support historization. Such cases may require the use of materialized data integration, or a hybrid approach (i.e. using virtual data integration to combine current data with historical data from a repository that uses a materialized approach).

An example for the use of the mediator/wrapper-based approach is the use of the Rule Engine for the implementation of the Cadastre of Public-law Restrictions on landownership (PLR-cadastre) for the two (half) cantons Nidwalden and Obwalden in Switzerland (Lüthy, 2014). For the PLR-cadastre, the integration of spatial and textual data (especially legal information) is necessary. The solution aims to reuse existing data, infrastructures and processes where possible. Using the Rule Engine as a mediator, the solution allows real-time access to data sources, including two-way searches (from spatial to textual data, and from textual to spatial). The resulting three-tier architecture of the application can be seen in Figure 1, with a client that serves as the presentation layer at the top, an application layer in the middle, and a storage layer at the bottom. The distributed architecture employing web services to realize the separation between client and application layer, as well as between application layer and storage layer, fits well with the mediator/wrapper-based approach to data integration.

Federated Schema

The GeocentraleApps platform uses an approach based on a federated schema for applications that require the reuse of existing data sources whose data have clearly defined relationships. This approach uses the same data adapter components as those employed by the mediator/wrapper-based architecture to access different data sources. Since the relationships between the data are clearly defined, the matching logic needed during data integration is less complex. Therefore, the computational overhead and complexity of the Rule Engine as a mediator are avoided, and a set of specialized components based on the so-called Configuration Processor are used. The views of the federated schema correspond to queries on the data sources that are accessed through the data adapters. These views and their relationships (based on shared keys) are defined in an XML (eXtensible Markup Language) based configuration. The Configuration Processor and its components are able to interpret these configurations and query the data of the views at runtime. The results are then mapped into the abstraction layer's internal data model and passed to the application.

This simplified, federated schema approach shares most advantages and disadvantages of the mediator/wrapper-based architecture. Reuse avoids data duplication and guarantees that the data is up to date, since it is accessed at query time. The approach also relies on the availability of the underlying data sources. Because the definition of the schema and its views are fixed, however, the federated schema approach does not allow the use of an alternative data source in case of unavailability, or the dynamic addition of new data sources through delegation to other mediators. Because the configurations are maintained by the applications' administrator, this implementation of a federated schema results in a loosely coupled system that still retains a fair degree of flexibility when it comes to additions and changes in the data sources. While the Configuration Processor and its components perform many tasks similar to those of a mediator, it is easier to implement and maintain, since the clearly defined relationships between the data sources require less matching logic. Furthermore, the computational overhead at query time is lower, which results in slightly better performance in most scenarios. For these reasons, an approach based on federated schema is more widely used within the GeocentraleApps platform than the mediator/wrapper-based approach.

Examples for the use of federated schema are the GeocentraleApps charting service, and infrastructure management cockpits (Hofer et al., 2015). Based on the individual strategies of municipalities, the cockpits support management processes as well as planning, coordinating and making decisions based on key facts and figures. For infrastructure management, this also requires the integration of data from the municipalities' Geographical Information Systems (GIS). Both the charting service and the map client used in the cockpits define federated schemas with individual views on the relational databases and web services of the GIS. The charting service uses views with aggregated data needed to generate tables and graphics, while the map client uses views in the detailed records in order to generate maps. Besides avoiding data duplication, the use of the same underlying data source for both applications benefits application-based data integration mechanisms (especially due to the existence of common keys and data structures, thereby minimizing the need for conversion and matching).

Application-based Data Integration

In cases where an application needs to perform a certain function on data and that function is already offered by another application within the GeocentraleApps platform, the applications are integrated with one another in order to reuse that function. This corresponds to one of the design principles in GeocentraleApps: the development of small, specialized apps (e.g. a map client or a project planning component) that can be combined into more complex applications (Hofer et al., 2015). This approach can be classified as a form of data integration through application (Ziegler & Dittrich, 2007). This integration is realized through application-based communication (Lüthy & Kaul, 2015), in which messages between the applications are exchanged using a generic communication mechanism called AppCom. Building upon web services and web sockets, AppCom allows apps to announce their functions as well as to consume functions of other apps within the context of a user's session.

The approach has the advantage of reducing the duplication of code and helps to lower the overall complexity of the individual applications. It also avoids the duplication of data, since the operations of the integrated applications are carried out on the fly. The approach requires the applications concerned to support an interface for their integration with one another. Using application-specific programming interfaces for this purpose can introduce a strong dependency between the applications. However, this dependency can be alleviated somewhat by using a generic interface like AppCom or service-based approaches. These approaches do, however, have the downside that the application's functioning depends on the availability of the other integrated applications.

The infrastructure management cockpits, whose web application offers map-based visualizations, offer a good example of the use of application-based data integration within the GeocentraleApps platform. Instead of implementing its own mapping functionality, the cockpit application integrates the existing map client application. The cockpit embeds an instance of the map client and accesses its functions using the AppCom interface. If, for example, a filter action on one of the charts in the cockpit is executed, the filter parameters are passed to the map client using an AppCom call. The map client in turn will filter the display on the map according to the parameters supplied by the call.

4 Approaches to Materialized Data Integration

While approaches for virtual data integration are predominant within GeocentraleApps, data warehousing and similar materialized data integration approaches are used for some applications. This is the case mainly if the data sources are not directly accessible, if there is a need for historical data but historization is not supported by the source, or if the data needs to be transformed and/or enriched in a way that cannot be performed on the fly. These approaches use commercially available data integration solutions to implement ETL processes. After reading the data from its source (e.g. a database, a web service or a file), the ETL process performs the necessary transformation steps before the data is loaded into a data warehouse or a suitable relational database.

Data warehousing allows for the integration of data from sources unsuited for virtual data integration because they are not directly accessible. This includes sources that have no interfaces for direct access (such as web services or certain databases), and sources that are inaccessible due to separated operation environments. To integrate such sources, suitable technical and organizational processes to extract and transfer the data (e.g. in the form of a file) have to be established. One of the major advantages of materialized data integration is that it allows historization, by capturing changes over time when periodically executing the data integration process. Furthermore, the ETL process can include complex calculations and data enrichment steps that are not feasible for on-the-fly data access performed in virtual data integration approaches. One of the downsides of the approach, however, is the duplication of data. This duplication also requires a strategy for updating existing records in the data warehouse and handling changes when repeatedly integrating the data. The up-to-dateness of the data may also be of concern, since only data dating back to the last execution of the ETL process will be available. When dealing with the integration of large amounts of data and possibly its historization, materialized data integration also requires considerable amounts of storage space. Furthermore, the complexity of an integrated data model for a data warehouse must not be underestimated. In the case of an integrated view across all data sources not being required, the complexity of a data warehouse may be avoided, and solutions using simpler data models in separate repositories can be employed.

The cockpits for assisting infrastructure management (Hofer et al., 2015) expand on previous work on management cockpits for small and medium-sized municipalities (Schaller et al., 2010). These cockpits also employ a classic data warehouse with a multi-dimensional schema for the integration of several data sources that are not directly accessible. Currently, the integration of data from the municipal citizen register (Einwohnerregister), financial data and tax data are the main use cases of the data warehouse. A materialized data integration approach in the form of a data warehouse was favoured in this case due to the lack of directly accessible web services and of support of historization on the side of the source applications, as well as for organizational reasons (in particular separate operating environments and security concerns). Furthermore, the ability to analyse changes over time is crucial for the cockpits. For future developments, however, the data warehouse will be extended only if there are compelling reasons to do so. Thus new topics will be added only sparingly to the data model of the warehouse.

5 Conclusion and Outlook

Conclusion

Depending on the use case and the data sources involved, the requirements for data integration in spatially enabled applications vary, and different approaches to data integration have to be taken. These may include the combination of multiple approaches. The requirements for modern SDIs favour the use of virtual data integration approaches that avoid data duplication and guarantee the availability of up-to date data by accessing the source at runtime. Depending on the requirements, approaches that use materialized data integration are, however, still viable or even necessary.

Experiences with the applications based on the GeocentraleApps platform have shown that there is no "one size fits all" approach to data integration. Rather, the approach has to be carefully selected and adapted to the individual use case. A common underlying platform such as GeocentraleApps that can be adapted to different approaches can simplify this task and reduce costs through reuse. The platform's flexible and complementing data integration approaches also facilitate its adaptation for new use cases and applications. The data abstraction layer in particular has proven to be a valuable asset that can be utilized in different scenarios, including virtual and materialized data integration.

Future Work

The GeocentraleApps platform and its underlying framework are steadily being extended and refined. Driving factors are the ongoing development of existing applications which use the platform, as well as the addition of new ones. Ongoing work includes the central management of configurations that will enable reuse across applications, thereby eliminating redundancies and inconsistencies. This will also allow for new scenarios supporting a kind of central federated schema within the GeocentraleApps platform based on centrally defined data abstraction layer configurations.

Efforts to integrate the GeocentraleApps framework into ETL processes aim to leverage the data adapters of the data abstraction layer to allow the integration of data sources not directly supported by commercial data integration solutions. Furthermore, solutions with hybrid data integration approaches are being explored in order to complement the virtual data integration of systems that lack the support for historization. The goal is to historize selected data and key indicators using materialized integration approaches, such that this data can be combined with current, detailed data using virtual data integration mechanisms.

References

Bédard, Y., Merrett, T., & Han, J. (2001). Fundamentals of spatial data warehousing for geographic knowledge discovery. Geographic data mining and knowledge discovery, 2, 53–73

Flowerdew, R. (1991). Spatial data integration. Geographical Information Systems, 1, 375–387

Gitis, V. G., Weinstock, A. P., & Derendyaev, A. B. (2015). Basic concepts of integration of the dynamic GIS technology into a monitoring system of spatial processes. Journal of Communications Technology and Electronics, 60(12), 1445–1458

Hofer, S., Schaller, C., Haring, P., & Lüthy, J. H. (2015). Infrastructure management in Swiss municipalities – Development of a modern, spatially enabled management cockpit. FIG Working Week 2015, Sofia, Bulgaria

Inmon, W. (2005). Building the Data Warehouse. New York: John Wiley & Sons

Janowicz, K., Schade, S., Bröring, A., Kessler, C., Maué, P., & Stasch, C. (2010). Semantic enablement for spatial data infrastructures. Transactions in GIS, 14(2), 111–129

Kaufmann, J. & Steudler, D. (2012), Common data integration concept. In D. Steudler & A. Rajabifard (Eds.), Spatially Enabled Society, FIG Report, 58, Copenhagen, Denmark: International Federation of Surveyors (FIG), 23–28

Kimball, R. & Ross, M. (2002). The Data Warehouse Toolkit: The Complete Guide to Dimensional Modeling. NewYork: John Wiley & Sons

Kutsche, R. D., Leser, U., & Weber, H. (1999). Federated information systems: Concepts, terminology and architectures. Forschungsberichte des Fachbereichs Informatik, 99(9), Technische Universität Berlin

Lenzerini, M. (2002). Data integration: A theoretical perspective. In Proceedings of the twenty-first ACM SIGMOD-SIGACT-SIGART symposium on Principles of database systems, 233–246

Lüthy, J. (2014). An integration platform for a spatially enabled society. In D. Steudler (Ed.), CADASTRE 2014 and Beyond, FIG Report, 61, Copenhagen, Denmark: International Federation of Surveyors (FIG), 43–48

Lüthy, J. & Kaul, C. (2015). Demands for a Spatial Information Infrastructure Fit for Cadastre 2034. FIG Working Week 2015, Sofia, Bulgaria

Mohammadi, H. (2008). The Integration of Multi-Source Spatial Datasets in the Context of SDI Initiatives (PhD Dissertation, University of Melbourne, Australia)

Mohammadi, H., Binns, A., Rajabifard, A., & Williamson, I. (2006). The development of a framework and associated tools for the integration of multi-sourced spatial datasets. Proceedings of 17th United Nations Regional Cartographic Conference for Asia and the Pacific, Bangkok, Thailand

Naeem, M. A., Dobbie, G., & Webber, G. (2008). An event-based near real-time data integration architecture. In 2008 12th Enterprise Distributed Object Computing Conference Workshops, IEEE, 401–404

Schaller, C., Neuroni, A. et al. (2010). Advanced cockpits for municipalities – focusing on the relevance and challenges of importing data. In: J.-L. Chappelet, O. Glassey, M. Janssen, A. Macintosh, J. H. Scholl, & E. Tambouris (Eds.), Electronic Government. Proceedings EGOV und EPART 2010, 69–78

Sheth, A. P. & Larson, J. A. (1990). Federated database systems for managing distributed, heterogeneous, and autonomous databases. ACM Computing Surveys (CSUR), 22(3), 183–236

Steudler, D. & Rajabifard, A. (Eds.) (2012). Spatially Enabled Society, FIG Report 58, Copenhagen, Denmark: International Federation of Surveyors (FIG)

Wiederhold, G. (1992). Mediators in the architecture of future information systems. Computer, 25(3), 38–49

Ziegler, P. & Dittrich, K. (2007). Data integration – problems, approaches, and perspectives. In J. Krogstie, A. Opdahl, L. & Brinkkemper, S. (Eds.), Conceptual Modelling in Information Systems Engineering, 39–58, Berlin / Heidelberg, Germany: Springer