Spatial Data Lake for Smart Cities: From Design to Implementation

Rodrique Kafando, Rémy Decoupes, Lucile Sautot, Maguelonne Teisseire

AGILE Association of Geographic Information Laboratories in Europe

Spatial Data Lake for Smart Cities: From Design to Implementation

Rodrique Kafando[1,2,3], Rémy Decoupes[1], Lucile Sautot[3], and Maguelonne Teisseire[1]

[1] INRAE, UMR TETIS, Université de Montpellier, France
firstname.lastname@inrae.fr,
[2] Montpellier Méditerranée Métropole (3M), France
[3] AgroParisTech, Montpellier, France
lucile.sautot@agroparistech.fr

Abstract. In this paper, we propose a methodology for designing data lake dedicated to Spatial Data and an implementation of this specific framework. Inspired from previous proposals on general data lake Design and based on the Geographic information — Metadata normalization (ISO 19115), the contribution presented in this paper integrates, with the same philosophy, the spatial and thematic dimensions of heterogeneous data (remote sensing images, textual documents and sensor data, etc). To support our proposal, the process has been implemented in a real data project in collaboration with Montpellier Métropole Méditerranée (3M), a metropolis in the South of France. This framework offers a uniform management of the spatial and thematic information embedded in the elements of the data lake.

Keywords: Spatial Data Management, Geographic Information Metadata, Territories

1 Introduction

According to [2], Smart Cities are intensive and advanced high-tech cities that connect people, information and elements of the city using new technologies to create a sustainable, greener city, competitive and innovative trade, and a better quality of life. In recent years, we have faced an exponential growth of new technologies and associated services within smart cities [23,1]. All these services generate large amounts of data which characterize, from a global point of view, the evolution and behavior of a Smart Territory.

In such a context, the adventure described in this paper was born through a collaboration of a multidisciplinary research lab with Montpellier Métropole Méditerranée (3M), a metropolis in the South of France. The main issue expressed by the end-users was how to catch the semantics, the meaning of the large amount of data available. Some data are produced by the citizens, some by the city supervision staff, by the transport department, the tourism department, and complaining that it never ends. The main drawback of existing tools is the fact that they don't offer flexible ways for massive and heterogeneous data exploration by end-users. In particular, Data Warehouses [9] are too strict to allow the end-users some fancy new analysis that are

AGILE: GIScience Series, 1, 2020.
Full paper Proceedings of the 23rd AGILE Conference on Geographic Information Science, 2020.
Editors: Panagiotis Partsinevelos, Phaedon Kyriakidis, and Marinos Kavouras
This contribution underwent peer review based on a full paper submission.
https://doi.org/10.5194/agile-giss-1-8-2020

not already defined [27]. To overcome this issue, data lake [16] is a new fashion way for data management with total or partial storage of the associated elements (data and metadata). In this new wave, there is a lack of methodological proposals on the design of such data infrastructure considering that it requires more technical skills than design skills. There is still a lack of methodologies or success stories in the data lake design domain. In this paper, we particularly address this topic, starting from large heterogeneous data with a strong spatial dimension and elaborating on how to design and implement this specific case study. Inspired from previous works on spatial data normalization [19] and data lake design [40,27,36], we develop a specific methodology and the associated implementation code shared with the community[4]. We demonstrate that data lake infrastructures are not only expert dedicated but could be end-users oriented by offering a suitable query interface.

This paper is organized as follows. In Section 2, we present the definitions and the works related to data lake design and spatial information management. The proposed methodology is detailed in Section 3 followed by the implementation description in Section 4. Section 6 concludes with the discussions and future works on Spatial data lake Solution.

2 Related Work

Many data management methods have emerged under the advent of Big Data [28]. These are essentially NoSQL databases (Not Only SQL) [7], data warehouses [21,31] and data lakes [35,16].

2.1 Data warehouse and Data Lake

Data warehouses have been designed as an optimization of relational databases for querying, and are used to support decision making in organization. Conceptual models of data warehouses are based on the following concepts: facts and measures, dimension, hierarchies and members [22]. In fact, designing a data warehouse is defining a space of possible cross-tabulations, that will be used by users to explore the data. Data warehouses enable the easy exploration of a large dataset by users. But the implementation of a data warehouse implies the normalization (and the automation of this normalization via ETL (Extract Transform Load) processes) of each entering data, from various data sources. Despite some propositions, the integration of documents and satellite images into a data warehouse is not a simple task. In summary, the implementation of a data warehouse goes a long way towards normalization of data.
Data Lake definitions have been introduced in [11], a detailed comparison with data warehouses is proposed on [27] and revisited in [36]. Data lake is a recent solution that has emerged to meet the needs related to the management and use of big data on which the data warehouse have shown their weaknesses. The main problem being related to the management of the heterogeneous nature of data. A data lake is a storage structure

[4] https://github.com/aidmoit/

AGILE: GIScience Series, 1, 2020.
Full paper Proceedings of the 23rd AGILE Conference on Geographic Information Science, 2020.
Editors: Panagiotis Partsinevelos, Phaedon Kyriakidis, and Marinos Kavouras
This contribution underwent peer review based on a full paper submission.
https://doi.org/10.5194/agile-giss-1-8-2020 | © Authors 2020. CC BY 4.0 License.

allowing to store massive data, from different sources in their native format without the need to perform processing beforehand [35,16]. According to [36], a data lake is a scalable storage system and analysis of data of all types, stored in their native format and essentially intended for data specialists who are among others statisticians, analysts and data scientists. The main characteristics associated with data lakes are:

- metadata catalog to facilitate the access and to reinforce the quality of data,
- policies and governance tools,
- accessibility for users,
- management of evolving items,
- ingestion of any type of data,
- physical and logical organizations.

Being a new Big Data technology, Data Lake is addressed by many studies. Thanks to its ability to handle large volumes of structured and unstructured data, an exploratory study was conducted to improve the understanding of the use of the data lake approach in enterprises context [25]. In [15,29], new architectures of data lakes were designed in order to extract relevant information within heterogeneous data based on their sources. In [33], a Generic and Extensible Metadata Management System for data lakes (called GEMMS) was developed to extract automatically metadata from a wide variety of data sources. In [16], they designed a metadata management system, firstly to extract metadata from data sources, and secondly to enrich data sources by using semantic information coming from whole data and metadata. A wide number of metadata system are proposed by the community, but data lake data management still faces some challenges that have to be overcome [30].

We define a data lake as a storage structure composed of datasets, having parts of previously cited characteristics as detailed in Section 3.

2.2 Geographical Information

Several definitions are associated with the concept of territory depending to the study domain. Among them, according to [38], a territory is the combination of three dimensions: geographical space, time and social relationship. The territory is thus defined as a complex system located in a specific geographical space that emerges from the co-evolution of a bundle of heterogeneous processes (anthropological-cultural, relational, cognitive and economic-productive) that characterizes that space in a unique and unrepeatable way. By taking into account these existing definitions, we define the concept of territory as:

- a set of physical and/or legal actors. Physical in the sense that it is inhabited by one or more groups of people interacting with each other, and legal in the sense that it is composed of several political, economic organizations, etc.
- described by a set of geographic information, namely spatial entities, thematic and temporal entities that interact with each other. This information evolving in time and space.

In this work, we mainly focus on the geographic information produced and managed by the city. We thus base our proposal on the spatial data normalization [19] to support the spatial dimension.

AGILE: GIScience Series, 1, 2020.
Full paper Proceedings of the 23rd AGILE Conference on Geographic Information Science, 2020.
Editors: Panagiotis Partsinevelos, Phaedon Kyriakidis, and Marinos Kavouras
This contribution underwent peer review based on a full paper submission.
https://doi.org/10.5194/agile-giss-1-8-2020 | © Authors 2020. CC BY 4.0 License.

3 Spatial Data Lake Design: a proposal

To the best of our knowledge, there is no studies dealing specifically with the design of a Spatial Data Lake. In the rest of this paper, we present a new methodology in order to provide to the users a guideline for conceiving and implementing such a framework. The design is performed to store the data produced and used by the french metropolis: Montpellier Métropole Méditerranée (3M). Our case study dictates some constraints:

- spatial description of datasets and spatial analyses are essential, particularly, we need to store satellite images,
- the proposed system should be inter-operable with other local, national and European systems,
- the users need to explore the data lake in order to find relevant data, and eventually discover new ones.

The proposed system is composed of three main parts: the data section, the metadata section and the inter-metadata section. The data section is the core storage structure, based on Hadoop Distributed File System (HDFS) [37]. The metadata section is a data catalog [24], describing the data stored in the data lake. The inter-metadata section is a part of the metadata section, that enables the storage of richly described relationships between the data in the data lake.

The HDFS is an efficient system to store big data, but cannot be used alone by our users. The users of a data lake need to explore data in order to find the most relevant data regarding to their query, and maybe discover new data, new knowledge. These functions (exploration, querying, discovery) are offered by the data catalog, by providing to the data lake users interesting metadata presented with a user friendly interface.

The proposed conceptual model is an extension of the norm ISO 19115 [19]. This norm includes spatial representations and is the basis of several metadata profiles (INSPIRE, Dublin core) used by public institutions.

In the following, we present our proposal of extension. See Figure 1 to have an overview of the proposed conceptual model. The white classes come from [19], and the yellow classes are our additions. In order to lighten the figures, we represent only the classes and not the details containing with attributes.

In the Data section (see Figure 2), we define a data lake as a set of resources. A resource can be a service (see ISO 19115) or a data series. A data series is composed of one (or more) datasets, that share a feature. A dataset is a collection of identifiable data. Three types of particular dataset are defined: document, vector and raster.

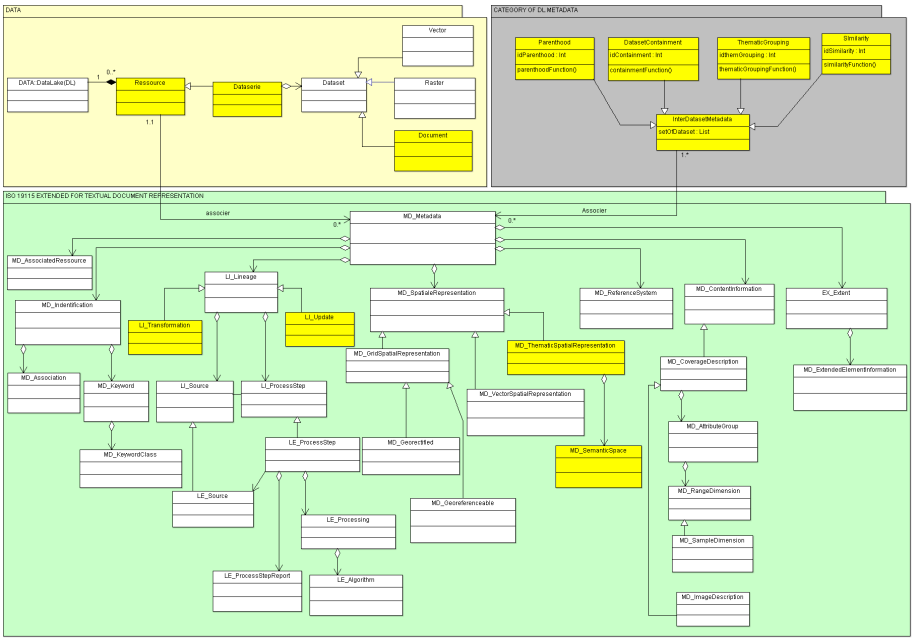
The Metadata section describes metadata records (see Figure 3). Each resource is associated to a metadata record. One metadata record is composed by:

- an identification (mandatory) that enables the recognition of each resource by users,
- a spatial representation (optional), a reference system (optional) and a spatial and/or temporal extent (optional) that describes the spatiality of the resource,
- a content description (optional),
- a lineage (optional), that explains how the resource has been obtained,

AGILE: GIScience Series, 1, 2020.
Full paper Proceedings of the 23rd AGILE Conference on Geographic Information Science, 2020.
Editors: Panagiotis Partsinevelos, Phaedon Kyriakidis, and Marinos Kavouras
This contribution underwent peer review based on a full paper submission.
https://doi.org/10.5194/agile-giss-1-8-2020 | © Authors 2020. CC BY 4.0 License.

– one or several associated resources,
– a reference system information (optional), that identifying the spatial, the temporal, and parametric reference system(s) used by a resource,
– an extent, that describes the spatial and temporal extent of a resource.

Finally, the Inter-Metadata section (see Figure 4) records relationships between datasets, and enables users to have a view of data related to their initial query. Four types of relationships are proposed, based on [36]: parenthood, containment, similarity and thematic grouping.



white classes are in [19], yellow classes are added to the norm.

Fig. 1. General view of the proposed conceptual model

4 Implementation for a specific city use-case

In this Section, we aim to report the first implementation of the Spatial Data Lake provided for 3M[5].

4.1 Presentation of the infrastructure

As described in the previous section, the data lake is composed of two parts: Data and Metadata. The Data section is based on a HDFS cluster. Our implementation uses three

[5] https://www.montpellier3m.fr/

AGILE: GIScience Series, 1, 2020.
Full paper Proceedings of the 23rd AGILE Conference on Geographic Information Science, 2020.
Editors: Panagiotis Partsinevelos, Phaedon Kyriakidis, and Marinos Kavouras
This contribution underwent peer review based on a full paper submission.
https://doi.org/10.5194/agile-giss-1-8-2020 | © Authors 2020. CC BY 4.0 License.

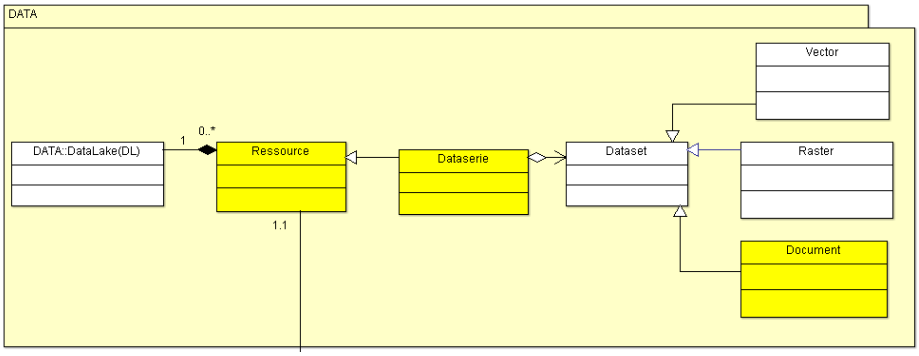


Fig. 2. Section data

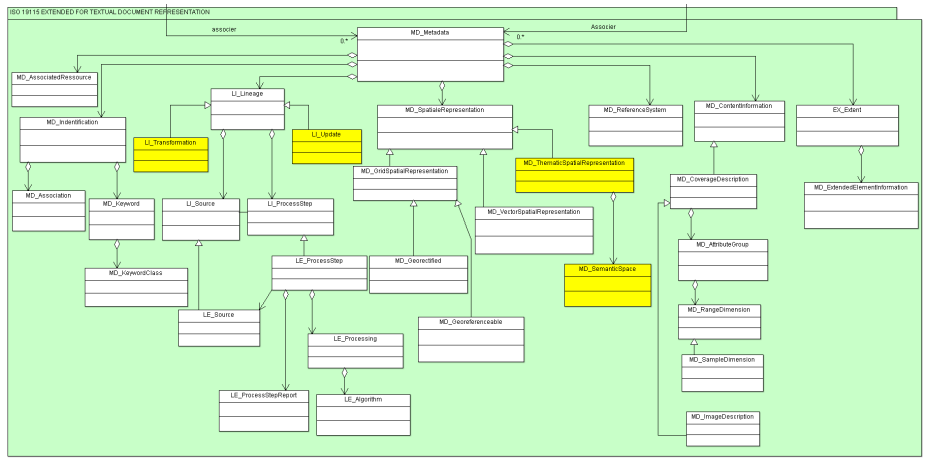


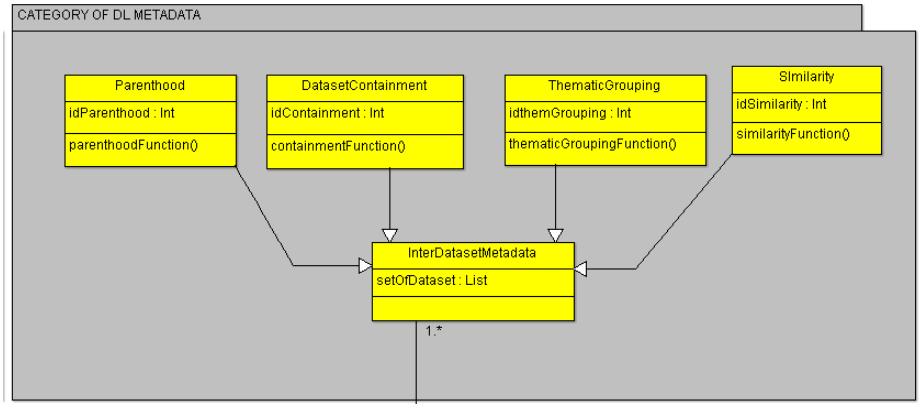
Fig. 3. Section Metadata

HDFS nodes. The first one is the name-node which distributes or aggregates blocks of datasets to the two other data-nodes.

Concerning the Metadata section, HDFS does not provide any indexing system nor a search engine. They have to be build upon the data lake by the administrator [20]. Elasticsearch, based on Apache Lucene, meets those two needs [8] and [20]. Our Metadata System implementation uses GeoNetwork[6]. This web application embeds Apache Lucene and implements the ISO 19115 conceptual model. GeoNetwork stores the mandatory and optional metadata described in the previous section, including the HDFS path inside the Data Lake. So when the user queries the search engine,

[6] GeoNetwork: a catalog application to manage spatially referenced resources. https://geonetwork-opensource.org/

AGILE: GIScience Series, 1, 2020.
Full paper Proceedings of the 23rd AGILE Conference on Geographic Information Science, 2020.
Editors: Panagiotis Partsinevelos, Phaedon Kyriakidis, and Marinos Kavouras
This contribution underwent peer review based on a full paper submission.
https://doi.org/10.5194/agile-giss-1-8-2020 | © Authors 2020. CC BY 4.0 License.



white classes are in [19], yellow classes are added to the norm.

Fig. 4. Section Inter-metadata

GeoNetwork responds with a collection of metadata describing datasets and offers HDFS links to download the corresponding data.

Inserting and Indexing datasets inside the Data Lake As shown in Figure 5, dataset insertion occurs in 5 steps. The administrator has to fill a CSV (Comma separating Values) file [step 1]. Columns are the metadata described in the Spatial Data Lake Design Model section. Lines refer to the datasets which have to be downloaded from the Internet and inserted into the data lake. Then administrator has to run a script [step 2]. This software reads the CSV file [step 3], downloads the datasets from the internet and inserts them inside the HDFS cluster [step 4]. Finally, the software creates metadata files and inserts them into GeoNetwork [step 5].

Discover and access to datasets using the datalake Users can discover and access the datasets using the GeoNetwork search engine. Queries can be a combination of three dimensions:

1. semantic: based on keywords or on a full text search on title, abstract and lineage,
2. spatialized: drawing a bounding box on a map to filter by a geographical extent,
3. temporal: filtering by years, month and day.

GeoNetwork returns a collection of metadata that describe the data. Users can browse these metadata and find HDFS links to download the corresponding dataset (Figure 6).

4.2 Deploy and populate the Data Lake

We have automated the deployment and the data ingestion of the data lake through two steps. First, the system infrastructure must be created, configured and initiated. The data and the metadata zones are atomically deployed on four computers.

AGILE: GIScience Series, 1, 2020.
Full paper Proceedings of the 23rd AGILE Conference on Geographic Information Science, 2020.
Editors: Panagiotis Partsinevelos, Phaedon Kyriakidis, and Marinos Kavouras
This contribution underwent peer review based on a full paper submission.
https://doi.org/10.5194/agile-giss-1-8-2020 | © Authors 2020. CC BY 4.0 License.

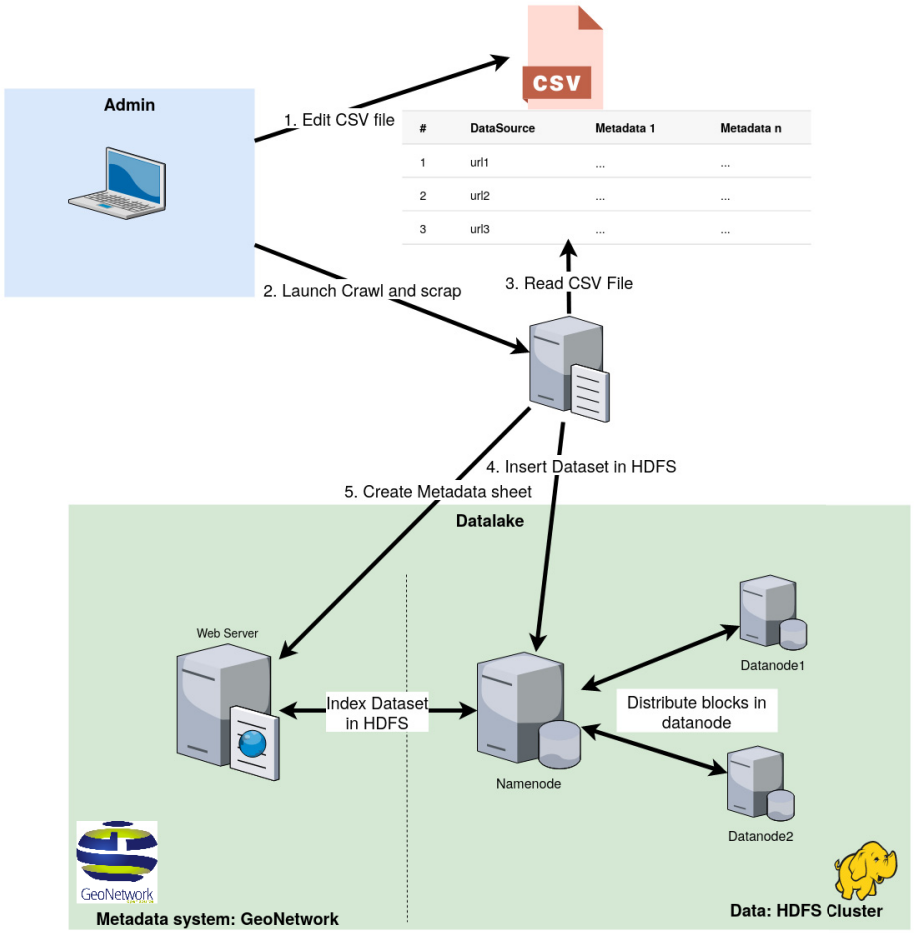


Fig. 5. Inserting and indexing datasets in datalake

Secondly, a python script must be started. Reading a CSV file, it will find links and data sources available on the Internet. Thanks to these information, the script will download data, insert them inside the data lake and reference them through GeoNetwork server.

Deploy a HDFS cluster and a GeoNetwork server The full cluster is deployed and maintained using these opensource projects:

- Debian: Operating System for the virtual machines. The version 9 is used, instead of the last one (version 10), because of compatibility reasons. Indeed, Hadoop needs java version 9, which is no longer available in the latest version of debian,
- VirtualBox as a hypervisor,
- Vagrant as a configuration manager of virtual machines,

AGILE: GIScience Series, 1, 2020.
Full paper Proceedings of the 23rd AGILE Conference on Geographic Information Science, 2020.
Editors: Panagiotis Partsinevelos, Phaedon Kyriakidis, and Marinos Kavouras
This contribution underwent peer review based on a full paper submission.
https://doi.org/10.5194/agile-giss-1-8-2020 | © Authors 2020. CC BY 4.0 License.

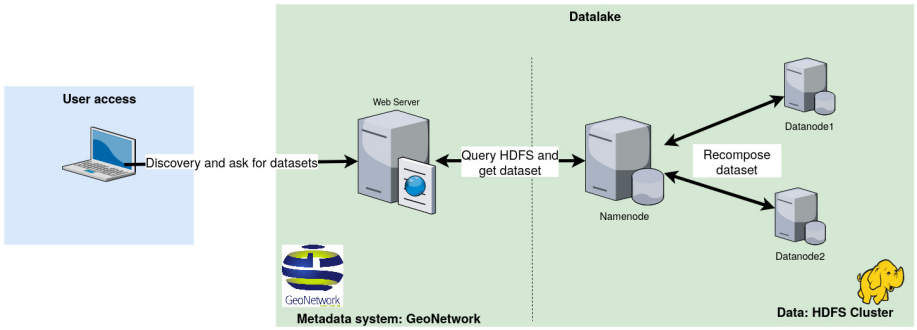


Fig. 6. Discover and query datasets

– Ansible as a software provisioning, configuration manager and application-deployment tool.

The HDFS cluster can be deployed and configured using a single vagrant command line. Then the administrator has to connect to the namenode in order to format the file system. Afterwards, GeoNetwork can be deployed using a similar vagrant command line. After starting the two parts (the metadata Management system and the HDFS cluster) of the data lake, four virtual machines are started and set-up, three for the hdfs cluster and one for the metadata management system.

More technical information or instructions on how to deploy a HDFS Cluster with a GeoNetwork as a metadata system management, could be found in the README.md inside the git repository of our project.

If default variables are used, the data lake file system can be browsed graphically by connecting to the web server of the namenode at:
http://10.0.0.10:9870. Other information, such as cluster health and log accessibility are also available.

Populate the Data Lake The populating step is implemented by using two scripts written in python and R. Indeed, Python offers an excellent library to interact with HDFS, while R has interesting modules to manage ISO 19115 metadata. In order to reduce the complexity generated by the concomitant use of these two languages, the R script has been encapsulated inside the Python script. Thus, the administrator only needs to run the Python script. As mentioned above, all the code files are available (see section 4.4). Environment requirements (as dependencies) can be found in requirement.txt file inside the repository. Instructions on how to install and start the script are described in the README.md file.

The main Python script works on five steps. First, it parses information given by 'datasources.csv' such as data provider, dataset name and keywords. Secondly, script browses data provider's website in order to build a json file that contains web-links to download the corresponding data. Then, these data files are downloaded and stored inside the data lake. Finally, the R script is executed in order to create ISO 19139 (which

AGILE: GIScience Series, 1, 2020.
Full paper Proceedings of the 23rd AGILE Conference on Geographic Information Science, 2020.
Editors: Panagiotis Partsinevelos, Phaedon Kyriakidis, and Marinos Kavouras
This contribution underwent peer review based on a full paper submission.
https://doi.org/10.5194/agile-giss-1-8-2020 | © Authors 2020. CC BY 4.0 License.

is a standard of the ISO 19115 implementation into xml) xml files which are uploaded to the GeoNetwork.

In Figure 7, a screenshot of a metadata sheet of a dataset is presented. A set of data files is associated with a name and a link to download the data from the HDFS cluster.

Data can be easily obtained by following the given link. The namenode of the HDFS cluster offers a REST API that ensures the transfer of the data to the user.

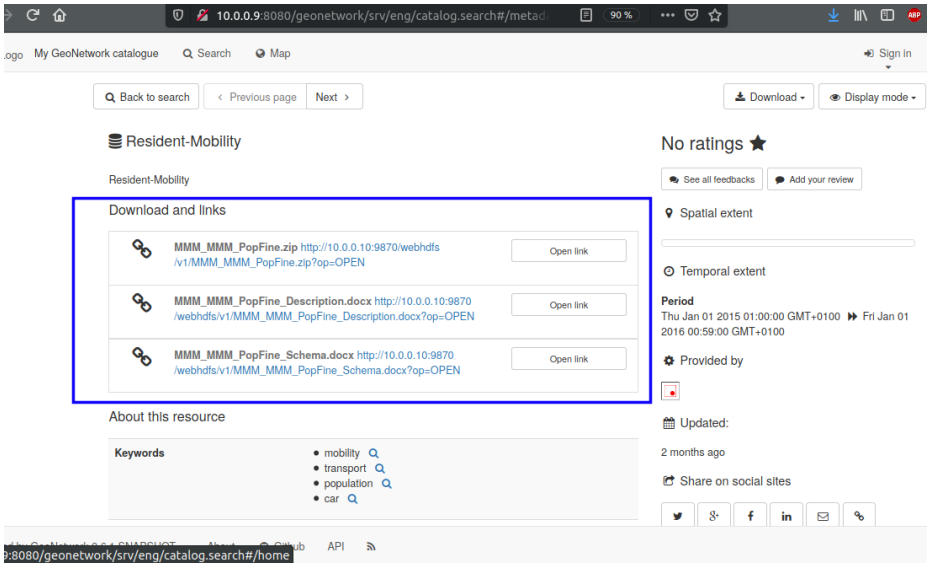


Fig. 7. Set of data files with HDFS path for a dataset

4.3 Example of an user query

The user can build complex queries mixing the three dimensions: spatial, temporal and semantic. These queries are made through the GeoNetwork search engine. The three dimensions can be requested in a full text query, such as the following example: "3M mobility 2013". The search engine will propose all datasets which the temporal period includes or intersects the year 2013 and which the spatial extent includes or intersect 3M spatial extent. Finally, only the datasets that have the keyword "mobility" in their metadata, will be shown to the user.

The spatial dimension can also be built by using a map. The user can draw a bounding box around a region. She/He gets a first filtering on the selected extent. She/He can enrich his query by adding in full text temporal and semantic constraints.

4.4 Data and Software

Computational environment The computational environment (HDFS clusters and GeoNetWork) is provided by using four virtual machines. The implementation and

AGILE: GIScience Series, 1, 2020.
Full paper Proceedings of the 23rd AGILE Conference on Geographic Information Science, 2020.
Editors: Panagiotis Partsinevelos, Phaedon Kyriakidis, and Marinos Kavouras
This contribution underwent peer review based on a full paper submission.
https://doi.org/10.5194/agile-giss-1-8-2020 | © Authors 2020. CC BY 4.0 License.

deployment of these machines has been automated and scripts are available at: https://github.com/aidmoit/ansible-deployment, with instructions included in the file README.md in the repository. The corresponding commit number is: 65de950a336ee2828cdb19db976b7946649c439c and the repository is published under GPL-3 license.

software All software for retrieving data, ingesting them in the HDFS clusters and their descriptions in the GeoNetwork is orchestrated by a python script. Implementation resources are available through this repository: https://github.com/aidmoit/collect.
The corresponding commit number is: da9f63f9287a191d7e8fd24884a731bae02e1034 and the repository is published under GPL-3 license. They operate two R packages: geometa [6] and geonapi [5] published under MIT licence. Finally, scripts are using OpenStreetMap[7] data, which are published under the "Open Data Commons Open Database License" (ODbL) licence.

datasets Research data supporting this publication come from three main data providers: 3M's open-data platform, a satellite image provider and OpenStreetMap :

- Multiple data from 3M opendata webservice, as local urban planning, mobility reports, land cover, population distribution, accessible on: https://data.montpellier3m.fr/ under the Open Data Commons Open Database License (ODbL),
- Images from Earth remote sensing satellite are provided by Geosud and accessible on http://ids.equipex-geosud.fr/web/guest/catalog1 for only visualization purposes,
- We use OpenStreetMap to geocode place names around our subject of study.

5 Discussion

In this section, we discuss about the main features (enumerated in [36]) provided by our proposal in comparison to existing solutions. Among these features, we notice:

- Semantic Enrichment (SI): offers semantic data related to the data in the Lake, either by context description or by descriptive tags.
- Data Indexing (DI): possibility to index data in the lake by using some keywords or indexes, etc.
- Link Generation and conservation (LG): allows to have relationships between data, either on the thematic, the source, etc.
- Data Polymorphism (DP): allows to manage the storage of the same data in different formats, by taking into account the different process carried out on it.
- Data Versioning (DV): allows to keep the history of process carried out on a data, while keeping the different states of the data.
- User Tracking (UT): allows to track the activities of each user, keeping his identity, the type of processing carried out, the data used, etc.

AGILE: GIScience Series, 1, 2020.
Full paper Proceedings of the 23rd AGILE Conference on Geographic Information Science, 2020.
Editors: Panagiotis Partsinevelos, Phaedon Kyriakidis, and Marinos Kavouras
This contribution underwent peer review based on a full paper submission.
https://doi.org/10.5194/agile-giss-1-8-2020 | © Authors 2020. CC BY 4.0 License.

Table 1. Features provided by data lake metadata systems

System	Type	SE	DI	LG	DP	DV	UT
SPAR [14]	♦ □	✓	✓	✓			✓
Alrehamy and Walker [42]	♦	✓		✓			
Data wrangling [41]	♦	✓	✓			✓	✓
Constance [16]	♦	✓	✓				
GEMMS [32]	◊	✓					
CLAMS [12]	♦	✓					
Suriarachchi and Plale [40]	♦				✓		✓
Singh, K. et al. [39]	♦	✓	✓	✓	✓		
Farrugia et al. [13]	♦			✓			
GOODS [17]	♦	✓	✓	✓		✓	
CoreDB [3]	♦		✓				✓
Ground [18]	◊ □	✓	✓			✓	✓
KAYAK [26]	♦	✓	✓	✓			
CoreKG [4]	♦	✓	✓	✓	✓		✓
Diamantini et al. [10]	◊	✓		✓	✓		
Ravat, F., Zhao, Y. [34]	♦ ◊	✓		✓	✓	✓	✓
MEDAL [36]	◊	✓	✓	✓	✓	✓	✓
Our proposal	♦ ◊	✓	✓	✓	✓	✓	✓

♦ : Data lake implementation
◊ : Metadata model
□ : Model or implementation assimilable to a data lake
✓ : feature is available

Table 1 shows the state-of-the-art approaches and the associated features provided.

Among seventeen (17) proposals, only one approach [34] proposes a data lake implementation associated with a metadata management system. Moreover, this approach, like the majority, has been set up for a very specific case study, and does not allow, or hardly takes into account, the case of complex data such as spatial data (satellite data). In terms of completeness of the state-of-the-art, regarding the features mentioned above, approaches such as [4,18,17,34,36,41] respectively cover more than the half of them. Unfortunately, these solutions are difficult to apply, since they are not very understandable, in the implementation ways, or have been proposed but not yet implemented.

Our solution does not only offer a data lake implementation, but also an associated metadata management system. It clearly shows how the two concepts are integrated to each other with a fully open source code. It also covers all the features described in Table 1, while implementing the data lake and the metadata management system. Moreover, our approach solves problems related to complex data storage (e.g. spatial), and takes into account any type of data, thanks to the ISO 19115 standard. Also, it is easily reproducible and compatible with international catalogs systems. In terms of access, our system offers through GeoNetwork, an user interface that allows any user

[7] OpenStreeMap: https://www.openstreetmap.org

AGILE: GIScience Series, 1, 2020.
Full paper Proceedings of the 23rd AGILE Conference on Geographic Information Science, 2020.
Editors: Panagiotis Partsinevelos, Phaedon Kyriakidis, and Marinos Kavouras
This contribution underwent peer review based on a full paper submission.
https://doi.org/10.5194/agile-giss-1-8-2020 | © Authors 2020. CC BY 4.0 License.

to explore or retrieve data from the lake.

Our solution can be used by two types of users. Firstly, the 'general public' users or any user, the system allows them to explore and retrieve data available in the lake through the web interface offered by GeoNetwork without the need of data lake exploitation skills. Then advanced users, who in addition to exploring, can perform processing and analysis directly thought the data lake using tools such as Apache Spark.

6 Conclusion and Perspectives

In this paper, we presented a new methodology for Spatial Data Lake Design. The main contributions are the introduction of the spatial dimension in the data lake design process based on the Geographic Information Metadata as well as an overall code process provided to the scientific community. We also showed that a data lake could be end-users oriented with a specific query interface. Future works are dedicated to better manage the evolution and the behavior of a territory. These concern mainly these two objectives:

1. How heterogeneous data can be linked semantically for an analysis of complex spatio-temporal phenomena on a territory,
2. Define original data mining techniques mainly suited for the processing and analysis of this massive heterogeneous data.

Achieving these objectives will allow us to describe the relationships between the themes taking into account the spatio-temporal aspect on the one hand, and on the other hand to show how, these themes contribute to the description of the territory evolution. In other words, Spatial data lake is becoming a fundamental element to reach Smart Territories.

Acknowledgments

This work was partially supported by Montpellier Méditerranée Métropole and by Songe Project (FEDER and Occitanie Region). The experiments were also supported by public funds received in the framework of GEOSUD, a project (ANR-10-EQPX-20) of the program "Investissements d'Avenir" managed by the French National Research Agency

References

1. Al Nuaimi, E., Al Neyadi, H., Mohamed, N., Al-Jaroodi, J.: Applications of big data to smart cities. Journal of Internet Services and Applications **6**(1), 25 (2015)
2. Albino, V., Berardi, U., Dangelico, R.M.: Smart Cities: Definitions, Dimensions, Performance, and Initiatives. Journal of Urban Technology **22**(1), 3–21 (2015). DOI 10.1080/10630732.2014.942092. URL http://www.tandfonline.com/doi/full/10.1080/10630732.2014.942092

3. Beheshti, A., Benatallah, B., Nouri, R., Chhieng, V.M., Xiong, H., Zhao, X.: Coredb: a data lake service. In: Proceedings of the 2017 ACM on Conference on Information and Knowledge Management, pp. 2451–2454 (2017)
4. Beheshti, A., Benatallah, B., Nouri, R., Tabebordbar, A.: Corekg: a knowledge lake service. Proceedings of the VLDB Endowment **11**(12), 1942–1945 (2018)
5. Blondel, E.: geonapi: R interface to geonetwork api (2018). DOI 10.5281/zenodo.1345102. URL https://doi.org/10.5281/zenodo.1345102
6. Blondel, E.: geometa: Tools for Reading and Writing ISO/OGC Geographic Metadata in R (2019). DOI 10.5281/zenodo.3524348. URL https://doi.org/10.5281/zenodo.3524348
7. Bruchez, R.: Les bases de données NoSQL et le BigData: Comprendre et mettre en oeuvre. Editions Eyrolles (2015). Google-Books-ID: SKu0CAAAQBAJ
8. Chen, D., Chen, Y., Brownlow, B.N., Kanjamala, P.P., Arredondo, C.A.G., Radspinner, B.L., Raveling, M.A.: Real-time or near real-time persisting daily healthcare data into hdfs and elasticsearch index inside a big data platform. IEEE Transactions on Industrial Informatics **13**(2), 595–606 (2017). DOI 10.1109/TII.2016.2645606
9. Devlin, B., Cote, L.D.: Data warehouse: from architecture to implementation. Addison-Wesley Longman Publishing Co., Inc. (1996)
10. Diamantini, C., Giudice, P.L., Musarella, L., Potena, D., Storti, E., Ursino, D.: An Approach to Extracting Thematic Views from Highly Heterogeneous Sources of a Data Lake. In: SEBD (2018)
11. Fang, H.: Managing data lakes in big data era: What's a data lake and why has it became popular in data management ecosystem. In: 2015 IEEE International Conference on Cyber Technology in Automation, Control, and Intelligent Systems (CYBER), pp. 820–824 (2015). DOI 10.1109/CYBER.2015.7288049
12. Farid, M., Roatis, A., Ilyas, I.F., Hoffmann, H.F., Chu, X.: Clams: bringing quality to data lakes. In: Proceedings of the 2016 International Conference on Management of Data, pp. 2089–2092 (2016)
13. Farrugia, A., Claxton, R., Thompson, S.: Towards social network analytics for understanding and managing enterprise data lakes. In: 2016 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM), pp. 1213–1220 (2016). DOI 10.1109/ASONAM.2016.7752393
14. Fauduet, L., Peyrard, S.: A data-first preservation strategy: Data management in spar. In: iPRES (2010)
15. Giudice, P.L., Musarella, L., Sofo, G., Ursino, D.: An approach to extracting complex knowledge patterns among concepts belonging to structured, semi-structured and unstructured sources in a data lake. Information Sciences **478**, 606–626 (2019)
16. Hai, R., Geisler, S., Quix, C.: Constance: An intelligent data lake system. In: Proceedings of the 2016 International Conference on Management of Data, pp. 2097–2100 (2016)
17. Halevy, A.Y., Korn, F., Noy, N.F., Olston, C., Polyzotis, N., Roy, S., Whang, S.E.: Managing google's data lake: an overview of the goods system. IEEE Data Eng. Bull. **39**(3), 5–14 (2016)
18. Hellerstein, J.M., Sreekanti, V., Gonzalez, J.E., Dalton, J., Dey, A., Nag, S., Ramachandran, K., Arora, S., Bhattacharyya, A., Das, S., et al.: Ground: A data context service. In: CIDR (2017)
19. ISO TC 211 Information géographique/ Géomatique: Iso 19115-1:2014 geographic information — metadata. Tech. rep., International Organization for Standardization (2019)
20. John, T., Misra, P.: Data Lake for Enterprises: Lambda Architecture for Building Enterprise Data Systems. Packt Publishing (2017)
21. Kimball, R., Ross, M.: The Data Warehouse Toolkit: The Complete Guide to Dimensional Modeling. John Wiley & Sons (2011). Google-Books-ID: XoS2oy1IcB4C

AGILE: GIScience Series, 1, 2020.
Full paper Proceedings of the 23rd AGILE Conference on Geographic Information Science, 2020.
Editors: Panagiotis Partsinevelos, Phaedon Kyriakidis, and Marinos Kavouras
This contribution underwent peer review based on a full paper submission.
https://doi.org/10.5194/agile-giss-1-8-2020 | © Authors 2020. CC BY 4.0 License.

22. Kimball, R., Ross, M.: The data warehouse toolkit: The definitive guide to dimensional modeling. John Wiley & Sons (2013)
23. Kitchin, R.: The real-time city? big data and smart urbanism. GeoJournal **79**(1), 1–14 (2014)
24. Lamb, I., Larson, C.: Shining a light on scientific data: Building a data catalog to foster data sharing and reuse. Code4Lib Journal (32) (2016)
25. Llave, M.R.: Data lakes in business intelligence: reporting from the trenches. Procedia computer science **138**, 516–524 (2018)
26. Maccioni, A., Torlone, R.: Kayak: a framework for just-in-time data preparation in a data lake. In: International Conference on Advanced Information Systems Engineering, pp. 474–489. Springer (2018)
27. Madera, C., Laurent, A.: The next information architecture evolution: the data lake wave. In: Proceedings of the 8th International Conference on Management of Digital EcoSystems, pp. 174–180 (2016)
28. McAfee, A., Brynjolfsson, E., Davenport, T.H., Patil, D., Barton, D.: Big data: the management revolution. Harvard business review **90**(10), 60–68 (2012)
29. Mehmood, H., Gilman, E., Cortes, M., Kostakos, P., Byrne, A., Valta, K., Tekes, S., Riekki, J.: Implementing big data lake for heterogeneous data sources. In: 2019 IEEE 35th International Conference on Data Engineering Workshops (ICDEW), pp. 37–44. IEEE (2019)
30. Nargesian, F., Zhu, E., Pu, K.Q., Miller, R.J., Arocena, P.C.: Data lake management: Challenges and opportunities **12**(12), 4. DOI 10.14778/3352063.3352116
31. Phipps, C., Davis, K.C.: Automating data warehouse conceptual schema design and evaluation. In: DMDW, vol. 2, pp. 23–32. Citeseer (2002)
32. Quix, C., Hai, R., Vatov, I.: Gemms: A generic and extensible metadata management system for data lakes. In: CAiSE Forum, pp. 129–136 (2016)
33. Quix, C., Hai, R., Vatov, I.: Metadata extraction and management in data lakes with gemms. Complex Systems Informatics and Modeling Quarterly (9), 67–83 (2016)
34. Ravat, F., Zhao, Y.: Data Lakes: Trends and Perspectives. In: International Conference on Database and Expert Systems Applications, pp. 304–313. Springer (2019)
35. Russom, P.: Data lakes: Purposes, practices, patterns, and platforms. TDWI White Paper (2017)
36. Sawadogo, P.N., Scholly, E., Favre, C., Ferey, E., Loudcher, S., Darmont, J.: Metadata systems for data lakes: models and features. In: European Conference on Advances in Databases and Information Systems, pp. 440–451. Springer (2019)
37. Shvachko, K., Kuang, H., Radia, S., Chansler, R., et al.: The hadoop distributed file system. In: MSST, vol. 10, pp. 1–10 (2010)
38. Simone, C., Barile, S., Calabrese, M.: Managing territory and its complexity: a decision-making model based on the viable system approach (vsa). Land use policy **72**, 493–502 (2018)
39. Singh, K., Paneri, K., Pandey, A., Gupta, G., Sharma, G., Agarwal, P., Shroff, G.: Visual bayesian fusion to navigate a data lake. In: 2016 19th International Conference on Information Fusion (FUSION), pp. 987–994. IEEE (2016)
40. Suriarachchi, I., Plale, B.: Crossing analytics systems: A case for integrated provenance in data lakes. In: 2016 IEEE 12th International Conference on e-Science (e-Science), pp. 349–354. IEEE (2016)
41. Terrizzano, I.G., Schwarz, P.M., Roth, M., Colino, J.E.: Data wrangling: The challenging yourney from the wild to the lake. In: CIDR (2015)
42. Walker, C., Alrehamy, H.: Personal data lake with data gravity pull. In: 2015 IEEE Fifth International Conference on Big Data and Cloud Computing, pp. 160–167. IEEE (2015)

AGILE: GIScience Series, 1, 2020.
Full paper Proceedings of the 23rd AGILE Conference on Geographic Information Science, 2020.
Editors: Panagiotis Partsinevelos, Phaedon Kyriakidis, and Marinos Kavouras
This contribution underwent peer review based on a full paper submission.
https://doi.org/10.5194/agile-giss-1-8-2020 | © Authors 2020. CC BY 4.0 License.