SYSTEM ARCHITECTURE FOR GEOSPATIAL VIRTUAL DATA INTEGRATION IN WEB-BASED APPLICATIONS

J. P. Duque[1]*, M. A. Brovelli[1]

[1] Department of Civil and Environmental Engineering (DICA), Politecnico di Milano – juanpablo.duque@mail.polimi.it; maria.brovelli@polimi.it

KEY WORDS: system architecture, data integration, geospatial data, geospatial standards, FAIR principles

ABSTRACT:

The wide availability of geospatial data from different sources makes it necessary to create systems that are able to use and integrate the data to generate added value. We propose a system architecture following FAIR principles (Findable, Accessible, Interoperable, Reusable) and state-of-the-art methodologies for a server-side web-based application that performs virtual data integration over data sources that implement geospatial information standards. The architecture extends the mediator-wrapper design pattern with additional components that provide the system with additional flexibility and modularity, much needed for modern web applications. The architecture is composed of the mask, which acts as the interface of the system towards external users; a mediator that handles processing and data integration logic; a set of wrappers that communicate with the external data sources; persistent storage to provide flexible configuration and metadata capabilities to the system; and messaging queue for enabling asynchronous processing. At the same time, the architecture's components are divided into four layers, each one with a specific role: presentation, configuration, processing, and communication.

1. INTRODUCTION

We live in the era of information. Hundreds of thousands of gigabytes of data are being produced by the minute from apps, websites, satellites, sensor networks, and many other sources. When raw data is provided with context and meaning, it becomes information. Furthermore, when data contains a geographic or spatial reference, i.e., a location, it is called geospatial data. With the advances in Geographic Information Systems (GIS) in the last couple of decades, geospatial information has gained importance for organizations and governments as a driver of change, being used for various applications such as disaster and risk mitigation, urban planning, environmental monitoring, land use/land cover change detection, hydrology, and many more. Geospatial data is usually interdisciplinary and spans many fields of study as it potentially includes every object, process, or entity that has a spatial location. In order to generate geospatial information it is necessary to have platforms and systems that are able to combine and operate on raw data from multiple domains and provide it with meaning. Nowadays, with the ever-growing volume of geospatial data produced by Earth Observation satellites, IOT and sensor networks, and Smart Cities, the pressing issue of GIS is: How to use the massive volume of data available and generate added value from it?

In this paper, we are documenting a system architecture designed for server-side, web-based geospatial applications that want to leverage the wide availability of geospatial data and standardised geospatial information web services. This architecture is based on the mediator-wrapper architecture, also known as mediator-based, and considers the virtual data integration of widely used standardised geospatial information web services (e.g., WMS or WFS), as well as any other modern geospatial API services (e.g., OGC APIs, ArcGIS REST API), as they are designed to better fit in modern web applications and systems. API-based standards, as the newer OGC standards, are still in the process of becoming mainstream and are only being mentioned or considered in a few recent academic articles. Therefore, it is important to start building and proposing systems around them. Additionally, this architecture is designed considering the FAIR principles of Findability, Accessibility, Interoperability, and Reusability (Wilkinson et al., 2016).

This proposed system architecture does not consider a specific technology or reference implementation but is solely built on top of (open) standards. Although, we will provide examples of technologies that could fit into each of the architecture's components.

This paper is divided into three more sections. In section 2, we address the theoretical background of interoperability and data integration in the geospatial sciences by exploring the state of the art and the different architectures and models proposed in academic literature, as well as present the geospatial data standards that are currently available. Section 3 describes the proposed architecture, each of its components, and the possible technologies that fit in each one of them. Finally, section 4 contains the conclusion and future work.

2. THEORETICAL BACKGROUND

2.1 State of the art

The beginning of GIS dates back to the decade of 1960, with the development of the Canada Geographic Information System, known as the first GIS. The appearance of the internet marked a breaking point for GIS technology, as it started moving from a private, desktop-only context to an open, online one. Web GIS started to become relevant as organisations realised the importance of geospatial information, and needed more dynamic tools and online sharing. Geospatial information standards became necessary to tackle the issue of data interoperability, as a multitude of new formats and services appeared due

---

* Corresponding author

https://doi.org/10.5194/isprs-archives-XLVIII-1-W2-2023-939-2023 | © Author(s) 2023. CC BY 4.0 License.

to the widespread of GIS. As a consequence, the Open Geospatial Consortium (OGC), ISO (ISO/TC 211 Geographic information/Geomatics) and other organisations began addressing the interoperability of geospatial data and proposed a set of (open) geospatial information standards. One of the overarching principles of OGC standards is the FAIR principle, which states that data must be Findable, Accessible, Interoperable, and Reusable.

With the wide availability of geospatial data from Earth Observation satellites, sensor networks, and local Spatial Data Infrastructures (SDI), the current challenge is how to use the data available to generate added value. The practice of combining or using data from multiple sources is known as data integration. Although related, it is important to make the distinction between interoperability and data integration. Interoperability is the ability of systems to operate effectively and efficiently in conjunction with other systems, while data integration refers to the seamless combination of data from different data sources (Noardo, 2022). Data integration is a computer science field that focuses on the integration and combination of data from multiple, heterogeneous sources. Virtual data integration deals directly with data hosted in external systems. This means that there is no data replication and the responsibility of maintaining and handling the data is passed to the provider (AnHai et al., 2012).

Standardisation comes very close to data integration. In the area of geospatial information, data integration has been applied in several contexts and with different methodologies and frameworks. An example is the Data Lake used by the European Union for the development of the Digital Twin Earth and leverage the high amount of Earth Observation and sensor data that they own (Juarez et al., 2023). The heterogeneous nature of geospatial data requires also strong metadata, ontologies, and semantic models. Although automatic geospatial data integration is complex, efforts are also documented in scientific literature (Bogdanović et al., 2015, Prudhomme et al., 2019).

In recent years, new technologies that strongly benefit from the integration of multiple geospatial data sources are blooming, such as Digital Twins (DT) and the Digital Earth. Smart cities and countries such as Singapore are assembling their own digital twins (Shahat et al., 2021), while efforts like Europe's Digital Twin Earth have already begun. The European Data Spaces is an initiative to homogenize and provide interoperable, huge volumes of European and global data. This is useful, among other things, for the development of the Digital Twin Earth, envisioned in the Destination Earth programme (Scerri et al., 2022, Nativi et al., 2021).

Such efforts require systems that are able to handle and integrate data properly. In many contexts and areas, system architectures have been proposed in the academic literature. A system architecture in computer science is a high-level design of the components that make up a system and its interactions. In particular, for geospatial data integration, many architectures have been proposed and documented. Service-Oriented Architectures (SOA) that leverage standardised geospatial services (Daly and Ranwashe, 2023, Xie and Li, 2018). Federated Systems for multi-source data management and integration (Cinquini et al., 2014). Standards-based geospatial services based on OGC standards (Rienzi et al., 2023). Layered architectures designed for internet-based applications (Osorio et al., 2017). Mediator-based architectures for data integration of multiple heterogeneous databases and geospatial services (Huang and Liang, 2014, Dončević et al., 2023). And Open-Source-based WebGIS architectures for geospatial WebGIS applications that use Free and Open-Source software (FOSS) (Agrawal and Gupta, 2014, Agrawal and Gupta, 2017, Bandyophadyay et al., 2012).

These architectures are usually oriented towards specific contexts that benefit greatly from geospatial data. For example, the idea of the Smart City is heavily supported by data integration and geospatial information (Rienzi et al., 2023, Santos et al., 2018) by integrating and merging data coming from different city systems, sensor networks, social behaviour, infrastructure, and 3D models to create a more precise representation of a city. Similar to the Smart City, the concept of Digital Twins benefits from geospatial data integration. Urban DT, City DT, and the Digital Twin Earth are concepts that are gaining traction in the academic and industrial circles (Saeed et al., 2022, Shahat et al., 2021), and efforts are being made to organize and define data integration strategies for the usage of the high volumes of data required to create such systems (Duque and Brovelli, 2022, Cinquini et al., 2014).

With the development of smart cities and Digital Twins, disaster and risk management is leveraging the use of geospatial data integration for the well-being of people and the environment. This discipline benefits from geospatial data integration as it allows more accurate and timely responses in case of disasters, and better, more precise, early warning systems and models (Osorio et al., 2017, Xie and Li, 2018).

2.2 Overview of geospatial data standards

To achieve interoperability and utilise to the full extent the data that is currently available, a strong set of standards is necessary. In the context of geospatial information, ISO/TC 211 is in charge of revising and publishing the ISO international standards for Geographic Information and Geomatics. In close collaboration, the OGC maintains, publishes, and reviews official standards as well as community standards. Other organisations have their own set of formats and services which, although not standard, are widely used and known as de-facto standards (i.e., ESRI's Shapefile, ArcGIS REST API, Google Maps API).

OGC provides standards for formats, web services, tile matrix sets, database schemas, and architectures. We focused only on the set of OGC API's and the OGC Web Services (OWS), which are the most important open standards for web technologies. Other API-based geospatial services exist, such as ESRI's ArcGIS REST API, Google Maps API, or MOTU (used by Copernicus Marine Services), among many others. We will focus mainly on the OGC APIs and the OWS for the architecture.

As mentioned, it is possible to divide the web-based OGC standards into two categories: the legacy OGC Web Services or OWS (e.g., WMS, WFS) and the more recent family of OGC APIs. OGC Web Services are widely used among organizations and are based on the eXtended Markup Language (XML), which was the most popular and widely used language for web services in the early stages of the internet. They constitute the most used OGC standards, as most of the mapping libraries implement them and are available in most of the map servers. On the other hand, OGC APIs are more recent and constitute a modern approach to geospatial web services, as they are designed for JSON and HTML responses, and with a well-defined structure based on OpenAPI. JSON is more optimized for modern internet browsers.

Table 1 reports the OWS and a short description of each of them.

Service	Description
Web Map Service (WMS)	Maps as images, used for visualization of geospatial data.
Web Feature Service (WFS)	Sharing of geospatial feature data (vector).
Web Coverage Service (WCS)	Sharing of coverages or gridded data, such as rasters or NetCDF files.
Web Processing Service (WPS)	Provides an interface for processing geospatial data. Defines the accepted inputs and the outputs of the process.
Web Map Tile Service Service (WMTS)	Similar to WMS, but uses tiles instead of full images.
Catalog Service for the Web (CSW)	Provides search capacities of descriptive information and metadata for geospatial data sets and services.

Table 1. List of available OGC Web Services (OWS) specifications

There are currently 15 OGC APIs for the web according to the OGC website (OGC API, n.d.). Table 2 reports the currently approved specifications and the ones that are under development (corresponding to 10 of the 15 specifications), a short description of each of them, and their homologous OWS specification. The first five reported standards are approved, while the rest are still a work in progress. Some of the OGC APIs match with one of the OWS, while others provide new features or extend the functionalities of the other specifications.

3. PROPOSED SYSTEM ARCHITECTURE

Considering FAIR principles, data integration methodologies, previous work, and the particularities of geospatial data, we propose a system architecture for server-side web applications whose purpose is to virtually integrate standardized geospatial data services.

This architecture is based on the mediator-wrapper design pattern. In particular, it is an extension of the Mask-Mediator-Wrapper architecture previously proposed by Dončević (Dončević et al., 2023). The core data integration strategy is realized by the mediator-wrapper pattern, while the presentation is handled by the mask. Additional components are added to provide additional flexibility to the architecture in a web-based environment, such as configuration and asynchronous processing. It also uses concepts from Service-Oriented Architectures (SOA) as the data integration is performed over web services and APIs. The architecture is composed of five components: mask, mediator, wrappers, persistent storage, and messaging queue (optional). Likewise, the components of the architecture are separated into 4 layers, each one with a different role within the system: the presentation layer, the configuration layer, the processing layer, and the communication layer.

OGC API	Description	OWS Homologous
Common	Specifies the building blocks that are shared by most or all OGC API standards to ensure consistency.	N/A
Features	Offers the capability to create, modify, and query feature geospatial data (vector) on the web.	WFS
Environmental Data Retrieval (EDR)	Lightweight interfaces to access Environmental Data resources.	N/A
Tiles	Provide extended functionality to other OGC API standards to deliver vector tiles, map tiles, and other tiled data.	Extended WMTS
Processes	Supports the wrapping of computational tasks into executable processes to be offered by a server through a Web API.	WPS
Coverages	Allow discovery, visualization and query of coverages, such as rasters, grids, and data cubes.	WCS
Records	API-based catalogue system for creating, modifying, and querying metadata on the Web.	CSW
Styles	API to manage and fetch map styles.	N/A
Maps	Serve spatially referenced and dynamically rendered electronic maps.	WMS
Routes	Allows applications to request routes in a manner independent of the underlying routing data set, routing engine or algorithm.	N/A

Table 2. List of approved and under development OGC API specifications

In general terms, the architecture uses the mediator-wrapper pattern by centrally enforcing access to external data from standardised web services through a mediator, while the communication (i.e., connections or requests) with external data sources is made through a set of modular wrappers. The modular nature of the architecture is well-suited for web applications, as it encourages the separation of concerns by separating the logic of connecting and translating the responses from external services, the global integration logic and data processing, and the presentation logic. It is important to mention that the wrappers connect to standardised web services and APIs as the interfaces to the data rather than through direct connections to the raw data. In addition to the mediator and the wrappers, the architecture's additional three components are in place to adapt it for modern web usage by providing additional flexibility and functionalities, considering the particularities of geospatial data. Finally, it is important to note that this architecture is designed for server-side web applications and does not consider client applications with features such as visualization or data presentation.

Another particularity of this architecture is that it is intended for virtual data integration applications. This means that the data is not hosted by the system, but by external systems that provide it through services. This encourages data reusability instead of replication, as producing and hosting data is usually expensive (Johnson et al., 2017), and provides a more cost-effective solution for organizations.

Figure 1 shows the complete architecture diagram, depicting the components and layers of the system and the relations between them. The description of the layers and components of the architecture is presented in the following sections.

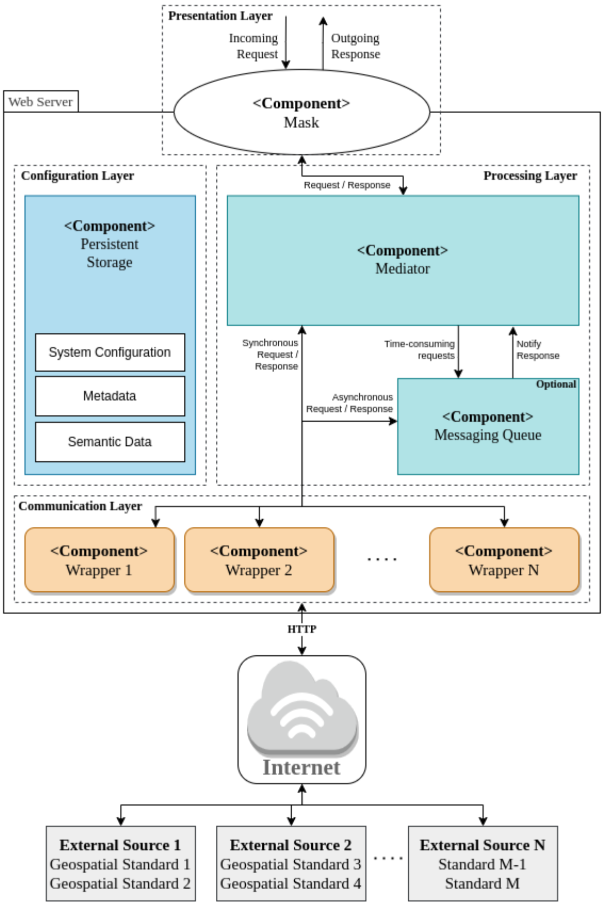

Figure 1. System architecture for geospatial virtual data integration.

3.1 The Architecture Layers

The components in the architecture are divided into 4 layers: the presentation layer, the configuration layer, the processing layer, and the communication layer. Each layer has a different role in the system and ensures the separation of concerns (SoC) between the components, a concept that is fundamental for maintaining modularity in a system.

Figure 2 shows the simplified architecture depicting the layers, components, and flow of a request within the system.

- **Presentation Layer**: The presentation layer represents the system towards the clients, which are external to the system, and the communication with them by exposing a predefined interface. This layer contains the mask component.

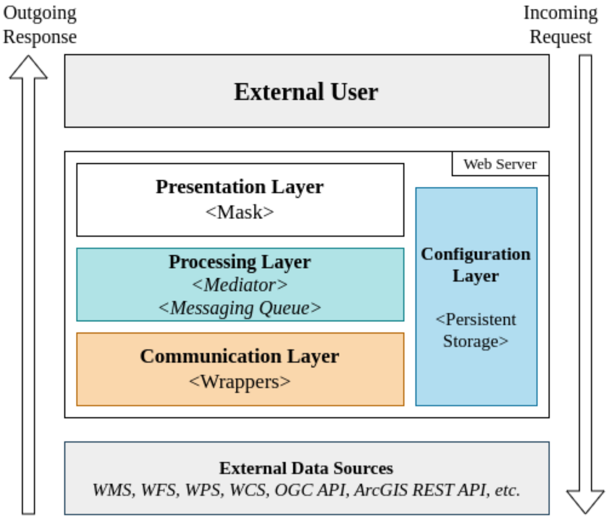

Figure 2. Simplified system architecture showing the layers and the request flow.

- **Configuration Layer**: Modern systems are highly configurable and flexible. In order to adapt to newer internet paradigms, this layer offers configuration capabilities through the persistent storage component. Additionally, provides semantic capabilities by allowing the storage of ontologies and metadata. This layer offers information transversally to the whole system as parameters.
- **Processing Layer**: This layer contains the business and integration logic of the whole system. It includes the mediator component and the optional messaging queue for providing asynchronous processing capabilities, in case the system needs it, due to processing complexity and heavy size of geospatial data.
- **Communication Layer**: The role of the communication layer is to act as the interface between external data sources over the internet and the processing layer (i.e., the mediator). This layer contains the various wrappers of the system that handle the requests and responses towards data sources.

3.2 Mask

This component constitutes a centralized interface and entry point to the system that provides predefined functions to externalize only certain internal functionalities of the system. This component is adapted from the architecture described by Dončević (Dončević et al., 2023) as an extension of the mediator-wrapper architecture. The rationale behind this component is to provide a unified interface to access the different underlying data sources that the system integrates and the functionalities that the system chooses to externalize (e.g., data processing or analysis, visualization, catalogue). This component can be implemented, for example, by means of an API, and, if possible, comply with geospatial standards.

From a functional level, it is important to have a mask on the system, as it represents the system towards external users and separates the presentation layer from the processing layer. This ensures the separation of concerns in the presentation and processing.

3.3 Mediator

This component acts as the middleware between the user requests made through the mask and the data source connection made through the wrappers. Additionally, the mediator contains the integration and processing logic of the whole system. As a middleware, it handles incoming requests and routes them to the specific wrapper that is able to process the request. As the processing unit of the system, it is in charge of receiving the responses from the wrappers and building responses that align with the predefined structure of the mask or performing any kind of processing with the incoming data.

Regarding the technologies that could take the place of the mediator, it is possible to create a custom implementation using any modern web server technology, such as Node.js, Django, or Flask. It is also possible to use a map server that supports WPS or OGC API Processes such as the open alternatives ZOO-Project, pygeoapi, or GeoServer. Nevertheless, using a map server usually requires that the data is within the system, breaking the virtual data integration.

3.4 Wrappers

The wrappers are modular components that handle communication with external data sources. By data sources we mean standardised geospatial web services or APIs. The wrappers contain the logic to connect to one or more data sources and provide modularity and flexibility to the system. By using the wrappers, the task of integrating new data sources relies on appending new wrappers to the system, rather than changing the complete integration logic. Wrappers work very closely with the mediator and must be implemented in the same web server technology.

Wrappers work by translating requests coming from the mediator into requests that a particular data source is able to process and respond to. When a response is received from the data source, the wrapper transforms the response to a specific global schema in which the mediator expects the response. To have a fully flexible system, it is necessary to define a robust global schema that covers any possible wrapper and data source possible. This is partially addressed by the fact that data sources are provided through standardised interfaces, meaning that the structure and responses are predictable.

3.5 Persistent Storage

This component is a multipurpose storage component that provides additional flexibility to the system by allowing the storage of dynamic configuration parameters, semantic links and information, indexes, schemas, metadata, translations, and any other configuration information that the system could use. This component is transversal to the entire system in order to modify and tailor the application during runtime. This component is not intended to be a geospatial data store, but to allow flexible configurations and persistent storage. In addition, an administration application can be coupled with the database to allow non-technical users to change the server configuration.

The implementation of this component can be realised using any database management system, both relational and non-relational. Open-source examples of databases include PostgreSQL, MySQL, MongoDB, or even a minimalist option such as SQLite.

3.6 Messaging Queue

As geospatial datasets are usually heavy, the processing of geospatial data may be time-consuming and lock down the system. Additionally, as the architecture depends on external sources and services, unexpected delays in them can affect the system response times. As availability is key for web servers, the Messaging Queue component (also known as Message Broker) allows to asynchronously handle processes that are time-consuming within the system without locking it, such as data preprocessing or analysis, downloads, or iterations over large datasets.

In general, computations can be synchronous or asynchronous. A synchronous computation is one where each step of the process is done in a sequential order, thus, for one step to be done it has to wait for the previous step to be completed. On the other hand, an asynchronous computation does not need to wait for the previous step to be completed to start its processing. Instead, the process is done in parallel. Asynchronous computations are very important for server-side applications, as they allow time-consuming computations to be completed without blocking other incoming requests, and also because requests usually have a timeout, so long computations will never be completed due to the timeout.

Many geospatial data analysis platforms compliant with WPS and OGC API Processes contain such components to provide asynchronous capabilities. This component can be implemented by using the asynchronous capabilities of any modern web server framework plus the capabilities of a message broker. Examples include RabbitMQ, Eclipse Mosquitto, and HiveMQ.

In practice, asynchronous processing works by queuing jobs and processes once they are requested, sending an ID as the response to the request. When the system finishes the processing, the message queue notifies the system and the response is temporarily stored. Then the user can consume the response using the ID they previously received.

As this is an optional component. For systems that do not require heavy processing capabilities, this component is not strictly necessary.

4. CONCLUSION AND FUTURE WORK

Through this paper, we presented a system architecture for server-side, web-based applications that perform virtual geospatial data integration over multiple, heterogeneous data sources by leveraging the use of open geospatial standardised web services. This proposed architecture follows state-of-the-art methodologies and extends existing data integration architectures to achieve the flexibility, modularity, and scalability that are required for modern web applications. It is relevant in the context of open geospatial information as it considers and is built on top of open geospatial standards.

The architecture is divided into 4 layers (presentation, configuration, processing, and communication), each one with a defined role, and comprises 5 components: mask, mediator, wrapper, persistent storage, and messaging queue. It is an extension of the mask-mediator-wrapper architecture, with two additional components to ensure enough flexibility to be used in modern web applications.

In future work, we will implement a prototype application that implements this architecture, and test it with respect to other map servers that are able to integrate geospatial data in a real-life setting. Additionally, we will extend the architecture to include a client-side application that could use the capabilities of a system following the proposed architecture.

REFERENCES

Agrawal, S., Gupta, R. D., 2014. Development and comparison of Open Source based web GIS frameworks on WAMP and Apache Tomcat web servers. *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences - ISPRS Archives*, 40, 1-5.

Agrawal, S., Gupta, R. D., 2017. Web GIS and its architecture: a review. *Arabian Journal of Geosciences*, 10, 1-13. https://link.springer.com/article/10.1007/s12517-017-3296-2.

AnHai, D., Alon, H., Zachary, I., 2012. *Principles of Data Integration*. Elsevier Inc.

Bandyophadyay, M., Singh, M. P., Singh, V., 2012. Integrated visualization of distributed spatial databases: An open source Web-GIS approach. *2012 1st International Conference on Recent Advances in Information Technology, RAIT-2012*, 619-621.

Bogdanović, M., Stanimirović, A., Stoimenov, L., 2015. Methodology for geospatial data source discovery in ontology-driven geo-information integration architectures. *Journal of Web Semantics*, 32, 1-15.

Cinquini, L., Crichton, D., Mattmann, C., Harney, J., Shipman, G., Wang, F., Ananthakrishnan, R., Miller, N., Denvil, S., Morgan, M., Pobre, Z., Bell, G. M., Doutriaux, C., Drach, R., Williams, D., Kershaw, P., Pascoe, S., Gonzalez, E., Fiore, S., Schweitzer, R., 2014. The Earth System Grid Federation: An open infrastructure for access to distributed geospatial data. *Future Generation Computer Systems*, 36, 400-417.

Daly, B., Ranwashe, F., 2023. South Africa's initiative toward an integrated biodiversity data portal. *Frontiers in Ecology and Evolution*, 11.

Dončević, J., Fertalj, K., Brčić, M., Krajna, A., 2023. Mask–Mediator–Wrapper: A Revised Mediator–Wrapper Architecture for Heterogeneous Data Source Integration. *Applied Sciences 2023, Vol. 13, Page 2471*, 13, 2471. https://www.mdpi.com/2076-3417/13/4/2471/htm https://www.mdpi.com/2076-3417/13/4/2471.

Duque, J. P., Brovelli, M. A., 2022. BUILDING A DIGITAL TWIN OF THE ITALIAN COASTS. *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences - ISPRS Archives*, 48, 127-133.

Huang, C. Y., Liang, S., 2014. A sensor data mediator bridging the OGC Sensor Observation Service (SOS) and the OASIS Open Data Protocol (OData). *Annals of GIS*, 20, 279-293.

Johnson, P. A., Sieber, R., Scassa, T., Stephens, M., Robinson, P., 2017. The Cost(s) of Geospatial Open Data. *Transactions in GIS*, 21, 434-445.

Juarez, J. D., Schick, M., Puechmaille, D., Stoicescu, M., Saulyak, B., 2023. Destination Earth Data Lake. *EGU23*. https://meetingorganizer.copernicus.org/EGU23/EGU23-7177.html.

Nativi, S., Mazzetti, P., Craglia, M., 2021. Digital Ecosystems for Developing Digital Twins of the Earth: The Destination Earth Case. *Remote Sensing 2021, Vol. 13, Page 2119*, 13, 2119. https://www.mdpi.com/2072-4292/13/11/2119/htm https://www.mdpi.com/2072-4292/13/11/2119.

Noardo, F., 2022. Multisource spatial data integration for use cases applications. *Transactions in GIS*, 26, 2874-2913.

OGC API, n.d. https://ogcapi.ogc.org/ (Accessed: 12 July 2023).

Osorio, E. E. C., Hayat, B., Kim, K. H., Kim, K. I., 2017. Geospatial data system architecture for disaster risk management. *International Journal of Grid and Distributed Computing*, 10, 39-52.

Prudhomme, C., Homburg, T., Ponciano, J. J., Boochs, F., Cruz, C., Roxin, A. M., 2019. Interpretation and automatic integration of geospatial data into the Semantic Web. *Computing 2019 102:2*, 102, 365-391. https://link.springer.com/article/10.1007/s00607-019-00701-y.

Rienzi, B., Sosa, R., Abellá, G., Machado, A., Susviela, D., González, L., 2023. Standards-Based Geospatial Services Integration for Smart Cities Platforms. *International Conference on Geographical Information Systems Theory, Applications and Management, GISTAM - Proceedings*, 2023-April, 167-175.

Saeed, Z. O., Mancini, F., Glusac, T., Izadpanahi, P., 2022. Future City, Digital Twinning and the Urban Realm: A Systematic Literature Review. *Buildings 2022, Vol. 12, Page 685*, 12, 685. https://www.mdpi.com/2075-5309/12/5/685/htm https://www.mdpi.com/2075-5309/12/5/685.

Santos, N. D., Gonçalves, G., Coutinho, P., 2018. Location intelligence for augmented smart cities integrating sensor web and spatial data infrastructure (SmaCiSENS). *GISTAM 2018 - Proceedings of the 4th International Conference on Geographical Information Systems Theory, Applications and Management*, 2018-March, 282-289. https://run.unl.pt/handle/10362/46400.

Scerri, S., Tuikka, T., de Vallejo, I. L., Curry, E., 2022. Common European Data Spaces: Challenges and Opportunities. *Data Spaces*, 337-357. https://link.springer.com/chapter/10.1007/978-3-030-98636-0$_1$6.

Shahat, E., Hyun, C. T., Yeom, C., 2021. City digital twin potentials: A review and research agenda. *Sustainability (Switzerland)*, 13.

Wilkinson, M. D., Dumontier, M., Aalbersberg, I. J., Appleton, G., Axton, M., Baak, A., Blomberg, N., Boiten, J. W., et al., 2016. The FAIR Guiding Principles for scientific data management and stewardship. *Scientific Data 2016 3:1*, 3, 1-9. https://www.nature.com/articles/sdata201618.

Xie, J., Li, G., 2018. Distributed geospatial data infrastructure for heterogeneous disaster data integration and application. 1-4. http://supersites.earthobservations.org/.