Building Earth Observatories using Scientific Database and Semantic Web Technologies

Manolis Koubarakis
National and Kapodistrian University of Athens, Greece
koubarak@di.uoa.gr

Stefan Manegold
Centrum Wiskunde & Informatica, Amsterdam, NLD
Stefan.Manegold@cwi.nl

Charalambos Kontoes
National Observatory of Athens, Greece
kontoes@noa.gr

ABSTRACT

Advances in remote sensing technologies have allowed us to send an ever-increasing number of satellites in orbit around Earth. As a result, Earth Observation data archives have been constantly increasing in size in the last few years (now reaching petabyte sizes), and have become a valuable source of information for many scientific and application domains (environment, oceanography, geology, archaeology, security, etc.). TELEIOS is a recent European project that addresses the need for scalable access to petabytes of Earth Observation data and the discovery of knowledge that can be used in applications. To achieve this, TELEIOS builds on scientific database technologies (array databases, SciQL, data vaults) and Semantic Web technologies (stRDF and stSPARQL) implemented on top of a state-of-the-art column-store database system (MonetDB). In this paper we outline the vision of TELEIOS (now in its second year), present its software architecture and give a detailed example of a fire monitoring application that we have completed.

Categories and Subject Descriptors

H.4 [**Information Systems Applications**]: Miscellaneous; D.2.8 [**Software Engineering**]: Metrics—*complexity measures, performance measures*

General Terms

????

Keywords

?????

1. INTRODUCTION

Advances in remote sensing technologies have enabled public and commercial organizations to send an ever-increasing number of satellites in orbit around Earth. As a result, Earth Observation (EO) data has been constantly increasing in volume in the last few years, and it is currently reaching petabytes in many satellite archives. For example, the multi-mission data archive of the TELEIOS partner German Aerospace Center (DLR) is expected to reach 2 PB next year, while ESA estimates that it will be archiving 20 TB of data before the year 2020. As the volume of data in satellite archives has been increasing, so have the scientific and commercial applications of EO data. Nevertheless, it is estimated that up to 95% of the data present in existing archives has never been accessed, so the potential for increasing exploitation is very big.

TELEIOS[1] is a recent European project that addresses the need for scalable access to PBs of Earth Observation data and the effective discovery of knowledge hidden in them. TELEIOS started on September 2010 and it will last for 3 years. In the first one and a half years of the project, we have made significant progress in the development of state-of-the-art techniques in Scientific Databases, Semantic Web and Image Mining and have applied them to the management of EO data.

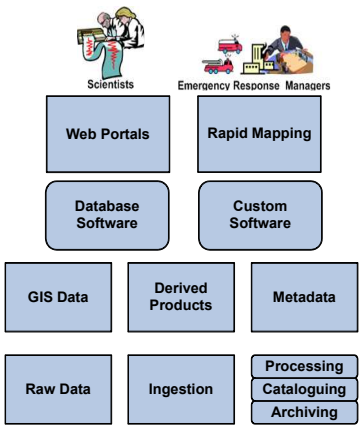
The techniques developed in TELEIOS are currently being implemented on top of the pioneer column-store database system MonetDB[2] which has many of the capabilities we need for scalable querying of PBs of satellite image data with SciQL, and billions of stRDF triples with stSPARQL. We have already demonstrated the scalability of our stSPARQL implementation to billions of stRDF triples with our system Strabon[3], originally developed on top of PostgreSQL [5]. Strabon has been ported to MonetDB with the aim of taking advantage of column-store functionalities for representating and querying geospatial data. Similarly, work is currently underway on implementing SciQL on top of MonetDB by extending its well-known SQL components [17].

The contributions of this paper are the following:

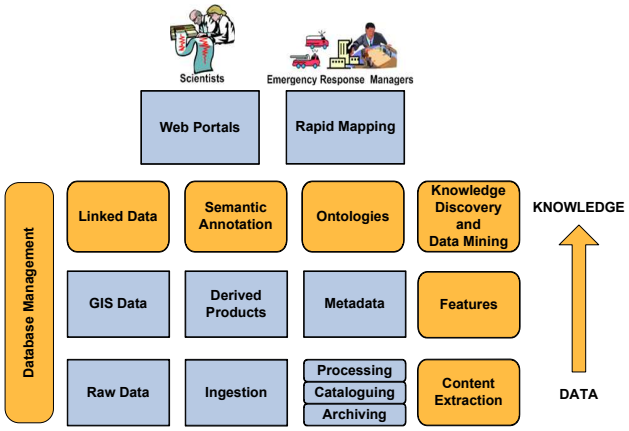
- We outline the vision of TELEIOS and explain in detail why it goes beyond operational systems currently deployed in various EO data centers.
- Because data models and languages play an important role in TELEIOS, we discuss in some detail the ones we utilize: the scientific database query language SciQL, and the data model stRDF with its query language stSPARQL targeted at geospatial data expressed in RDF.

EDBT/ICDT 2013 Joint Conference March 18-22, 2013 - Genoa, Italy

[1] http://www.earthobservatory.eu/
[2] http://www.monetdb.org/
[3] http://www.strabon.di.uoa.gr/



(a) Concept view of a state-of-the-art EO data center



(b) Concept view of the TELEIOS Earth Observatory

Figure 1: Pre-TELEIOS EO data centers and the TELEIOS Earth Observatory

- We give a detailed example of a fire monitoring application that we have just completed using TELEIOS technologies for one of the TELEIOS partners, the National Observatory of Athens (NOA).

The rest of the paper is organized as follows. ... Section 2 presents the fire monitoring service operational at NOA before TELEIOS, and Section 3 explains in detail how this service was improved using our technologies. Last, Section 4 discusses related work and Section 5 concludes the paper.

2. THE FIRE MONITORING APPLICATION OF NOA

Fire monitoring and management in Europe, and in the wider Mediterranean region in particular, is of paramount importance. Almost every summer massive forest wildfires break out in several areas across the Mediterranean, leaving behind severe destruction in forested and agricultural land, infrastructure and private property, and losses of human lives.

European initiatives in the area of EO like GMES (Global Monitoring for Environment and Security)[4] have therefore undertaken an active role in the area of fire monitoring and management in Europe, and supported the development of relevant European operational infrastructures through projects such as linkER (Supporting the implementation of an operational GMES service in the field of emergency management) and SAFER (Services and Applications For Emergency Response)[5].

In the framework of SAFER, NOA has been archiving and processing on a routine basis, large volumes of satellite images of different spectral and spatial resolutions (low, middle, and high spatial resolution) in combination with auxiliary geo-information layers (land use/land cover data, administrative boundaries, and roads and infrastructure networks data) to generate, validate and deliver fire-related products and services to the entire Southern Europe (Spain, France, Italy, Portugal, and Greece).

[4]http://gmes.info/
[5]http://www.emergencyresponse.eu/

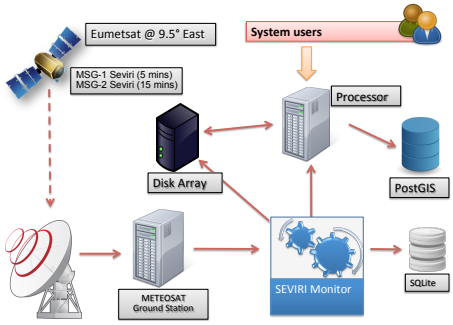


Figure 2: The NOA fire monitoring service

In this context NOA has been developing a real-time fire hotspot detection service for effectively monitoring a fire-front. The technique is based on the use of acquisitions originating from the SEVIRI (Spinning Enhanced Visible and Infrared Imager) sensor, on top of MSG-1 (Meteosat Second Generation satellite, renamed to Meteosat-8) and MSG-2 (renamed to Meteosat-9) satellite platforms. Since 2007, NOA operates an MSG/SEVIRI acquisition station, and has been systematically archiving raw satellite images on a 5 and 15 minutes basis, the respective temporal resolutions of MSG-1 and MSG-2. The archives of raw imagery are now in the order of 2 Terabytes, corresponding to the summer fire periods of the last five years.

The fire monitoring service active in NOA *before TELEIOS* is presented graphically in Figure 2 and can be summarized as follows:

(1) The ground-based receiving antenna collects all spectral bands from MSG-1 and MSG-2 every 5 and 15 minutes respectively.

(2) The raw datasets are decoded and temporarily stored in the METEOSAT Ground Station as wavelet compressed images.

(3) The application SEVIRI Monitor, written in Python, manages the data stream in real-time by offering the following functionality:

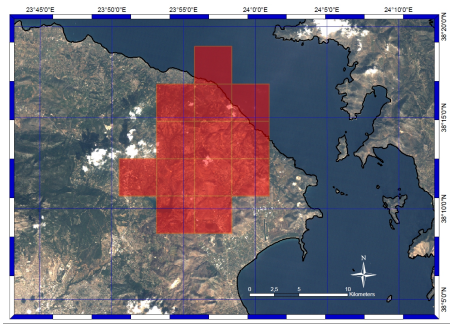


Figure 3: A detailed vector representation of fires at Attica, Greece, in 2010

1. Extract and store the raw file metadata in an SQLite database. This metadata describes the type of sensor, the acquisition time, the spectral bands captured, and other related parameters. Such a step is required as one image is comprised of multiple raw files which might arrive out-of-order.

2. Filter the raw data files, disregarding non-applicable data for the fire monitoring scenario, and dispatch them to a dedicated disk array for permanent storage.

3. Remotely trigger the processing chain by transferring the appropriate spectral bands via FTP to a dedicated machine and initiating the distinct processing steps described in [15]. These steps are: (i) cropping the image to keep only the area of interest, (ii) georeferencing to the geodetic reference system used in Greece (HGRS 87), (iii) classifying the image pixels as "fire" or "non-fire" using the algorithm of [4], and finally (iv) exporting the final product to raster and vector formats (ESRI shapefiles).

4. Dispatch the derived products to the disk array and additionally store them to a PostGIS database system.

The products that are stored in PostGIS cover the geographical area of Greece and are disseminated to the end user community (civil protection agencies, regional authorities, and decision makers) through a web application that uses the interoperable tool GeoServer[6] for sharing geospatial data.

The fire pixels derived by the above processing chain have dimensions equal to the sensor's spatial resolution, in this case nearly 4x4 km. Thus, MSG/SEVIRI is a low resolution observational system, compared to other very high resolution sensors with similar fire detection capabilities (e.g., WorldView-2 at 0.5 m, Quickbird at 2.4 m, IKONOS at 4 m or Formosat-2 at 8 m), high resolution sensors (e.g., Spot-5 at 10 m and Landsat-5 TM at 30 m), or medium resolution sensors (e.g., MODIS Terra and Aqua with 2 bands at 250 m, 5 bands at 500 m and 29 bands at 1 km). However, the unique advantage of MSG/SEVIRI is its geostationary orbit, which allows for a very high observational frequency (5-15 minutes) over the same area of interest. Other satellite platforms with better spatial resolution are forced to undertake orbits that are closer to the earth, which considerably reduces their revisit time. For example, Aqua MODIS, with its near-polar orbit, passes over Greece twice a day (at 00:30 and 11:30) and the same applies for Terra MODIS (at 9:30 and 20:30). Another important advantage of the MSG/SEVIRI sensor is that its sensitivity is not at all affected by its low spatial resolution, i.e., it is not necessary for an entire 4x4 km pixel to be "on fire" to detect the corresponding hotspot. A small pixel portion, exhibiting increased temperature due to a wildfire, will suffice. In conclusion, the increased five minute temporal resolution of MSG/SEVIRI is an exceptional capability that allows civil protection operators to have an almost real-time overview of the situation in terms of forest wildfires. A typical example that highlights the usefulness of the hotspot products in Greece is shown in Figure 3. Additionally, another comparative advantage of MSG/SEVIRI with respect to higher spatial resolution sensors, is the increased field of view, i.e., its footprint on the Earth. While, for high and very high resolution sensors, this is limited to 10-200 km, MSG/SEVIRI covers with a single image most of Europe and Africa, allowing for applications with a global coverage to be developed.

One of the goals of TELEIOS is to improve the hotspot detection and fire monitoring service of NOA described above. The main issues that need to be addressed are the following:

- The thematic accuracy of the generated products has to be refined in a clear and systematic way, to ensure the reliability and transferability of the service to other geographic areas. The main problem with the current thematic accuracy is the existence of false alarms and omission errors in the fire detection technique that relate to the following scenarios:
  - Cases of hotspots occurring in the sea or in locations represented by fully inconsistent land use/land cover classes, like urban or permanent agriculture areas. If these hotspots correspond to real fires, these fires occur in the vicinity of coasts or urban areas, but due to the low spatial pixel resolution of the MSG/SEVIRI instrument and errors in image geo-referencing, the hotspots wrongly appear to be over inconsistent underlying land use/land cover classes. This type of error could be easily corrected if derived hotspot products are compared with auxiliary GIS layers by a NOA operator. However, this would certainly require time for manual GIS layer integration and visual interpretation, an operation that is not possible in the available 5 minute time frame.
  - Cases of hotspots located outside forested areas. These can be false fire detections due to known problems with existing hotspot detection algorithms (e.g., inappropriate fire/no-fire thresholds in the algorithm of [4]). They can also be real cases of fires located in big agricultural plains that are put by farmers as part of their agricultural practices. Whichever the case, they are not real forest fires, and they are not emergency situations to be handled. This type of noisy information could be avoided if derived hotspot products are combined together with land use/land cover information, again an operation that cannot be done manually in the 5 minute time frame.
  - Spatial and temporal inconsistencies in the final product. Today hotspot detection at a given time

[6] `http://geoserver.org/`

is done by using a single image acquisition corresponding to that time, without taking into consideration hotspots and their locations in previous image acquisitions, e.g., hotspots detected during the last 1 to 2 hours. Given the inaccuracies of existing hotspot detection algorithms [4], this single-scene processing approach results in some spatial and temporal inconsistencies between the different observations. A simple heuristic, which would result in significant noise removal, is to check the number of times a specific fire was detected over the same or near the same geographic location during the last hour(s), considering the observation's temporal and spatial persistence, and hence attributing a level of confidence to each detected pixel.

- The need to generate added-value thematic maps combining diverse information sources. As a service provider NOA aims at delivering to the end-user community reliable and comprehensive information for fire related emergency situations. Although vector shapefiles are useful for analysis in the aftermath of a crisis, in real-time emergency response scenarios, civil protection agencies and local firefighting teams find it more useful to refer to a map depicting the active fire-front and its evolution in the last hours/days and identify nearby crucial infrastructure (hospitals, schools, industrial sites, fire hydrants, etc.). This is of paramount importance for the effective allocation of resources during the crisis. Therefore, a desired functionality that is currently missing is automatic map generation enriched with easily accessible geo-information layers.

- Dispersion of the various processes of the fire monitoring service in many machines and pieces of software makes it difficult for NOA to keep all functionalities synchronized. There is no consistent management policy, but various independent components (as seen in Figure 2) that are glued together with the Python-based application SEVIRI Monitor. This in not a good solution for effectively managing the raw satellite imagery, the generated products and the static GIS layers. A more robust and user-friendly management system is needed that will allow the integration and customization of the available capacities.

3. IMPROVING THE FIRE MONITORING APPLICATION OF NOA USING TELEIOS TECHNOLOGIES

In this section we describe the implementation of the fire monitoring service of NOA using TELEIOS technologies. Let us describe briefly the improvements that have been done. First, loading can be performed without any preprocessing of raw data because the data vault module has been developed that transforms input data into SciQL arrays. Secondly, the processing chain and other operations such as georeferencing, cropping images and classification of measurements have been implemented using SciQL. This leads to more expressive queries that can easily be changed if needed. Finally, using Strabon and combining standard products with auxiliary data enables a user to easily create added-value thematic maps and increase their thematic accuracy.

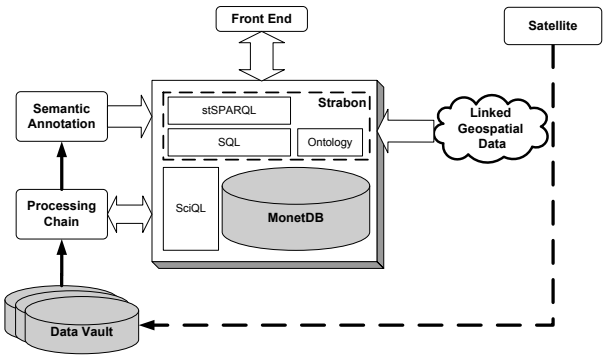


Figure 4: The improved fire monitoring service

Figure 4 depicts the new fire monitoring application of NOA developed in TELEIOS. The system consists of the following parts:

- The data vault which is responsible for the ingestion policy and enables the efficient access to large archives of image data and metadata in a fully transparent way, without worrying for their format, size and location.

- The back-end of the system. The back-end relies on MonetDB for two tasks: (i) the implementation of the hotspot detection processing chain (using the SciQL front-end) and (ii) the evaluation of semantic queries for improving the accuracy of the product shapefiles and generating thematic maps (using a stSPARQL front-end, i.e., Strabon).

- A geospatial ontology which links the generated hotspot products with stationary GIS data (Corine Land Cover, Coastline, Greek Administrative Geography), and with linked geospatial data available on the web (LinkedGeoData, GeoNames). This ontology is expressed in OWL. This ontology is described in more detail in Section 3.2.1 below.

- The front-end interface, for controlling the back-end functionality with user-friendly tools, and disseminating the products to the end-user community. A visual query builder is currently being developed as well to allow NOA personnel to express complex stSPARQL queries.

Let us now describe in more detail two of the more interesting, from a database perspective, functionalities of the fire monitoring service: the implementation of the hotspot detection processing chain using data vaults and SciQL, and the use of stSPARQL to query the generated products and to combine them with other geospatial data.

3.1 The processing chain

The processing chain as described in Section 2 comprises of the following submodules: (a) ingestion, (b) cropping, (c) georeference, (d) classification, and (e) output generation. All submodules are implemented inside the MonetDB DBMS using SciQL. In the following we describe each of them in detail.

3.1.1 Loading

One of the major issues that arise when dealing with earth observation data is the abundance of available file formats. In this particular application the input format is High Rate Information Transmission (HRIT) or Low Rate Information Transmission (LRIT). These are the CGMS standards agreed upon by satellite operators for the dissemination of digital data originating from geostationary satellites to users via direct broadcast. Loading such data requires an external program that transforms the original satellite image format into a representation as table or array that the DBMS can handle. The reason therefore is that DBMSs in general do not know anything about any external file formats. Thus, the knowledge of how to convert a given file format into a relational table or an array needs to be available and kept outside the DBMS. This can be a major hurdle, not only in terms of inconvenience for the user, but also in terms of performance. All external files that are to be loaded into the database first need to be converted entirely, even if not all files, or not all data of each file, to the appropriate format required for query processing at a subsequent stage. As a first solution, we exploited the extensibility of MonetDB and developed an extension module that can load a satellite image given as HRIT file into an SQL table or SciQL array. The module provides a user-defined SQL/SciQL function "HRIT_load_image()" that returns the table/array. The function expects as parameters URIs indicating the location of the respective image files. A similar function, returning an SQL table, is responsible for reading image metadata such as number of rows, columns and bands.

The Data Vault goes one step further, into a more generic solution that addresses the principal problem of ingestion of data from external file formats into database tables or arrays. The main idea of the Data Vault is to make the DBMS aware of external file formats and keep the knowledge how to convert data from external file formats into database tables or arrays inside the database. With this, inserting external files (of known format) into the database basically consists of copying the files "as-is" into a directory that is under exclusive control of the database. Only after issuing queries that actually access data of a certain file, the DBMS will take care of loading the data from the file into the respective table or array.

3.1.2 Cropping and georeference

The classification algorithm used for the fire monitoring application requires as input IR bands 3.9 and 10.8. Following the data loading step, both bands are stored into a SciQL array. The input of these two bands is subsequently transformed into temperature values. Thus, it is safe to assume that the input looks like the arrays created by the following SciQL statements:

```
CREATE ARRAY hrit_T039_image_array
            (x INTEGER DIMENSION, y INTEGER DIMENSION, v FLOAT);
CREATE ARRAY hrit_T108_image_array
            (x INTEGER DIMENSION, y INTEGER DIMENSION, v FLOAT);
```

NOA is interested only in a specific part of the image that is received from the satellite. Cropping only the relevant parts of the image which contain the area of interest is performed in a straightforward manner using a range query. Cropping the image early on, significantly reduces the input size of the remaining image processing operations and thus the time required for the execution of the processing chain.

```
CREATE ARRAY hrit_T039_image_array
            (x INTEGER DIMENSION, y INTEGER DIMENSION, v FLOAT);
CREATE ARRAY hrit_T108_image_array
            (x INTEGER DIMENSION, y INTEGER DIMENSION, v FLOAT);

SELECT [x], [y],
 CASE
  WHEN v039 > 310 AND v039 - v108 > 10 AND v039_std_dev > 4   AND
       v108_std_dev < 2
   THEN 2
  WHEN v039 > 310 AND v039 - v108 >  8 AND v039_std_dev > 2.5 AND
       v108_std_dev < 2
   THEN 1
  ELSE  0
 END AS confidence
FROM (
 SELECT [x], [y], v039, v108,
  SQRT( v039_sqr_mean - v039_mean * v039_mean ) AS v039_std_dev,
  SQRT( v108_sqr_mean - v108_mean * v108_mean ) AS v108_std_dev
 FROM (
  SELECT [x], [y], v039, v108,
   AVG( v039 ) AS v039_mean, AVG( v039 * v039 ) AS v039_sqr_mean,
   AVG( v108 ) AS v018_mean, AVG( v108 * v108 ) AS v108_sqr_mean
  FROM (
   SELECT [T039.x], [T039.y], T039.v AS v039, T108.v AS v108
   FROM hrit_T039_image_array AS T039
   JOIN hrit_T108_image_array AS T108
     ON T039.x = T108.x AND T039.y = T108.y
  ) AS image_array
  GROUP BY image_array[x-1:x+2][y-1:y+2]
 ) AS tmp1;
) AS tmp2
```

Figure 5: Fire detection algorithm in SciQL

After the cropping operation the algorithm georeferences the image by transforming it to a new image where the location of each pixel is well known. The MSG satellite is geostationary, so in effect remains stationary above a point on the earth. Thus, after the necessary transformation has been calculated by hand, every image can be transformed in exactly the same way. The NOA application resamples the image into a slightly larger size and applies a two degree polynomial in order to map pixels of the old image to the pixels of the new image. The coefficients of the polynomial as well as the target image dimensions are all precalculated. The implementation of these operations is expressed in a very concise way using SciQL.

3.1.3 Classification

The fire classification module of the processing chain receives as input the cropped, resampled and georeferenced image with the two pixel temperatures, each derived from one band. The algorithm [4] slides a 3x3 window over every pixel of the image and computes the standard deviation of the temperatures inside the window. Figure 5 shows the classification algorithm in SciQL.

The query first computes for each pixel the standard deviation for each of the two bands. It uses the structural grouping capabilities of the SciQL, in order to gather for each pixel the values of its neighbors inside a 3x3 window. The classification process outputs a per-pixel value of 0, 1, or 2. The value 2 denotes fire, value 1 denotes potential fire while 0 denotes no fire. The decision is based on thresholding. A set of 4 thresholds, one for the temperature of the IR 3.9 band, one for the difference between the temperatures of the IR 3.9 and the IR 10.8 band, and two for the standard deviations of the two temperatures, are used for the classification of the pixel. The actual choice of thresholds used in the figure are those for an image acquired during

the day. During the night a different set of thresholds is used. âĂIJDayâĂİ is defined with a local solar zenith angle lower than 70° while âĂIJnightâĂİ with a solar zenith angle of higher than 90°. For solar zenith angles between 70° and 90° the thresholds are linearly interpolated. While not shown in the query, the solar zenith angle is computed on a per-pixel basis given the image acquisition timestamp and the exact location of the pixel which is already known after the georefencing step.

3.1.4 Output generation

The final output is produced by a SciQL query which selects pixels classified as fire or potential fire and outputs a POLYGON description in Well-known Text (WKT) format. The location of each pixel is already known after the georeference step and its shape is a 4x4 km square.

3.2 stRDF and stSPARQL in the NOA application

In TELEIOS, standard products produced by processing chains of EO data centers can be combined with auxiliary data to offer to users functionalities that go beyond the ones currently available to them (see Figure 1(b)). In this section we give concrete examples of this by showing how to improve the outputs of the hotspot detection processing chain discussed above. We start by presenting an ontology for annotating NOA standard products. Then, we present one by one all the geospatial datasets utilized in the fire monitoring application. Last, we present stSPARQL queries that improve the accuracy of NOA standard products and enable us to produce rich thematic maps.

3.2.1 Ontology for NOA standard products

To annotate semantically standard products produced by the hotspot detection processing chain of NOA, we have developed an ontology (called the NOA ontology from now on). The ontology is encoded in OWL and it is publicly available[7]. The main classes of the current version of the NOA ontology, which is depicted graphically in Figure 6, are `RawData`, `Shapefile`, and `Hotspot` which represent files with raw data (e.g., sensor measurements), ESRI shapefiles which are the outputs of the hotspot detection processing chain and hotspots which are extracted from shapefiles, respectively. For interoperability purposes, these classes have been defined as subclasses of corresponding classes of the SWEET[8] ontology. Each instance of these three classes is annotated with the satellite and the sensor from which it is derived, as well as with the date and time at which it was detected. Products (instances of `Hotspot` and `Shapefile`) are also annotated with the method (processing chain) which was used for their production and with the organization which is responsible for the production (e.g., NOA). For each file (instances of `Shapefile` and `RawData`) its filename is stored. Finally, hotspots are additionally annotated with a spatial literal and a numeric (float) literal. The former corresponds to the region (pixel) where the hotspot lies and the latter indicates the confidence that a pixel is a hotspot.

3.2.2 Hotspot data

[7] `http://www.earthobservatory.eu/ontologies/noaOntology.owl/`

[8] `http://sweet.jpl.nasa.gov/ontology/`

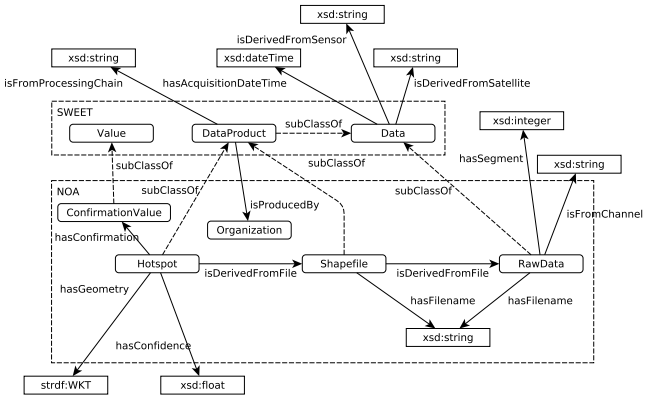


Figure 6: An ontology for NOA products

The result of the processing chain described in Section 3.1 is a collection of shapefiles. These files hold information about the coordinates of detected fire locations, the date and time of image acquisition, the level of reliability in the observations, and the names of the processing chain and the sensor that was used for the acquisition. To be able to query these shapefiles using stSPARQL and combine them with linked data available freely on the web, the produced shapefiles are first transformed into RDF. Due to the simple form of the shapefiles, each attribute of a shapefile becomes a predicate, each attribute value becomes an object and finally a subject is created as a unique URI identifying the corresponding hotspot. The following triples are an example of such information about a hotspot.

```
noa:Hotspot_1 a noa:Hotspot ;
 noa:hasAcquisitionDateTime "2007-08-24T18:15:00"^^xsd:dateTime;
 noa:hasConfidence 1.0 ; noa:hasConfirmation noa:confirmed ;
 strdf:hasGeometry "POLYGON ((21.52 37.91,21.57 37.91,21.56
  37.88,21.56 37.88,21.52 37.87,21.52 37.91))"^^strdf:geometry ;
 noa:isDerivedFromSensor "MSG2"^^xsd:string ;
 noa:isProducedBy noa:noa ;
 noa:isFromProcessingChain "cloud-masked"^^xsd:string .
```

3.2.3 Auxiliary data

We now give a short description of the auxiliary datasets utilized in the fire monitoring application.

Corine Land Cover. The Corine Land Cover project[9] is an activity of the European Environment Agency that collects data regarding the land cover of European countries. The project uses a hierarchical scheme with three levels to describe land cover. The first level indicates the major categories of land cover on the planet, e.g., forests and seminatural areas. The second level identifies more specific types of land cover, e.g., forests, while the third level narrows down to a very specific characterization, e.g., coniferous forests. The land cover of Greece is available as an ESRI shapefile that is based on this classification scheme. This shapefile is transformed in RDF as follows. Every land cover type is represented with a class (e.g., `ConiferousForest`), and the hierarchy of land cover types is expressed with the respective class taxonomy. For each specific area in the shapefile, a unique URI is created and it is connected with an instance of the third level. Additionally a property of each area with

[9] `http://www.eea.europa.eu/publications/COR0-landcover/`

value a spatial literal indicates its geometry. Some sample triples representing such an area are shown below.

```
clc:Area_45 a clc:Area ;
    strdf:hasGeometry "POLYGON ((22.07 40.62,
                          ..., 22.07 40.62))"^^strdf:geometry ;
    clc:hasLandUse clc:coniferousForest .
```

Coastline of Greece. This is an ESRI shapefile describing the geometry of the coastline of Greece. For each polygon contained in the shapefile, a unique URI is created and a spatial literal is attributed to it. The spatial literal corresponds to the closed polygon which defines the underlined area. For example:

```
coast:Coastline_1 a coast:Coastline ;
     strdf:hasGeometry "POLYGON ((24.12 34.80, ...,
                               24.12 34.80))"^^strdf:geometry .
```

Greek Administrative Geography. This is an ontology that describes the administrative divisions of Greece (prefecture, municipality, district, etc.). The ontology has been populated with relevant data that are available in Greek open government data portal[10]. For each administrative unit in the ontology (e.g., a municipality) various pieces of information are available (e.g., population and geographical boundaries). The following is a small example of such kind of information for the municipality of Athens.

```
gag:munAthens a gag:Municipality ;
        rdfs:label "Athens" ;
        gag:hasPopulation "655780"^^xsd:integer ;
        gag:isPartOf gag:preAttica ;
        strdf:hasGeometry "POLYGON((23.74,38.03, ...,
                           23.74,38.03))"^^strdf:geometry .
```

LinkedGeoData. LinkedGeoData (LGD)[11] is a project focused on publishing OpenStreetMap (OSM)[12] data as linked data. OSM maintains a global editable map that depends on users to provide the information needed for its improvement and evolution. The respective ontology is derived mainly from OSM tags, i.e., attribute-value annotations of nodes, ways, and relations. A sample from the LGD dataset describing a fire station is shown in the following triples.

```
lgd:node1119854639 a lgdo:Amenity, lgdo:FireStation, lgdo:Node;
  lgdo:directType lgdo:FireStation ;
  rdfs:label "Fire Service of Stagira - Akanthos" ;
  strdf:hasGeometry "POINT(23.8778 40.4003)"^^strdf:geometry .
```

GeoNames. GeoNames[13] is a gazetteer that collects both spatial and thematic information for various placenames around the world. GeoNames data is available through various Web services but it is also published as linked data. The features in GeoNames are interlinked with each other defining regions that are inside the underlined feature (*children*), neighboring countries (*neighbors*) or features that have certain distance with the underlined feature (*nearby features*). A sample from the GeoNames dataset describing the city of Patras is shown in the following triples.

[10] `http://geodata.gov.gr/`
[11] `http://linkedgeodata.org/`
[12] `http://www.openstreetmap.org/`
[13] `http://www.geonames.org/`

```
<http://sws.geonames.org/255683/> a gn:Feature ;
  gn:alternateName "Patrae", "Patras"@en ;
  gn:name "Patras" ; gn:countryCode "GR" ;
  gn:featureClass gn:P ; gn:featureCode gnP.PPLA ;
  gn:parentADM1 <http://sws.geonames.org/6697810/> ;
  gn:parentCountry <http://sws.geonames.org/390903/> ;
  strdf:hasGeometry "POINT(21.73 38.24)"^^strdf:geometry .
```

3.2.4 Improving hotspot products using linked data

Let us now see how the datasets presented above can be combined to improve the thematic accuracy of the generated hotspot products enabling the automatic generation of related thematic maps.

Improving the thematic accuracy. The thematic accuracy of the shapefiles generated by the processing chain is improved by an additional processing step that refines them by correlating them with auxiliary geospatial data. This is done by a series of stSPARQL update statements that update the RDF representation of the generated shapefiles by taking into account relevant RDF data sets from the ones presented above. As an example, consider the following update query.

```
DELETE {?h ?property ?object}
WHERE {
 ?h a noa:Hotspot; strdf:hasGeometry ?hGeo; ?hProperty ?hObject.
 OPTIONAL {?c a coast:Coastline ; strdf:hasGeometry ?cGeo .
 FILTER (strdf:anyInteract(?hGeo, ?cGeo))}
FILTER(!bound(?c))}
```

The condition in the first FILTER pattern of this statement utilizes the function `strdf:anyInteract` which checks if two spatial literals intersect with each other, while the condition in the second FILTER pattern ensures that retrieved hotspots do not intersect with land. Thus, it retrieves and deletes hotspots lying entirely in the sea. Similarly, the following update statement retrieves hotspots that partially lie in the sea and deletes the part of their geometry that lies in the sea.

```
DELETE {?h strdf:hasGeometry ?hGeo}
INSERT {?h strdf:hasGeometry ?dif}
WHERE {
  SELECT DISTINCT ?h ?hGeo
        (strdf:intersection(?hGeo, strdf:union(?cGeo)) AS ?dif)
  WHERE { ?h a noa:Hotspot ; strdf:hasGeometry ?hGeo .
          ?c a coast:Coastline ; strdf:hasGeometry ?cGeo .
          FILTER(strdf:anyInteract(?hGeo, ?cGeo))}
  GROUP BY ?h ?hGeo
  HAVING strdf:overlap(?hGeo, strdf:union(?cGeo))}
```

In the above query, the spatial aggregate function `strdf:union` of stSPARQL is utilized. For each hotspot all coastline regions that intersect with it are grouped and their union is calculated. Afterwards the part that is not contained in this union is deleted from the geometry of the hotspot.

Improving automatic map generation. In Section 2 we explained that the automatic generation of fire maps enriched with relevant geo-information is of paramount importance to NOA since the creation of such maps in the past has been a manual process. Using an stSPARQL endpoint where the RDF datasets described above reside, a NOA operator can now simply overlay the retrieved data using some GIS software (e.g., QGIS or GoogleEarth). For example, by posing the following queries and overlaying their results, NOA operators can create a map like the one shown in Figure 7 that exploits information from the above datasets.

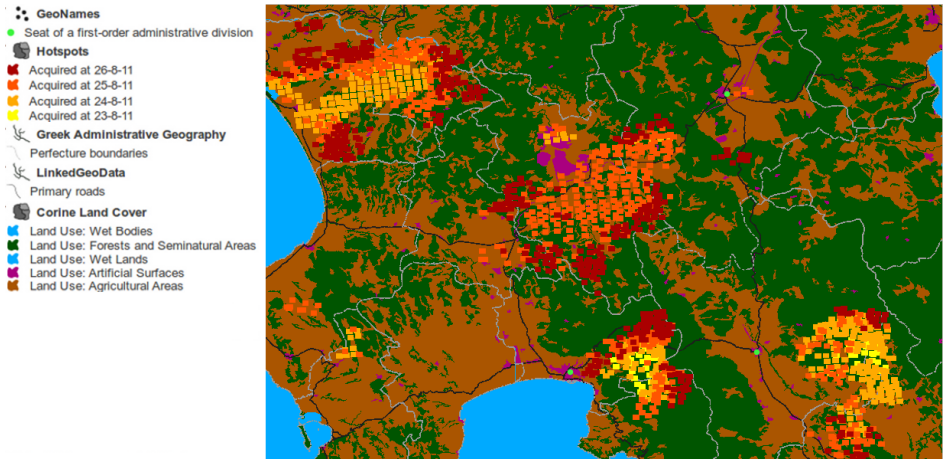


Figure 7: A map that can be created by overlaying data computed by stSPARQL queries

Query 1: "Get all hotspots in southeastern Peloponnese that were detected from 23rd to 26th of August 2007" (from Hotspot data).

Query 2: "Get the land cover in southeastern Peloponnese" (from Corine Land Cover data).

Query 3: "Get all primary roads in southeastern Peloponnese" (from LinkedGeoData).

Query 4: "Get all capitals of prefectures of southeastern Peloponnese" (from GeoNames).

Query 5: "Get all municipality boundaries in southeastern Peloponnese" (from Greek Administrative Geography).

Due to space considerations, we give only Query 4 in stSPARQL:

```
SELECT ?n ?nName ?nGeo
WHERE { ?n a gn:Feature ;
        strdf:hasGeometry ?nGeo ;
        gn:name ?nName ; gn:featureCode gn:P.PPLA .
 FILTER(strdf:contains("POLYGON((21.67 36.87, 22.74 36.87,
 22.74 37.68,21.67 37.68,21.67 36.87))"^^strdf:geometry, ?nGeo))}
```

This query asks for every feature in the GeoNames dataset that is contained in a specific rectangle covering southeastern Peloponnese and its `gn:featureCode` equals to `gn:P.PPLA`, i.e., the feature is a first-order administrative division (for Greece this corresponds to capitals of prefectures). Apart from thematic information about a node (variables `?n` and `?nName`), the geometry (variable `?nGeo`) of the feature is also returned so it can be depicted on a map.

4. RELATED WORK

TELEIOS is a multidisciplinary research effort bringing together contributions from database management, semantic web, remote sensing and knowledge discovery from satellite images. We now review some of the most relevant research efforts in these areas, and compare them with the work carried out in TELEIOS which has been presented in this paper.

With respect to systems offering array query processing capabilities there are only few systems that can handle sizable arrays efficiently. RasDaMan [2] is a domain-independent array DBMS for multidimensional arrays of arbitrary size and structure. RasDaMan provides a SQL-92 based query language, RasQL [1], to manipulate raster images using foreign function implementations and provides raster web services which are based on OGC standards. Such web services are beyond the scope of TELEIOS. A recent attempt to develop an array database system from scratch is undertaken by the SciDB group [16]. Its mission is the closest to SciQL, but Version 0.5 and the design documents indicate that their language is a mix of SQL syntax and algebraic operator trees, instead of a seamless integration with SQL:2003 syntax and semantics. SciQL takes language design a step further by providing a seamless symbiosis of array-, set-, and sequence- interpretation using a clear separation of the mathematical object from its underlying implementation. A key innovation of SciQL is the extension of value-based grouping in SQL:2003 with structural grouping which leads to a generalization of window-based query processing with wide applicability in science domains.

In the context of the Semantic Web, the development of geospatial extensions to SPARQL has received some attention recently which resulted in the creation of a forthcoming OGC standard for querying geospatial data encoded in RDF, called GeoSPARQL [10]. GeoSPARQL draws on the concepts developed in earlier languages such as SPARQL-ST [11], SPAUK [6] and the original version of stSPARQL [8].

There have been some works in the past where ontologies have been applied to the modeling of EO data [12, 3] or in a similar virtual observatory context [13, 9]. TELEIOS has benefited from the modeling concepts developed in these efforts and has tried to re-use parts of these public ontologies whenever possible.

In the area of remote sensing, most of the fire detection algorithms using MSG/SEVIRI data are based on variations of EUMETSAT's classification methodology [4] (EUMETSAT is the European organization managing the Meteosat series of geostationary meteorological satellites). The pre-TELEIOS approach used by NOA for this problem has been discussed in detail in [15, 7].

Finally, the vision of having knowledge discovery and data mining from satellite images as a fundamental capability of information systems for today's EO data centers has been stressed in earlier project KEO/KIM[14] funded by the European Space Agency, and the US project GeoIRIS [14]. Compared to these projects, TELEIOS has a much stronger technical foundation because it builds on state of the art database and semantic web technologies, as well as more advanced knowledge discovery and data mining techniques.

5. CONCLUSIONS

In this paper we report on a virtual earth observatory that we are currently building in the context of the European project TELEIOS. Given the rapidly growing Earth Observation data archives, TELEIOS addresses the need for scalable access to petabytes of Earth Observation data and the discovery of knowledge that can be used in applications. To achieve this, TELEIOS aims at leveraging and extending data management technologies. The main focus is on scientific database technologies (array databases, SciQL, data vaults) and on geospatial Semantic Web technologies (stRDF and stSPARQL). Using a forest fire monitoring application as representative example, we discuss in detail how the developed technologies, integrated into MonetDB, a state-of-the-art open-source column-store database system, can be deployed to support and improve processing of large-scale Earth Observation data. While focusing on Earth Observation within the TELEIOS project, we are confident that the developed technologies can also be deployed in other scientific disciplines like astronomy, meteorology, seismology, biology, etc.

6. ACKNOWLEDGMENTS

This work has been funded by the FP7 project TELEIOS (257662).

7. ADDITIONAL AUTHORS

Harokopio University of Athens (Greece): Dimitrios Michail;
NKUA: K. Kyzirakos, M. Karpathiotakis, C. Nikolaou,
G. Garbis, K. Bereta, M. Sioutis;
S. Giannakopoulou, P. Smeros, K. Dogani;
NOA: I. Papoutsis, T. Herekakis;
CWI: M. Kersten, M. Ivanova, H. Pirk, Y. Zhang;

8. REFERENCES

[1] P. Baumann. A database array algebra for spatio-temporal data and beyond. In *Next Generation Information Technologies and Systems*, pages 76–93, 1999.

[2] P. Baumann, A. Dehmel, P. Furtado, R. Ritsch, and N. Widmann. The Multidimensional Database System RasDaMan. In *SIGMOD*, pages 575–577, 1998.

[3] C. Carlino, S. D. Elia, A. D. Vecchia, M. Iacovella, M. Iapaolo, C. M. Scalzo, and F. Verdino. On sharing Earth Observation concepts via ontology. In *ESA-EUSC*, 2008.

[4] EUMETSAT. Active Fire Monitoring with MSG Algorithm - Theoretical Basis Document. Technical report, EUM/MET/REP/07/0170, 2007.

[5] G. Garbis et al. An Implementation of a Temporal and Spatial Extension of RDF and SPARQL on top of MonetDB - Phase I. Deliverable 4.1, TELEIOS project, 2012.

[6] D. Kolas and T. Self. Spatially-augmented knowledgebase. In *ISWC/ASWC*, pages 792–801, 2007.

[7] C. Kontoes et al. Wildfire monitoring via the integration of remote sensing with innovative information technologies. In G. R. Abstracts, editor, *EGU2012-PREVIEW*, 2012.

[8] M. Koubarakis and K. Kyzirakos. Modeling and Querying Metadata in the Semantic Sensor Web: The Model stRDF and the Query Language stSPARQL. In *ESWC*, pages 425–439, 2010.

[9] D. L. McGuinness et al. The virtual solar-terrestrial observatory: A deployed semantic web application case study for scientific research. In *AAAI*, 2007.

[10] O. G. C. I. OGC. GeoSPARQL - A geographic query language for RDF data, November 2010.

[11] M. Perry. *A Framework to Support Spatial, Temporal and Thematic Analytics over Semantic Web Data*. PhD thesis, Wright State University, 2008.

[12] M. Podwyszynski. Knowledge-based search for Earth Observation products. Master's thesis, Passau Univ., 2009.

[13] R. Raskin and M. Pan. Knowledge representation in the semantic web for Earth and environmental terminology (SWEET). *Computers & Geosciences*, 2005.

[14] C.-R. Shyu et al. GeoIRIS: Geospatial Information Retrieval and Indexing System - Content Mining, Semantics Modeling, and Complex Queries. *IEEE TGRS*, 2007.

[15] N. Sifakis, C. Iossifidis, C. Kontoes, and I. Keramitsoglou. Wildfire Detection and Tracking over Greece Using MSG-SEVIRI Satellite Data. *Remote Sensing*, 2011.

[16] M. Stonebraker, J. Becla, D. J. DeWitt, K.-T. Lim, D. Maier, O. Ratzesberger, and S. B. Zdonik. Requirements for Science Data Bases and SciDB. In *CIDR*, 2009.

[17] Y. Zhang et al. An implementation of ad-hoc array queries on top of MonetDB. Del. D5.1, TELEIOS, 2012.

[14] `http://earth.esa.int/rtd/Projects/KEO/index.html/`