CHIS: A Big Data Infrastructure to Manage Digital Cultural Items

Aniello Castiglione[a,*], Francesco Colace[b], Vincenzo Moscato[c], Francesco Palmieri[a]

[a] *University of Salerno, Dipartimento di Informatica, Via Giovanni Paolo II, 132 - 84084 Fisciano (SA), Italy*
[b] *University of Salerno, Dipartimento di Ingegneria Industriale, Via Giovanni Paolo II, 132 - 84084 Fisciano (SA), Italy*
[c] *University of Naples, Dip. di Ingegneria Elettrica e delle Tecnologie dell'Informazione, Via Claudio 21, - 80125 Napoli, Italy*

Abstract

In this paper, we describe CHIS (*Cultural Heritage Information System*), a big data infrastructure that can be used to query, browse, analyze and process digital contents related to cultural heritage from a set of heterogeneous and distributed repositories. CHIS is characterized by the following technical features: capability to gather information from distributed and heterogeneous data sources (e.g., Sensor Networks, Social Media Networks, Digital Libraries and Archives, Multimedia Collections, Web Data Services, etc.); advanced data management techniques and technologies; ability to provide useful and personalized data to users based on their preferences and context; advanced information retrieval facilities, data analytics and other utilities/services, according to the SOA paradigm. By means of a set of ad-hoc APIs, and value-added data processing and analytics services, our system can support several applications: mobile multimedia guides for cultural environments, web portals to promote the cultural heritage of a given organization, multimedia recommender and storytelling systems and so on. We discuss the main ideas that characterize the system, showing its use for several applications.

Keywords: Big Data, Cultural Heritage, Resource Management

1. Introduction

Italy boasts one of the largest and most priceless cultural heritage in the world. This invaluable resource can be opportunely protected, preserved and promoted by "embedding" it in the digital ecosystem of a *Smart City*, where economic, tourist, recreational and logistic aspects have to be considered all together.

As highlighted by recent publications and reiterated by European projects discussing Smart Cities [1, 2, 3, 4], the adoption of *Future Internet* technologies, particularly the paradigms of the *Internet of Things* and *Internet of Services*, now represents the "de facto standard" in the design and implementation of IT platforms that can provide effective support to the "smartness" of a city. In such a context, it is possible to design *context-aware* services that take into account both the surrounding environment, whose state is captured by sensors, and the characteristics of the users. These services are then all accessible through a *Cloud Computing* environment [5]. In addition, according to the vision of *participatory sensing*, [6] mobile devices of the latest generations (e.g., smart-phones, tablets, etc.) form an interactive network that allows users to access, analyze and share information and knowledge with an "active" role. Eventually, following the recommendations of the W3C *Semantic Web* framework [7, 8], in order to allow the enormous amount of data collected (*Big Data*[9, 10]) in a smart environment to be used by different applications, the data must be properly processed and stored in the form of *linked open data*, in order to facilitate both access and semantic processing.

A number of proposals, which focus on how the discussed technological solutions should be applied to the cultural domain, has been already presented for the Italian heritage.

*Corresponding author: Aniello Castiglione, Department of Computer Science, University of Salerno, Via Giovanni Paolo II, 132 I-84084 Fisciano (SA), Italy - Phone: +39089969594, Fax: +39089969821, email: castiglione@acm.org, castiglione@ieee.org

Email addresses: castiglione@ieee.org (Aniello Castiglione), fcolace@unisa.it (Francesco Colace), vmoscato@unina.it (Vincenzo Moscato), fpalmieri@unisa.it (Francesco Palmieri)

For example, in the Technological District for Cultural Heritage and Activities, Lazio - DTC[1], a platform for the access to cultural heritage has been proposed and is focused on tourists through the realization of digital scenes, virtual reconstructions, augmented reality techniques and mobile applications.

Similarly, the Calabria region, in the context of the MESSIAH project[2], has proposed enabling methods and multi-functional technologies to support cultural heritage identification, monitoring and restoration.

Indeed, the problem of evaluation and care of cultural heritage through smart city enabling technologies, is one of the most important issues that has to be taken into consideration, not only at national level, but also within the European scenario.

As a first example, Niker[3] addresses the problem of the protection of cultural heritage sites from the effects of seismic phenomena, through an integrated approach based on the use of a Knowledge Base; in turn, H-KNOW [4] proposed a similar approach for the restoration of buildings.

Other projects, based on *Knowledge Management* and *Knowledge Discovery* techniques aim to promote cultural heritage. Significant examples are: SMARTMUSEUM[5] which implements a platform for "knowledge exchange", and PAPYRUS[6], which address the problem of "dynamic" knowledge discovery from digital libraries and news archives.

Several projects, recently funded by the European community, propose methodologies and the best ways to manage and organize the knowledge related to cultural heritage.

ARIADNE[7] aims to create an application infrastructure for the management of archaeological data to enable archaeologists and scholars of the ancient world to access the digital archives of European countries online and to be able to use new technologies as an element of the methodology of archaeological research.

DC-NET (Digital Cultural heritage NETwork)[8] aims to develop and strengthen coordination between public research programs in the European countries involved in the field of digital cultural heritage, through the development of data infrastructure and services dedicated to virtual community research in digital cultural heritage.

MeLa (European Museums and Libraries in/of the Age of Migrations)[9] aims to develop new strategies to organize multi-inter-trans-cultural conservation, presentation and transmission of knowledge, in ways and forms that reflect the conditions imposed by the migration of people and ideas in the global world. Several other projects dealing with the digitization and use of cultural heritage have been proposed , with their emphasis on digital libraries and museums (EOD, Europeana 1914-1918 HOPE, Linked Heritage, etc..).

On the other hand, regarding the challenges in the management of big data, in recent years several projects (EUDAT, iCordi, EUHIT, smartData, Projectcome, etc..) have been created to propose solutions for the effective and efficient treatment of such data, in particular in the context of scientific data processing. More recent proposals have then investigated the use of big data management techniques and architectures for digital cultural contents, proposing applications [11].

Summarizing, the main research problems that can be correlated to the management of Cultural Heritage digital contents in the Smart Cities context are:

- the adoption of architectural models and standards in the context of Future Internet and Big Data [12, 13];
- the interfacing and communication with the sensors of a site [14, 15, 16, 17];
- the access, retrieval, integration and analysis of information from all data sources and the correlation with spatial data [18, 19];
- the transformation of "captured" data in the form of knowledge and its management [20, 21];
- the localization and tracking of users on a site [22, 23];

[1] http://www.futouring.it/web/filas/distretto-tecnologico
[2] www.culturaeinnovazione.it
[3] http://www.niker.eu/
[4] http://www.hknow.eu/
[5] http://www.smartmuseum.eu/
[6] http://www.ict-papyrus.eu/
[7] http://www.ariadne-infrastructure.eu/
[8] http://www.dc-net.org
[9] http://www.mela-project.eu

- the access to the knowledge based on the user profile, the context and the use of applications [24, 25, 26];
- the analysis of users' opinions from social networks [27].

In this paper, we describe CHIS (*Cultural Heritage Information System*), a system to query, browse and analyze cultural digital contents from a set of distributed and heterogeneous multimedia repositories. In particular, the system prototype has been developed within the DATABENC project [10].

CHIS is able to manage all the digital contents related to *Cultural Items*. More in details, in our vision each *Cultural Heritage environment* (e.g., museums, archaeological sites, old town centers, etc.) is grounded on a set of cultural *Points of Interest* (PoI), which correspond to one or more cultural items (e.g., specific ruins of an archaeological site, sculptures and/or pictures exhibited within a museum, historical buildings and famous squares in an old town center and so on).

In order to meet variety, velocity and volume of the managed information, CHIS is characterized by the following technical features that are typical of a *Big Data* platform [28, 29]:

- capability to gather information from distributed and heterogeneous data sources (e.g., Sensor Networks, Social Media Networks, Digital Libraries and Archives, Multimedia Collections, Web Data Services, etc.);
- advanced data management techniques and technologies;
- ability to provide useful and personalized data to users based on their preferences and context;
- advanced information retrieval services, data analytics and other utilities, according to the SOA paradigm.

Regarding data sources, CHIS can manage the following kinds of data, providing detailed information about the cultural heritage belonging to some specific geographic areas of Campania:

- data coming from sensor networks deployed in a given cultural environment;
- cultural items' descriptions - coming from open web sources (e.g., Wikipedia) or from several digital libraries and archives;
- multimedia data - video, text, image and audio - associated with the various items of interest, retrieved from open (Social Media Networks as Panoramio, Picasa, YouTube, and Flickr) or private collections;
- social data - e.g., user comments and opinions from common on-line social networks like Facebook, Twitter, etc.;
- web service data - any kind of useful data gathered by means of web services.

Thus, CHIS provides all the necessary retrieval and presentation functionalities to search information of interest and present it to the users in a suitable format and according to their needs.

By means of a set of ad-hoc APIs, and exploiting the provided services, our system can support several applications: mobile multimedia guides for cultural environments, web portals to promote the cultural heritage of a given organization, multimedia recommender and storytelling systems and so on.

The paper is organized as follows. Section 2 describes the proposed data model. Section 3 presents the system architecture and related functionalities with several implementation details. Section 4 reports a possible application of our system to support a mobile multimedia guide for the archaeological site of Paestum. Eventually, Section 5 discusses some conclusions and the future work.

[10] The High Technology District for Cultural Heritage (DATABENC) management of the Campania Region, in Italy (www.databenc.it).

2. Data Model

As previously described, the introduced data model for the management of cultural contents relies on the concept of "Cultural Item" ($C\mathcal{I}$), that is in turn related to a given "Cultural Environment" (e.g., a museum, an archaeological site, an old town center, an historical building etc.). Examples of $C\mathcal{I}$s are specific ruins of an archaeological site, sculptures and/or pictures exhibited within a museum, historical buildings and famous squares in an old town center and so on.

In the Cultural Heritage domain, a $C\mathcal{I}$ can be opportunely described with respect to a variety of annotation schemata, for example the archaeological view, the architectural perspective, the archivist vision, the historical background, etc., that usually exploit different harvesting sets of "metadata" and possibly domain taxonomies or ontologies [30].

In a simplified way, we consider an *ontology* $O = (V, E)$ as a network of concepts belonging to a given domain of interest, where a node $v \in V$ represents a "concept" and an edge $e \in E$ represents a relationship between two concepts[11].

Thus, we define an *annotation schema* and a *semantic annotation*[31] for a cultural item as follows.

Definition 2.1 (Annotation Schema). *Given a set of ontologies $\mathcal{O}$, an* Annotation Schema *is a particular tuple $\lambda_O = (A_1, \ldots, A_n, B_1, \ldots, B_m)$, where $A_1, \ldots, A_n$ are attributes for which $\forall i \in [1, n], \exists O = (V, E) \in \mathcal{O}$ s.t. $dom(A_i) \subseteq V$ (i.e., A_i assumes values corresponding to nodes of some ontology), and $B_1, \ldots, B_m$ are attributes for which $\forall j \in [1, m], \nexists O = (V, E) \in \mathcal{O}$ s.t. $dom(B_j) \subseteq V$ (i.e., B_j does not assume values corresponding to nodes of any ontology).*

In other words, the attributes $A_1, \ldots, An$ are *Ontological Attributes* (OAs) and correspond to concepts that are relevant for the specific domain(s) being modeled. In turn, *Non-Ontological Attributes* $B_1, \ldots, B_m$ (NOAs) can contain other useful information, such as measures returned by the sensors attached to cultural items, or multimedia objects (e.g., audio, video, images, texts and 3D models, etc.) characterized by a set of low-level features and other metadata.

In addition, a $C\mathcal{I}$ may be associated with a specific "Point of Interest" ($\mathcal{P}o\mathcal{I}$), defined by a set of geographic coordinates, and corresponding either to a single point or to a set of lines and more complex polygons of the considered environment.

In particular, we can adopt both "literals" or a set of URIs (*Uniform Resource Identifiers*), that allow to access the related cultural information according to the *Linked Data*/*Linked Open Data* (LD/LOD) paradigms, as values of the annotation attributes.

Definition 2.2 (Semantic Annotation). *Given a set of ontologies $\mathcal{O}$, an annotation schema λ_O and cultural item $C\mathcal{I}$, a* Semantic Annotation *of $C\mathcal{I}$ is a tuple $\lambda_O(C\mathcal{I}) = (a_1, \ldots, a_n, b_1, \ldots, b_m)$, where $\forall i \in [1, n], a_i \in dom(A_i)$ and $\forall j \in [1, m], b_i \in dom(B_i)$*

Using various sets of ontologies/taxonomies and semantic annotations, we can thus describe a cultural item from different points of view supporting several applications.

A large set of relationships can also be instantiated among cultural items and the entire system *Knowledge Base* (KB) can be modeled as a particular *graph*.

Definition 2.3 (Knowledge Base). *The* Knowledge Base *is a graph $G = (C, R)$: each node $c \in C$ can be a cultural item or an ontological attribute (concept), while each edge $r \in R$ represents a relationship derived from a semantic annotation or established between two cultural items.*

All possible relationships in the model are opportunely defined "a-priori" and the related meaning can be found in a proper thesaurus.

Figure 1 shows how a portion of knowledge related to the Paestum ruins can be easily represented in our model. In particular, we can see some of the several cultural items a tourist in the Paestum ruins can be interested in: the Archaeological Museum of Paestum (containing evidence from the entire archaeological area and related to a specific POI on the cultural environment map), the Temple of Neptune, and the Tomb of the Diver fresco (the famous burial

[11] Note that a taxonomy is a particular ontology only containing ISA (concept-subconcept) relationships.

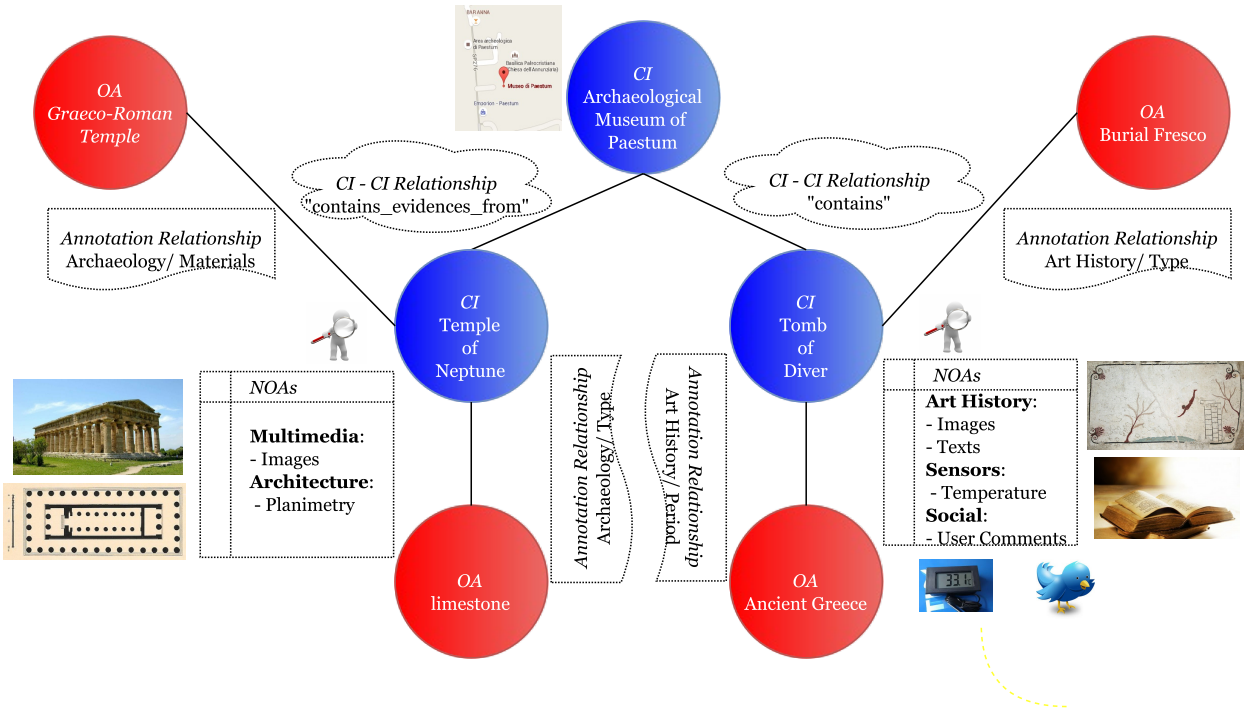

Figure 1: The System Data Model

painting belonging to the Greek period in Paestum) that is located within the museum. The non-ontological attributes of the temple are essentially images related to planimetry (information useful for architects) and some pictures retrieved from social media networks; in turn, the fresco's non-ontological attributes represent information related to Art History (images and text from digital archives), but also comments from on-line social networks and environmental measures collected by the sensors deployed in the museum rooms. Form the other hand, ontological attributes are then referred to different metadata (type and period for the museum artifact, and type and materials of the ancient building) whose values can be linked to the nodes of taxonomies and ontologies available for the Cultural Heritage domain. Leveraging different annotation schemata and available ontologies, our model allows achieving interoperability goals[32].

The Knowledge Base content can be easily exported in the most used formats (e.g., XML, RDF, OWL) and according to the most diffused harvesting standards for CH applications (e.g., EDM, Italian ICCD, etc.). On the other hand, the LD/LOD paradigm permits us to deal with several problems related to data consistency and copyright constraints in an effective manner: some cultural items descriptions are accessible only using URI, thus the data management issues are in charge to the related source.

3. System Description

Our system has to manage a large set of data characterized by significant variety, volume and velocity, representing the well-known big data features.

First of all, we have to deal with the large and heterogeneous amount of information related to the different cultural items, as an example:

- annotations and descriptions provided by cultural heritage foundations' archives or by open encyclopedias;
- multimedia contents – video, text, image and audio – coming from social media networks (e.g., Panoramio, Picasa, YouTube, and Flickr) and digital libraries;
- opinions and comments of users from common on-line social networks like Facebook, Twitter, etc.

In addition, information about users (in terms of needs, preferences and behaviors) and measures (e.g., humidity and temperature) captured by the different sensors deployed in a cultural environment should be also taken into account together with web service data (any kind of useful data gathered by means of web services such as touristic attractions or accommodations in the same geographic area, or meteorological information and so on) in order to provide "smart" services to final users and applications.

The functionalities provided by our system and built on the top of such data are:

- capability to gather information from distributed and heterogeneous data sources (e.g., Social Media Networks, Digital Libraries and Archives, Multimedia Collections, Web Data Services, Sensor Networks, etc.);
- advanced data storage and management techniques and technologies;
- system resources management;
- advanced information retrieval services (ability to provide useful and personalized data to users/applications based on their preferences and needs), browsing facilities, data analytics and other utilities.

By means of a set of ad-hoc APIs, and exploiting the provided services, our system can support several applications: mobile guides for cultural environments, web portals to promote the cultural heritage of a given organization, multimedia recommender and storytelling systems, and so on, also integrated with sensor data, applications for cultural heritage preservation and security.

Thus, our system provides all the necessary retrieval and presentation functionalities to search information of interest and present it to the users/applications in a suitable format and according to their needs.

3.1. Architecture

Figure 2 describes at a glance a functional overview of the proposed system that presents a layered architecture typical of a Big Data platform [28, 29], exploiting the related stack of technologies.

The *resource management layer* coordinates and optimizes the use of the different resources needed for implementing the whole system.

In the *data source layer*, each data source is properly "wrapped" in order to extract the information of interest that is then represented as required by the described data model. In particular, each Wrapper is specialized for a particular kind of source (i.e., Social Media Networks, Digital Repositories, Sensor Networks) and must address all the interoperability issues, providing a set of functionalities to access data sources and gather all the desired data, possibly leveraging the available APIs. Data integration problems for heterogeneous data sources are addressed by means of classical schema mapping techniques, record linkage and data fusion techniques [33] or other type of approaches, according to the specific data source. In addition, data stream management problems have to be considered.

In the *data storage and management layer*, data are stored in the Knowledge Base in compliance with the above-described data model, and managed also exploiting the LD/LOD paradigm. In addition, specific semantics to be attached to the data is provided using the annotation schemata, including ontologies, vocabularies, taxonomies, etc. related to the Cultural Heritage domain. The KB leverages different Data Repositories realized by advanced data management technologies (e.g., Distributed File Systems, NoSQL and relational systems) and provides a set of basic APIs to read/write data by an Access Method Manager.

As a basis for the *data processing layer* our system provides a Query Engine that can be invoked by user applications to search data of interest using information retrieval facilities. In particular, our system supports all the basic functionalities for multimedia and semantic information retrieval by means of proper Information Filters:

- query by keyword or tags and using specific metadata of the annotation schemata,
- query by example for cultural items with images,
- ontology based and query expansion semantic retrieval,
- browsing of a $\mathcal{CI}$ collection.

The *data analytics layer* is based on different Analytics Services allowing to create personalized "dashboards" for a given cultural environment. In addition, it provides basic data mining, graph analysis and machine learning algorithms useful to infer new knowledge .

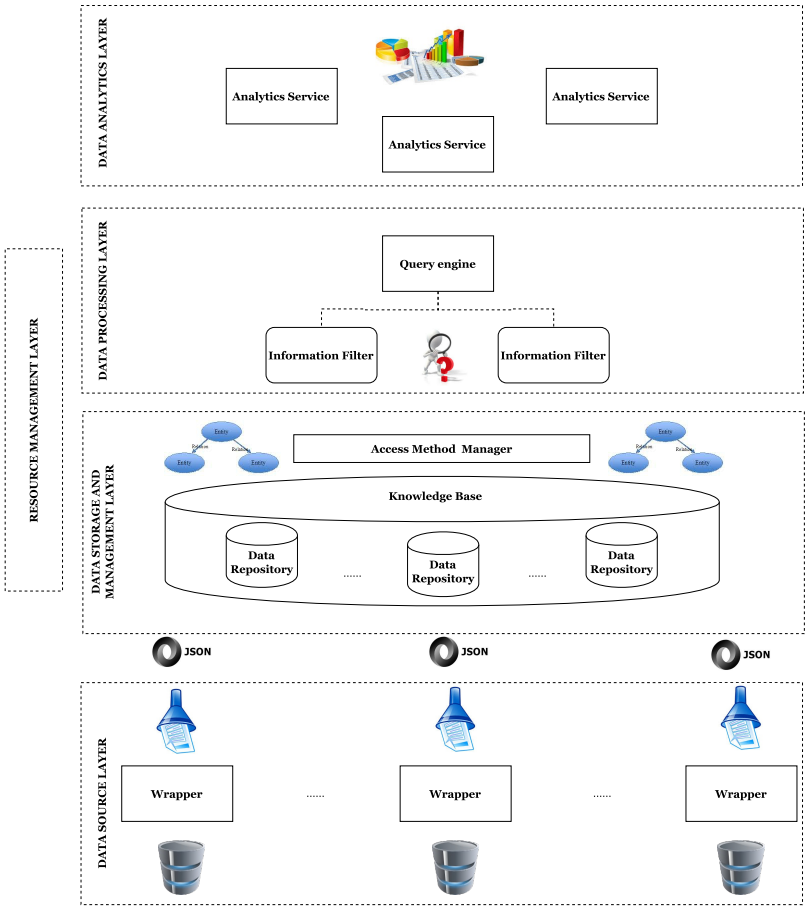

Figure 2: System Architecture.

3.2. Functionalities and Implementation Details

3.2.1. Resource Management

Resource management is an architectural element of paramount importance when designing big data processing solutions on a wide area scale. In such a scenario, typically characterized by strong runtime and storage distribution needs, the right resource management choice can result in significant improvements in both performance and fairness. Indeed, running complex analytics applications on huge amounts of data, continuously collected from hybrid and heterogeneous sources, cannot be effectively accomplished neither on a single machine nor by relying on a centralized data repository.

First, we need to break down processing activities into smaller tasks to be distributed and managed in parallel on multiple machines available on the network. Furthermore, coping with ever increasing data volumes characterized by continuously changing types and formats/structures, requires extremely flexible and elastic storage management policies capable to follow the evolving demand and go beyond the vertical scaling limits of the traditional storage architectures. This implies the adoption of new fully distributed storage paradigms providing horizontal scalability and hence virtually unlimited storage space available.

The *Infrastructure as a Service* (IaaS) facilities provided by modern cloud organizations allow multiple tenants to dynamically obtain or dispose runtime/storage resources available as *virtual machines* (VMs) or containers according to a pay-per-use paradigms. Depending on the cloud service providers' commercial strategies and management policies these object may be provided with fixed or variable amounts of virtual CPUs/cores, available memory and disk space, so that the involved tenants can modify the number of VMs/containers or their available resource pool when their resource demands varies over time.

Our resource management strategy is based on a multi-tenant cloud model, where tenants can buy multiple VMs or containers, characterized by heterogeneous resource demands (resources of different types, such as storage space or memory), to run their applications. The fundamental resource management tasks include discovery, scheduling, allocation and performance monitoring. The discovery activity refers to the identification of a suitable pool of resources that can be used to match the users' demands, whereas the scheduling task aims at selecting the best available options between the matching resources in the above pool, in order to perform provisioning to the requiring runtime objects. This implies allocating runtime or storage resources to specific VMs, Containers and processes, whose usage will be continuously monitored until their final release, after their legitimate use. Several runtime resource management technologies exploiting multiple levels of parallelism in multi-core, many-core, cluster, grid or cloud computing architectures are available, differentiating in their scalability, performance, cost, usage flexibility, robustness etc. [34]. Analogously, many Storage resource management options are available for optimizing the efficiency of shared pools of storage devices distributed throughout the network and support their flexible configuration, together with storage tiering and performance monitoring, by also forecasting future space demands and usage patterns.

In particular, YARN [35] is actually considered one of the most effective and promising resource management solutions for data-intensive processing in distributed cloud-empowered scenarios. Its Hadoop-based architecture provides the foundation enabling big data processing applications to share computing clusters and storage space over the cloud while guaranteeing consistent service levels and acceptable response times. We also selected the *Apache Spark* [36] framework, developed at UC Berkley, in order to support in-memory data sharing across complex, multi-step data pipelines implemented by using a direct acyclic graph abstraction, so that multiple different applications can simultaneously work on the same data sets.

Spark relies on the *Hadoop Distributed File System* (HDFS) in order to provide advanced storage and access functionalities within a unified and flexible big data management solution that is able to cope with a wide range of use cases and requirements. Essentially Spark consists in an execution engine that is able to operate both in-memory and on stable storage devices. It keeps intermediate processing results in memory rather than on semi-permanent storage, that becomes extremely useful mainly when the same data need to be accessed multiple times by different users. However, Spark is able to effectively perform data processing also when their aggregate volume does not fit into the available memory. In these cases it tries to split the data by putting as much as possible in memory and leaving the remaining part on semi permanent storage devices. Due to its in-memory data storage functions and near real-time processing capabilities, it can obtain the best from the *MapReduce* paradigm, limiting the effect of expensive shuffling operations in processing activities, by also achieving significantly improved performances respect to the other existing big data solutions.

Spark provides an abstraction of memorization capabilities based on distributed and fully reliable atomic storage elements known as *Resilient Distributed Datasets* (RDDs) that can be resident in memory or on stable storage, depending on their access patterns. Several operations can be accomplished on these element by generating new RDDs:

- through transformation operators, such as map, *filter* and *reduceByKey*, that define the new elements without immediately processing them,
- as values resulting from action operators like count, collect and save, returning their output to applications or exporting some data sets to stable storage.

For efficiency purposes RDDs can be implemented according to multiple caching/storage options such as, for example, MEMORY ONLY, MEMORY and DISK and DISK ONLY.

In order to support applications runtime facilities Spark creates a master program known as driver that is in charge for RDDs definition and management activities as well as a set of slaves, known as executors, that perform all the computational tasks. In this scenario, RDD objects can be implemented in several different ways:

- by using a file within a distributed file system, such as HDFS;
- by partitioning them into multiple pieces to be delivered on different nodes, through the parallelization of a collection of elements within the driver program;

- by transforming them through an operation such as Map (passing their components through a user-provided function that maps a set of element A into another set B) and filter (selecting all the elements that match a user-defined predicate);

- by modifying the persistence characteristics of an already existing RDD, e.g., materializing a block of ephemeral data to be used in the parallelization of some processing operations, or discarding it when no more needed. In detail, there are two specific operations that can be used to cope with data persistence: "save" that writes the dataset to HDFS after evaluating it, and "cache" that suggest to keep a dataset in memory when possible.

Spark is also able to perform query processing and evaluation on big data, that can be extremely useful in optimizing huge data management workflow, by also providing an high level Application Programming Interface that can introduce significant effects on productivity in applications development. Applications can request distributed processing operations such as map, reduce and filter by passing specific closures (i.e., functions) to the Spark runtime framework. Like in traditional functional programming environments, such closures are typically referred to processing entities and variables that only exist within the scope in which they have been instantiated, so that, when running a closure on a specific node, such entities have to be copied on the node itself.

The overall system architecture is made by using multiple virtual machines rent by a cloud provider, each equipped with Hadoop/YARN and Spark packages, accessing an horizontally scaling dynamic storage space that increases on demand, according to the evolving data memorization needs. It consists in a certain number of slave virtual machines (that can be increased over time in presence of changing demands) and two additional ones that run as masters: one for controlling HDFS and another for resource management.

3.2.2. Data Gathering from Big Data Sources

One of the most important functionalities provided by our system consists in the capability of gathering the different kinds of data from the various Big Data sources.

In particular, we distinguish four classes of data sources that require different wrapping techniques: *Sensor & User Data*, *Social Data*, *Digital Repository Data* and *Web Data*.

Sensor and User Data

The Sensor Data generated by different devices are managed using the "publish-subscribe" paradigm. For such kind of data, we have to deal with data stream management challenges [37]; in addition, event processing techniques can be exploited to reduce the amount of information flow [38].

In particular, the wrapper for such kind of data is composed by a set of Gateways and Message Brokers together with a Coordination and Discovery module. Data generated by sensors are captured by gateways and sent to Knowledge Base by message brokers; from the Knowledge Base side, a Stream Processing Engine receives messages and convert them in a JSON format in according to our data model.

In the current system implementation we deal only with *Wireless Sensor Network* (WSN) data sources. We leverage the *PerLa* WSN management middleware [39] as gateway, which abstracts a sensor network as a virtual distributed database system providing a SQL-like language for query formulation and data retrieval. Data integration issues are coped by a LAV (*Local As View*) schema-mapping approach.

Concerning the other implementation details, we use *Apache Kafka* to realize message brokers and the publish-subscribe communication, *Apache Storm* as stream processing engine (see Figure 3) and *Apache Zookeper* as coordinator.

Measures returned by sensors' stream are finally associated with one or more cultural items in the considered environment[12]. An example of resource that could be derived from such data sources is reported in Figure 4.

The example shows that in a given indoor location (corresponding to a room of the Paestum museum) a certain sensor (identified by the string "S01") has performed a temperature measure (whose value is "32°") in a given temporal instant ("12-Apr-16 14:20") related to a given $C\mathcal{I}$ (with identifier "00ayr4hfdD2")[13]. The meaning of all the metadata

[12] Sensor data can be used both to report environmental conditions (temperature and humidity of indoor or outdoor locations), and especially – for monitoring aims – to support preservation and security of cultural items.

[13] All the information about this measure is accessible via web using the URI www.databenc.it/resource/measure/?id=321

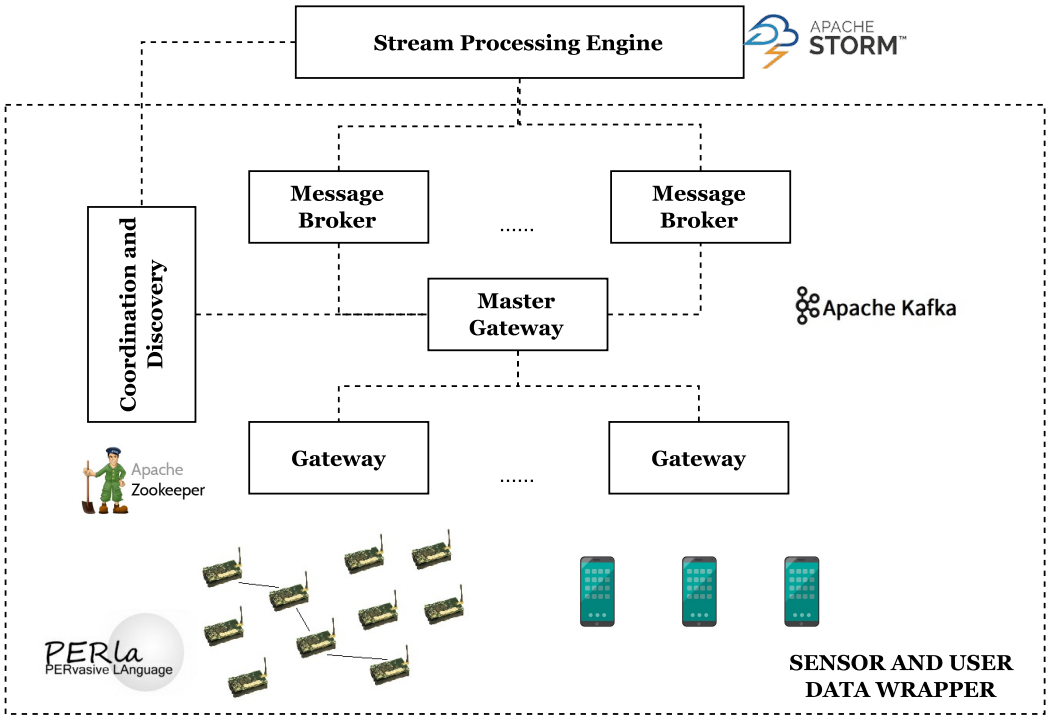

Figure 3: Sensor and User Data Wrapper.

describing the measures (e.g., Type, Timestamp, Location, Value, Source) can be retrieved from the annotation schema *Sensor*.

We note that even the user devices, which can communicate the effective users' location by using the GPS technology or capture their preferences and needs, constitute a different sort of sources, generating User Data that have to be opportunely wrapped and coded in the described format.

For user localization within indoor cultural environments, the applications communicating with our system can use beacons or NFC technologies.

User data may include other static and dynamic information, useful to define the related *profiles*. In particular, recall that user preferences, needs and behaviors can be acquired either explicitly, by means of suitable questionnaires provided by the applications, or in an implicit manner using application logs. Static information (e.g., favorite artistic genre) is stored in the KB and possibly updated; in turn, dynamic information (e.g., behaviors) is continuously updated[14].

Also for this kind of data, we use a publish-subscribe based wrapping: the gateway provides a set of basic API allowing applications to communicate information about position and profiles (see Figure 3).

Social and Digital Repositories Data

As to *Social Data*, the current prototype considers multimedia information coming from Twitter and different social media networks (e.g., Flickr and Panoramio). In particular, the wrapper is constituted by a Social Harvesting module that retrieves user comments and multimedia contents about a given CI by exploiting the related APIs and directly covert them in the described JSON format (see Figure 6). We note that the implementation of the described module is based on the *Social Harvest* framework.The example in Figure 5 shows that user "Vincenzo.Moscato" has expressed a comment ("Wonderful") on a post about *The Tomb of the Diver* in a certain date ("14-Apr-16"). The meaning of all the metadata describing the user activity (e.g., Type, Timestamp, Language, Text, Topic, User, etc.)

[14]Such information constitutes the *Personally Identifiable Information* (PII).

```
Sensor_message={
        “URI”: “www.databenc.it/resource/measure/?id=321”
        “Annotation Schema”: “ Sensor”
        “CI” [
          {
            “id”: “00ayr4hfdD2”
            “URI”: “www.databenc.it/resource/CI/?id=00ayr4hfdD2”
          }
        ]
        “Type”: “ Temperature”
        “Value”: “32°”
        “Timestamp”: “12-Apr-16 14:20 ”
        “Source”: “WSNSensor:S01”
        “Location”: “Indoor/Room”
}
```

Figure 4: An example of JSON message related to sensor data

```
Social_message={
        “URI”: “www.databenc.it/resource/social/?id=76421”
        “Annotation Schema”: “ Social Users”
        “CI” [
          {
            “id”: “00ayr4hfdD2”
            “URI”: “www.databenc.it/resource/CI/?id=00ayr4hfdD2”
          }
        ]
        “Type”: “ Twitter Comment”
        “Language”: “en”
        “Text”: “Wonderful”
        “Timestamp”: “14-Apr-16 10:40 ”
        “Topic”: “The Tomb of the Diver”
        “hashtag”: “#tombofdiver”
        “User”: “foaf:Vincenzo.Moscato”
}
```

Figure 5: An example of JSON message generated by the Twitter Wrapper

can be retrieved from the annotation schema *Social Users*. Social data can be used in applications requiring a "social vision" of cultural items (e.g., general users' mood of a given social community with respect a given picture).

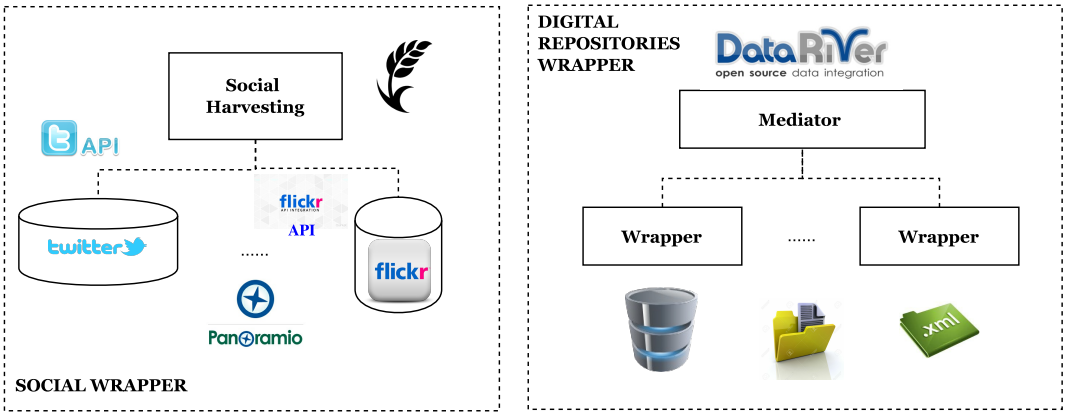

Figure 6: Social and Digital Repositories Wrapper.

We collect *Digital Repositories Data* information describing cultural items from on-line digital repositories (e.g., museums, libraries, theatrical foundations digital archives, multimedia collections, etc.).

Resources are often described on the base of different annotation schemata and provided in XML or RDF based formats (as in *DBpedia*).

The wrapper for this kind of sources can import such data descriptions and convert them into the described JSON format according to the system data model using schema-mapping techniques; in particular we use the *DataRiver* system (see Figure 6).

The example in Figure 7 shows a possible description (together with an image) of the *The Tomb of the Diver*. The meanings of all the metadata describing the painting (e.g., Type, Name, Author, Period, Description, Subject, etc.) can be retrieved in annotation schema *Art History*. Clearly, textual descriptions of cultural items can be made available in different languages and users' applications can choose the desired ones.

All the multimedia web resources (e.g., images, texts, video, etc.) related to a given cultural item can be similarly collected using the wrapper facilities, and associated with a given annotation schema.

In particular, the descriptions in terms of basic metadata are captured and stored within our systems, while raw multimedia data can be opportunely linked and, in other cases, temporarily imported into the system for content-based analysis [40, 41][15].

Web Data

Finally, other information of interest, such as that related to meteorological or traffic conditions, can also be captured using the available *Web Data* services and represented in the described format.

3.2.3. Data Storage and Management

The data gathered by the Wrappers are stored and managed by the Knowledge Base. Data are represented according to the described model, based on the LD/LOD paradigm. One of the basic functionalities of the KB is to export the related content into the *Europeana Data Model*[16] (EDM) format, a proposal that aims at bridging the gaps between different harvesting standards providing a unique solution to metadata specification.

In the EDM [30], each *Cultural Heritage Object* (CHO) - the main entity of the model corresponding to a given CI - can be described at various levels of detail using different metadata schemata by means of the *aggregation*

[15]Note that multimedia data that are managed by the system are suitably filtered before the storing process. The number and kinds of multimedia data required by the application are tuned by means of configuration parameters.

[16]http://pro.europeana.eu/edm-documentation

```
Repository_message={
        “URI”: “www.databenc.it/resource/datarepository/?id=7502”
        “Annotation Schema”: “ Art History”
        “CI” [
          {
            “id”: “00ayr4hfdD2”
            “URI”: “www.databenc.it/resource/CI/?id=00ayr4hfdD2”
          }
        ]
        “Multimedia” [
          {
            “Type”: “external/image”
            “URI”: “https://it.wikipedia.org/wiki/.../PaestumTaucher.jpg”
          }
        ]
        “Type”: “ Burial Fresco”
        “Name”: “The Tomb of the Diver”
        “Period”: “Ancient Greek”
        “Description” [
          {
            “Text”: “The painted decoration of the tomb called the diver,
            found in 1968, shows a great moment of Greek painting,
            around 480 BC, characterized by the same spirit of the
            painters of the vascular severe style.”
            “Language”: “en”
          }
        ]
        “Subject”: “Real Life activity”
        “Actual Location”: “Archeological Museum of Paestum”
}
```


Figure 7: An example of JSON message generated by the Digital Repository Wrapper

construct. Thus, our annotation schemata correspond to specific metadata aggregations: as an example, social and sensor aggregations permit to associate/link interesting environmental measures or users' comments from an Online Social Network to each cultural item. Similarly, the archeological aggregation supports the description of a cultural item by metadata that are useful for the annotation work of an archeologist.

Two further EDM basic concepts are considered and mapped in our model: *Place* (where the object is located corresponding to our $\mathcal{PoI}$ entity) and *Time Span* (allowing to deal with different temporal views of a cultural item).

Possibly, a certain number of web resources (e.g., images, videos, texts, etc.) can be linked to each item. Finally, metadata semantics is provided by the set of annotation schemata (in XML, RDF or OWL formats). For instance, the data can be represented as sequences of triples ($\langle$ subject, predicate, object $\rangle$) according to the described data model.

In particular, the KB is based on several technologies (see Figure 8):

- The data describing basic properties of $\mathcal{CI}$s (e.g., name, short description, etc.) and basic information on users profiles are stored into a *key-value data store* (i.e., *Redis*).
- The complete description in terms of all the metadata of $\mathcal{CI}$s using the different annotation schemata are in turn saved using a *wide column data store* (i.e., *Cassandra*). We use a table for each kind of $\mathcal{CI}$s having a column for each "metadata family"; column values can be literals or URIs.
- The *document store* technology (i.e., *MongoDB*) is used to deal with JSON messages, complete user profiles and descriptions of internal resources (multimedia data and textual documents, etc.) associated with a cultural item.
- All the relationships among cultural items within a cultural environment and interactions with users (behaviors) are managed by means of a *graph database* (i.e., *Titan*).
- The entire cartography related to a cultural environment together with $\mathcal{PoI}$s is managed by a GIS (i.e., *PostGIS*), which provides the functionalities to filter and visualize on a map the geographic area around a given $\mathcal{PoI}$;
- Multimedia data management have been realized using the *Windsurf* library [41];
- We exploit an *RDF store* (i.e., different *Allegrograph* instances) to memorize data views in terms of triples related to a given cultural environment and useful for specific applications, providing a SPARQL endpoint for the applications;
- All system configuration parameters, internal catalogs and thesauri are stored in a relational database (i.e., *PostegreSQL*)
- All the basic analytics and query facilities are provided by *Spark* based on *HDFS*;
- Semantics of data can be specified by linking values of high-level metadata to some available internal (managed by Sesame) or external ontological schemata.

This heterogeneous Knowledge Base provides basic *Restful APIs* to read/write data and further functionalities for importing/exporting data in the most common diffused Web standards.

We chose to adopt such heterogeneous technologies in order to meet the specific requirements of the applications dealing with the huge amount of data at stake. For example, Social Networking applications typically benefit of graph database technologies because of their focus on data relationships. In turn, more efficient technologies (key-value ore wide-column stores) are required by Tourism applications to quickly and easily access the data of interest.

3.2.4. Data Processing and Analytics

The search of data useful for the applications can be eased by using different Information Filters that implement the right queries to the various databases.

The implementation of such filters is based on *Spark*, and we distinguish three four kinds of query on the KB:

- *query by keywords/tags*: through such a query a user/application can search a set of $\mathcal{CI}$s using keywords (as in Google search engine) or specific tags (the query is then "expanded" with similar search terms leveraging the system thesauri);

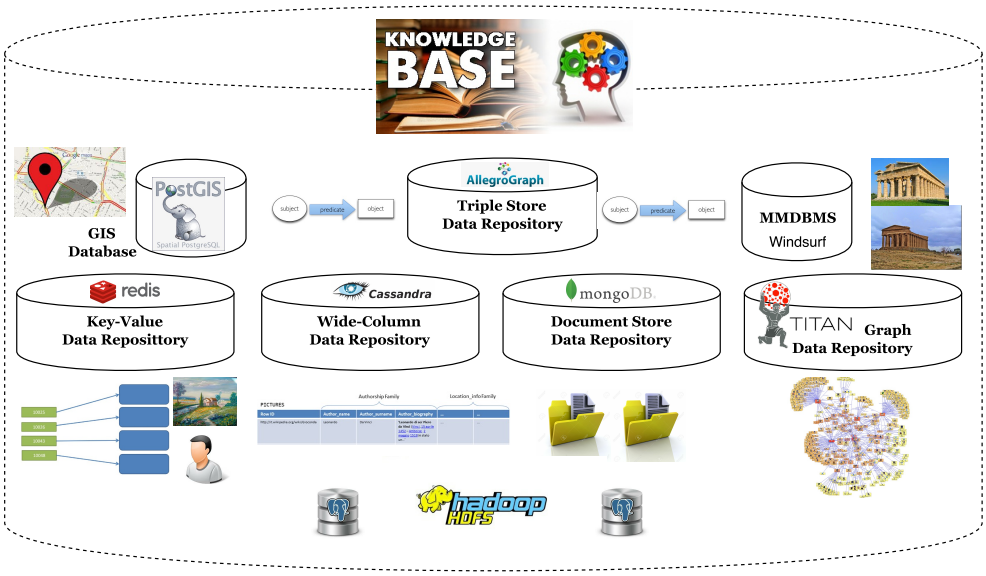

Figure 8: Knowledge Base.

- *query by metadata*: by this query a user/application can search a set of $C\mathcal{I}$s using specific metadata of internal or external annotation schemata (the query for OAs is semantically "expanded" with the concepts of managed ontologies that are similar to the target one);
- *query by example*: through such a query a user/application can search a set of multimedia contents – related to $C\mathcal{I}$s – that are similar to a target one (the query processing is base on the *Windsurf multimedia libraries*);
- *query by user preferences*: user profiles are exploited to find the set of cultural items that are more similar to user preferences using co-clustering techniques [42].

The different types of query are combined by the Query Engine that provides a system query endpoint and support different query languages.

Eventually, several analytics can be performed on the stored data using *Graph Analysis* and *Machine Learning* library provided by Spark. As an example, the analysis of the graph coding interactions among users, multimedia objects and cultural items can be properly used to support recommendation or influence analysis tasks[43].

4. System Running Example

In this section we describe a possible application of our system to support the development of a multimedia guide for the *Paestum archeological site*.

The archaeological site of Paestum is one of the major Graeco-Roman cities in Southern Italy. Here, a set of ancient buildings is the main cultural attraction for a tourist: three main temples of Doric style (the *First Temple of Hera*, also called *Basilica*, the *Second Temple of Hera*, also known as *Temple of Neptune*, and the *Temple of Athena*), the *Roman Forum* with several ruins, and the *Amphitheater*. The buildings are surrounded by the remains of the city walls. In addition, a museum near the ancient city contains many evidences of the Graeco-Roman life (e.g., amphorae, paintings and so on).

The ancient buildings, together with the museum and its main artifacts, constitute the set of cultural items for our case study. Tourists, both from their places and while visiting ruins, can browse these cultural items and enjoy a useful multimedia guide describing them, or be recommended other nearby places, comments of other users and other information of interest.

As an example, suppose that some users from the nearby seaside resorts would like to know more about the ruins in front of them, beyond their names and a very generic and basic explanation of the main cultural items.

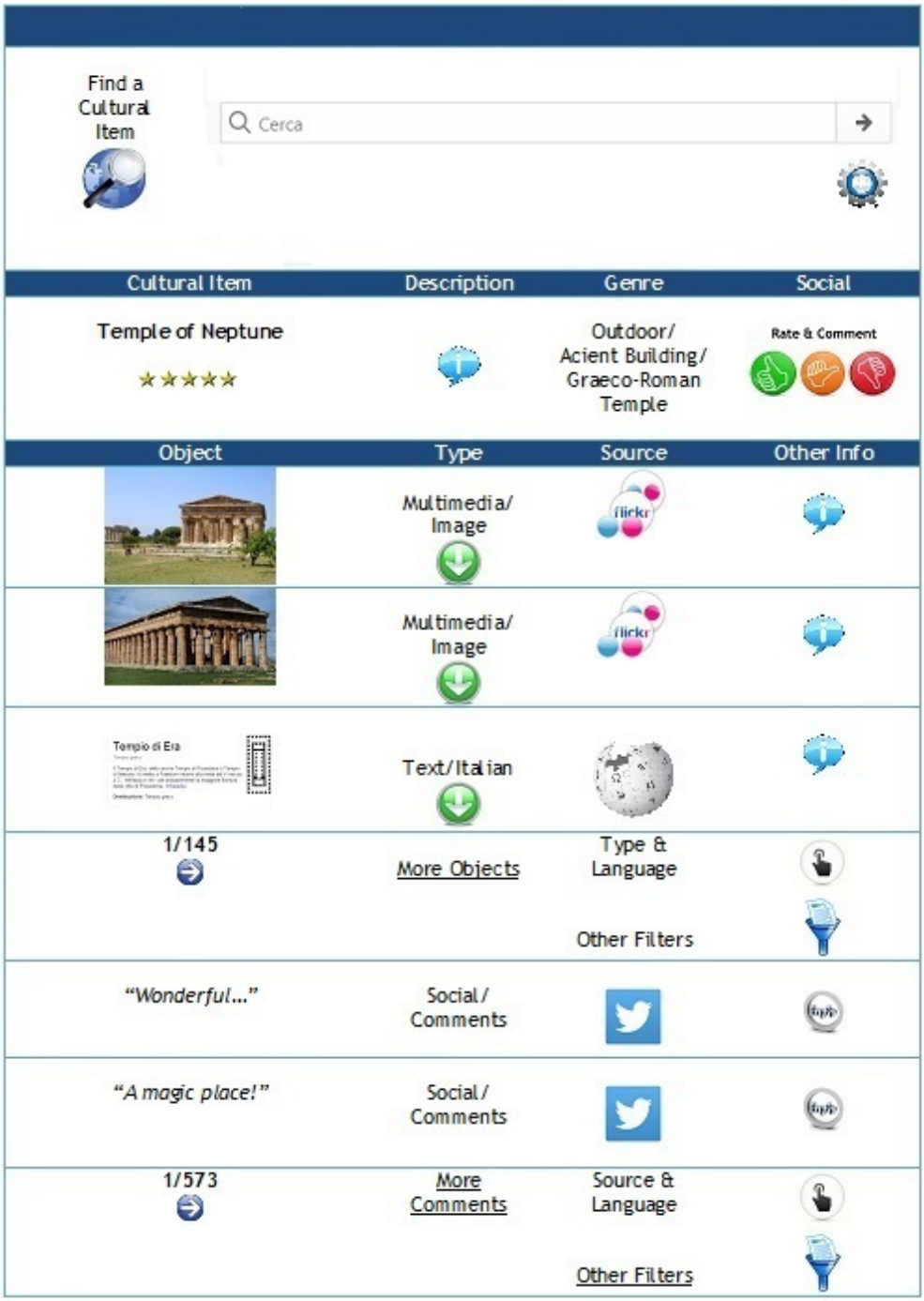

Figure 9: System Query Interface

When users search a specific cultural item, as an example the Temple of Neptune, our system provide a basic description with the related multimedia objects (i.e., audio, images, video and texts) and detailed users' comments. The list of proposed cultural items with descriptions and multimedia objects depends on the user's preferences and system settings: images rather than voice, or expert-level rather than layman-level descriptions of the art pieces; specific metadata and annotation schemata. In addition, *query by example* facilities can be exploited to determine other images that are similar to a given multimedia object. In addition, a *semantic based search* can be performed on specific ontological attributes to find other cultural items of the same type. At the same time, users can choose to retrieve some interesting information, to read comments, opinions and ratings about the visited cultural items and to express their own ratings and opinions that will be posted on social networks.

Figure 9 shows a running example (obtained by assembling different screenshots of our system) concerning the search of $C\mathcal{I}$s related to the Paestum ruins. Users can browse the data by means of an appropriate GUI; they can filter objects belonging to a given $C\mathcal{I}$ using different criteria: type of multimedia data, language, size, and so on.

Thus, the information can be dynamically tailored on the users depending on their preferences and needs. As an example, young people with high-school education and basic notions of history and art could receive only some general concepts about the cultural items, while CH experts could be interested in more detailed descriptions and richer multimedia data.

Eventually, other cultural items of interest can be automatically suggested using recommendation techniques that exploit sontext information, user preferences and needs and items' features [42].

All the information and functionalities (provided as API) can be exploited by third-part applications such as a multimedia guide for the Paestum ruins.

The system prototype is currently running on a cloud-based big data processing environment based on Spark . In particular, we exploited 5 computing nodes, each one composed by 8 cores and 15 GB RAM.

5. Conclusions and Future Work

In this paper we showed an application and some examples of the use of Big Data technologies for the Cultural Heritage domain.

In particular, we described CHIS - a scalable prototype for the management and context-driven browsing of cultural environments . The system is characterized by several features that are typical of the modern big data systems: i) information gathering from distributed and heterogeneous pervasive data sources (e.g., Sensor Networks, Social Media Networks, Digital Libraries and Archives, Multimedia Collections, Web Data Services, etc.); ii) context-awareness, provisioning of useful and personalized data and services, on the base of users preferences and of the surrounding environment; iii) advanced data management techniques and technologies for dealing with the information variety, velocity and volume; iv) advanced information retrieval facilities, data analytics and other utilities/services.

We detailed all the design choices and characteristics of the implemented big data infrastructure, providing a system running example concerning the cultural data management of the archaeological site of Paestum.

Future work will be devoted to collect the huge amount of data related to all the different cultural objects of the Campania region and to experiment our system from the efficiency and effectiveness points of view with respect to the information retrieval and filtering tasks providing a comparison with other systems.

References

[1] A. Caragliu, C. Del Bo, P. Nijkamp, Smart cities in europe, Journal of urban technology 18 (2) (2011) 65–82.

[2] H. Schaffers, N. Komninos, M. Pallot, B. Trousse, M. Nilsson, A. Oliveira, Smart cities and the future internet: Towards cooperation frameworks for open innovation, in: The Future Internet Assembly, Springer, 2011, pp. 431–446.

[3] J. M. Hernández-Muñoz, J. B. Vercher, L. Muñoz, J. A. Galache, M. Presser, L. A. H. Gómez, J. Pettersson, Smart cities at the forefront of the future internet, in: The Future Internet Assembly, Springer, 2011, pp. 447–462.

[4] N. Komninos, H. Schaffers, M. Pallot, Developing a policy roadmap for smart cities and the future internet, in: eChallenges e-2011 Conference Proceedings, IIMC International Information Management Corporation, IMC International Information Management Corporation, 2011.

[5] M. Armbrust, A. Fox, R. Griffith, A. D. Joseph, R. Katz, A. Konwinski, G. Lee, D. Patterson, A. Rabkin, I. Stoica, et al., A view of cloud computing, Communications of the ACM 53 (4) (2010) 50–58.

[6] J. A. Burke, D. Estrin, M. Hansen, A. Parker, N. Ramanathan, S. Reddy, M. B. Srivastava, Participatory sensing, Center for Embedded Network Sensing.

[7] C. Bizer, T. Heath, T. Berners-Lee, Linked data-the story so far, Semantic Services, Interoperability and Web Applications: Emerging Concepts (2009) 205–227.
[8] T. Berners-Lee, J. Hendler, O. Lassila, et al., The semantic web, Scientific american 284 (5) (2001) 28–37.
[9] S. Lohr, The age of big data, New York Times 11.
[10] C. Bizer, P. Boncz, M. L. Brodie, O. Erling, The meaningful use of big data: four perspectives–four challenges, ACM SIGMOD Record 40 (4) (2012) 56–60.
[11] F. Colace, M. D. Santo, V. Moscato, A. Picariello, F. A. Schreiber, L. Tanca, Patch: A portable context-aware atlas for browsing cultural heritage, in: Data Management in Pervasive Systems, 2015, pp. 345–361.
[12] L. Tan, N. Wang, Future internet: The internet of things, in: 2010 3rd International Conference on Advanced Computer Theory and Engineering (ICACTE), Vol. 5, IEEE, 2010, pp. V5–376.
[13] L. Atzori, A. Iera, G. Morabito, The internet of things: A survey, Computer networks 54 (15) (2010) 2787–2805.
[14] J. Zhang, V. Varadharajan, Wireless sensor network key management survey and taxonomy, Journal of Network and Computer Applications 33 (2) (2010) 63–75.
[15] J. J. Rodrigues, P. A. Neves, A survey on ip-based wireless sensor network solutions, International Journal of Communication Systems 23 (8) (2010) 963–981.
[16] N. Mohamed, J. Al-Jaroodi, A survey on service-oriented middleware for wireless sensor networks, Service Oriented Computing and Applications 5 (2) (2011) 71–85.
[17] H. Alemdar, C. Ersoy, Wireless sensor networks for healthcare: A survey, Computer Networks 54 (15) (2010) 2688–2710.
[18] S. LaValle, E. Lesser, R. Shockley, M. S. Hopkins, N. Kruschwitz, Big data, analytics and the path from insights to value, MIT sloan management review 52 (2) (2011) 21.
[19] P. Russom, et al., Big data analytics, TDWI Best Practices Report, Fourth Quarter (2011) 1–35.
[20] M. Damova, A. Ontotext, B. A. Kiryakov, M. Grinberg, M. K. Bergman, F. Giasson, K. Simov, Creation and integration of reference ontologies for efficient lod management, Semi-Automatic Ontology Development: Processes and Resources: Processes and Resources (2012) 162.
[21] R. Ruggles, Knowledge management tools, Routledge, 2009.
[22] H. M. Khoury, V. R. Kamat, Evaluation of position tracking technologies for user localization in indoor construction environments, Automation in Construction 18 (4) (2009) 444–457.
[23] L. M. Ni, D. Zhang, M. R. Souryal, Rfid-based localization and tracking technologies, IEEE Wireless Communications 18 (2) (2011) 45–51.
[24] K. Kabassi, Personalisation systems for cultural tourism, in: Multimedia services in intelligent environments, Springer, 2013, pp. 101–111.
[25] F. Ricci, L. Rokach, B. Shapira, Introduction to recommender systems handbook, Springer, 2011.
[26] C. Becker, F. Dürr, On location models for ubiquitous computing, Personal and Ubiquitous Computing 9 (1) (2005) 20–31.
[27] J. Scott, Social network analysis, Sage, 2012.
[28] C. P. Chen, C.-Y. Zhang, Data-intensive applications, challenges, techniques and technologies: A survey on big data, Information Sciences 275 (2014) 314–347.
[29] F. Colace, M. De Santo, V. Moscato, A. Picariello, F. A. Schreiber, L. Tanca, Pervasive systems architecture and the main related technologies, in: Data Management in Pervasive Systems, Springer, 2015, pp. 19–42.
[30] M. Doerr, S. Gradmann, S. Hennicke, A. Isaac, C. Meghini, H. van de Sompel, The europeana data model (edm), in: World Library and Information Congress: 76th IFLA general conference and assembly, 2010, pp. 10–15.
[31] F. Amato, A. Mazzeo, A. Penta, A. Picariello, Building rdf ontologies from semi-structured legal documents, in: 2008 International Conference on Complex, Intelligent and Software Intensive Systems, 2008.
[32] F. Amato, A. R. Fasolino, A. Mazzeo, V. Moscato, A. Picariello, S. Romano, P. Tramontana, Ensuring semantic interoperability for e-health applications, in: Complex, Intelligent and Software Intensive Systems (CISIS), 2011 International Conference on, IEEE, 2011, pp. 315–320.
[33] X. L. Dong, D. Srivastava, Big data integration, in: Data Engineering (ICDE), 2013 IEEE 29th International Conference on, IEEE, 2013, pp. 1245–1248.
[34] J. L. Reyes-Ortiz, L. Oneto, D. Anguita, Big data analytics in the cloud: Spark on hadoop vs mpi/openmp on beowulf, Procedia Computer Science 53 (2015) 121–130.
[35] V. K. Vavilapalli, A. C. Murthy, C. Douglas, S. Agarwal, M. Konar, R. Evans, T. Graves, J. Lowe, H. Shah, S. Seth, et al., Apache hadoop yarn: Yet another resource negotiator, in: Proceedings of the 4th annual Symposium on Cloud Computing, ACM, 2013, p. 5.
[36] M. Zaharia, M. Chowdhury, M. J. Franklin, S. Shenker, I. Stoica, Spark: cluster computing with working sets., HotCloud 10 (2010) 10–10.
[37] E. Panigati, F. A. Schreiber, C. Zaniolo, Data streams and data stream management systems and languages, in: Data Management in Pervasive Systems, Springer, 2015, pp. 93–111.
[38] G. Cugola, A. Margara, The complex event processing paradigm, in: Data Management in Pervasive Systems, Springer, 2015, pp. 113–133.
[39] F. A. Schreiber, R. Camplani, M. Fortunato, M. Marelli, G. Rota, Perla: A language and middleware architecture for data management and integration in pervasive information systems, Software Engineering, IEEE Transactions on 38 (2) (2012) 478–496.
[40] F. Amato, F. Colace, L. Greco, V. Moscato, A. Picariello, Semantic processing of multimedia data for e-government applications, Journal of Visual Languages & Computing 32 (2016) 35–41.
[41] I. Bartolini, M. Patella, Multimedia queries in digital libraries, in: Data Management in Pervasive Systems, Springer, 2015, pp. 311–325.
[42] F. Colace, M. D. Santo, L. Greco, V. Moscato, A. Picariello, A collaborative user-centered framework for recommending items in online social networks, Computers in Human Behavior 51 (2015) 694–704.
[43] F. Amato, V. Moscato, A. Picariello, G. Sperlì, Multimedia social network modeling: A proposal, in: Tenth IEEE International Conference on Semantic Computing, ICSC 2016, Laguna Hills, CA, USA, February 4-6, 2016, 2016, pp. 448–453.