



PUBLICAMUNDI
SCALABLE
REUSABLE
OPEN
GEOSPATIAL
DATA

REPORT FOR

DELIVERABLE D2.2

I INTRODUCTION

This report provides an overview of the technical characteristics and functionality of deliverable D2.2 “CKAN Plugins”. Its purpose is present the functionalities of all CKAN plugins extended or developed by the PublicaMundi project. In particular:

- We highlight our contributions in existing CKAN plugins
- We present the new CKAN plugins, extensions, and APIs developed specifically for the needs of PublicaMundi

The reader is encouraged to visit the software’s repository (<https://github.com/PublicaMundi>) to receive:

- Up-to-date versions of the software, along with documentation targeted to developers
- Detailed information regarding all development effort (commits, activity, issues)
- Instructions regarding the installation of the software and its dependencies



2 INTRODUCTION

PublicaMundi aims to make open geospatial data easier to discover, reuse, and share by fully supporting their complete publishing lifecycle in open data catalogues. To achieve this, we are extending and integrating leading open source software for open data publishing and geospatial data management.

The goal of Work Package 2 (WP2) was to develop technologies to efficiently support open geospatial data across their lifecycle in CKAN, focusing on vector data. The work package aimed to extend and develop CKAN through its plugin architecture in order to: (a) design and implement an efficient and sustainable publishing workflow, (b) integrate geospatial metadata/data management/querying facilities, and (c) develop geospatial data interlinking and multilinguality-support tools.

During the third quarter of the project, the developed software was rolled-out to geodata.gov.gr. We focused on stabilizing our software, integrate more geospatial functionality on CKAN (<http://ckan.org/>), and deploy the stable part of the system into production, providing open access to data publishers and developers for Greek open geospatial data.

For the purpose of testing the developed CKAN plugins before rolling out to production deployment, we perform continuous beta deployments on labs.geodata.gov.gr, and through automated procedures the tested software is migrated to the production environment (geodata.gov.gr).



3 CONTRIBUTIONS TO EXISTING PLUGINS

As described in Deliverable 2.1, CKAN is a Pylons¹ (*Python software framework*) application, providing a mechanism to add functionality through extensions. The CKAN extensions consist of many (or one) CKAN plugins which usually fulfill software dependencies and are used as software modules, under a common hierarchy of files. The plugins are enabled or disabled by the core configuration files of the CKAN project and may have independent configuration files according to their complexity and functionality. This way the core application is improved but at the same time the source code is easier to be maintained through mainstream code revisions (releases).

Our main objective in PublicaMundi was our software to be reusable and extensible, while maintaining our goal to spatially enable CKAN. We focused on maintaining compatibility with upstream CKAN core, so that our software would be easily applicable not only to geodata.gov.gr open data portal but also on other CKAN deployments, without breaking existing functionality.

In order to achieve the goals set by WP2, we extended many existing CKAN plugins, to make them work with our software architecture and to comply with the needed functionality. In the following we present the CKAN plugins that are being used for our beta and production deployment, including some technical details on what was used and what exactly was extended under PublicaMundi.

3.1 CKANEXT-DATASTORE

The CKAN DataStore plugin provides an ad hoc database for storage of structured data from CKAN resources. Data can be pulled out of resource files and stored in the DataStore. When a resource is added to the DataStore, automatic data previews are activated on the resource's page. This plugin also includes an API to search, filter and update the data, without having to download and upload the entire data file. The DataStore is integrated into the CKAN API and authorization system.

This plugin was extended during this project (through new plugins that depend on it - *ckanext-publicamundi* and *ckanext-multilinguality*), to provide automatic data previews to the spatial data stored in the spatially enabled database (PostGIS). In addition, this plugin was also extended to provide multiple database views from translated datasets.

Source code is available here:

¹ <http://www.pylonsproject.org/>



<https://github.com/PublicaMundi/ckan/tree/dev.publicamundi.eu/ckanext/dastore>

3.2 CKANEXT-MULTILINGUAL

The CKAN Multilingual plugin is used for translating CKAN's web interface. In addition to user interface internationalization, a CKAN administrator can also enter translations into CKAN's database for terms that may appear in the contents of datasets, groups or tags created by users. When a user is viewing the CKAN site, if the translation terms database contains a translation in the user's language for the name or description of a dataset or resource, the name of a tag or group, etc., then the translated term will be shown to the user in place of the original.

This plugin was extended to translate INSPIRE themes and dataset groups. Furthermore we added support for the Greek language and extended the plugin API to be used by the CKAN toolkit functions.

Source code is available here:

<https://github.com/PublicaMundi/ckan/tree/dev.publicamundi.eu/ckanext/multilingual>

3.3 CKANEXT-RECLINEPREVIEW

The CKAN Recline plugin is used for previewing structured data in CKAN. The Data Explorer JavaScript application provides a rich, queryable view of the data, and allows the data to be filtered, graphed and mapped. To be viewed, the data must either be in the CKAN DataStore or in csv or xls format.

This plugin was extended in PublicaMundi to support filtering capabilities to specific default formats. This way, the plugin can be configured not to preview certain datasets that are previewed by other plugins by default (*e.g. in cases of translated datasets or datasets that are derived from interlinking*)

Source code is available here:

<https://github.com/PublicaMundi/ckan/tree/dev.publicamundi.eu/ckanext/reclinepreview>



3.4 CKANEXT-ARCHIVER

The CKAN Archiver provides a set of Celery tasks for downloading and saving CKAN resources. It can be configured to run automatically, saving any new resources that are added to a CKAN instance (and saving any resources when their URL is changed). It can also be run manually from the command line in order to archive resources for specific datasets, or to archive all resources in a CKAN instance.

The archiver plugin was extended to handle update and empty notifications for new resources, as well as to update resource metadata in cases the resource has changed by checking the sha1 hash. Also, some security patches have been applied and submitted upstream.

Source code is available here:

<https://github.com/PublicaMundi/ckanext-archiver>

3.5 CKANEXT-DATASTORER

The CKAN Datastorer provides a Celery task for automatically saving CKAN resources that link to csv and excel files into the datastore. The Datastorer plugin was modified to filter resources by resource-format and not only by MIME type. Additionally, in order to support Shapefiles that usually are packaged into zip files (or similar archives), datastorer was extended to support zip files directly, and thus be able to store the .dbf files (including attribute table) into the datastore.

Source code is available here:

<https://github.com/PublicaMundi/ckanext-datastorer>

3.6 CKANEXT-GOOGLEANALYTICS

CKAN-googleanalytics integrates Google Analytics data into CKAN. It provides download statistics on package pages, a list of most popular packages, and many additional statistics. This CKAN extension both sends tracking data to Google Analytics and retrieves statistics from Google Analytics inserting them into CKAN pages.

This plugin is used in our production environment to provide dataset popularity analytics. The plugin was modified to avoid API conflicts with our own analytics plugin (presented below).

Source code is available here:



<https://github.com/PublicaMundi/ckanext-googleanalytics>

3.7 CKANEXT-HARVEST

This is a plugin used to provide a common harvesting framework for other CKAN extensions and adds a command line interface and a graphical user interface to CKAN to manage harvesting sources and jobs.

This plugin is used in our production environment as a dependency to the ckanext-spatial extension for harvesting spatial datasets from various sources like CSW catalogues, OpenDataJSON, WAF etc. This plugin *was not extended* but used as a framework for our own spatial harvesters.

Source code is available here:

<https://github.com/PublicaMundi/ckanext-harvest>

3.8 CKANEXT-SPATIAL

This extension contains plugins that add geospatial capabilities to CKAN. The spatial extension adds a spatial field to the default CKAN dataset schema, using PostGIS as the backend. This allows performing spatial queries and displaying the dataset geospatial extent on the frontend. It also provides harvesters to import geospatial metadata into CKAN from other sources, as well as commands to support the OGC CSW standard via pycsw.

More specifically, the ckanext-spatial extension includes the following plugins:

- SpatialMetadata plugin, which extends the CKAN metadata model to include a spatial field to store the BBOX or the geometry linked to a dataset
- SpatialQuery plugin, which can perform a PostGIS or a SOLR based spatial query to display results from a spatial search in the catalogue
- CatalogueServiceWeb plugin, which links harvested datasets to the pycsw CSW API
- HarvestMetadataAPI plugin, which implements spatial harvesting on external catalogues like CSW, WAF, OpenDataJSON etc.

In the context of PublicaMundi some of these plugins were extended or replaced by new functionality included in new spatial plugins. More specifically, the data previewer in the SpatialMetadata plugin were replaced



by our own Mapping API, to provide a single interface for spatial data preview in CKAN. It includes support for both OpenLayers 3 and Leaflet JavaScript libraries, under a single unified API, to make easy for new developers to create custom maps from the CKAN catalogue datasets. The PublicaMundi Mapping API has been delivered as part of WP3 and the source code is available here:

<https://github.com/PublicaMundi/MappingAPI>

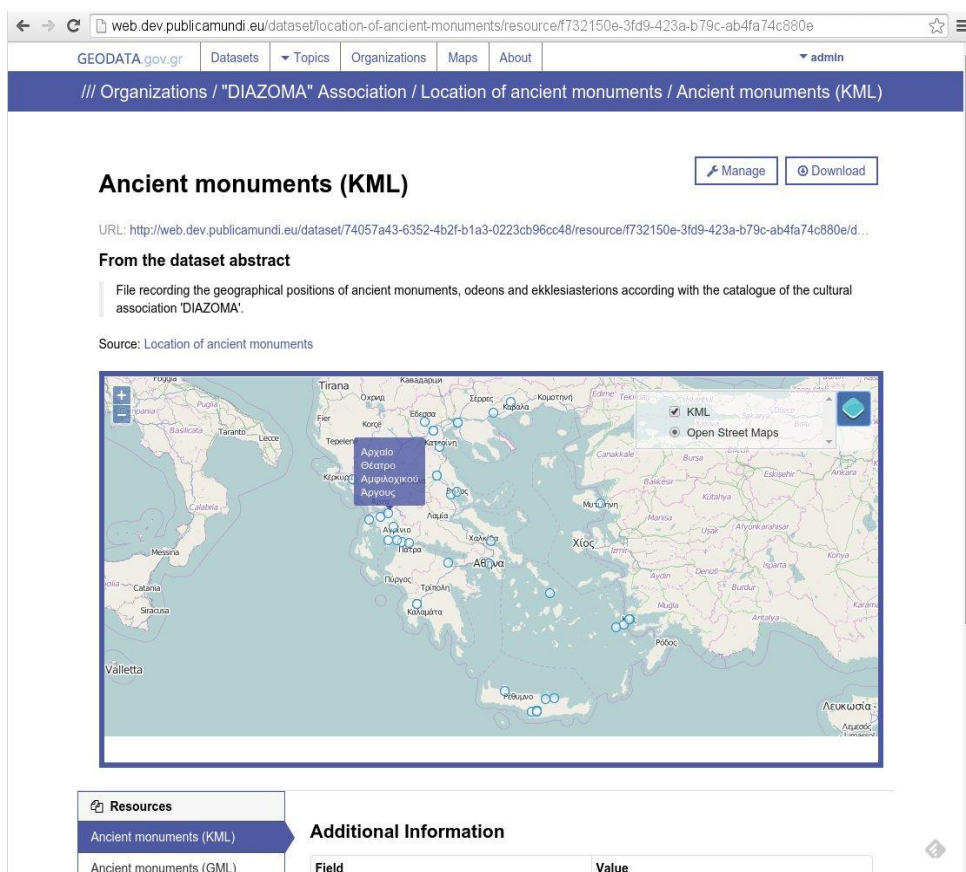


Figure 1: Geospatial data previewer based on PublicaMundi Mapping API

Furthermore, we added support for previewing GML, KML, WMS and WFS resources directly through our Mapping API. In addition, we extended the preview plugin to collect information from our vector and raster storer plugins and be able to preview rasdaman-based web services. The new plugin is named `spatial_publicamundi_preview`.

Source code is available here:

<https://github.com/PublicaMundi/ckanext-spatial>



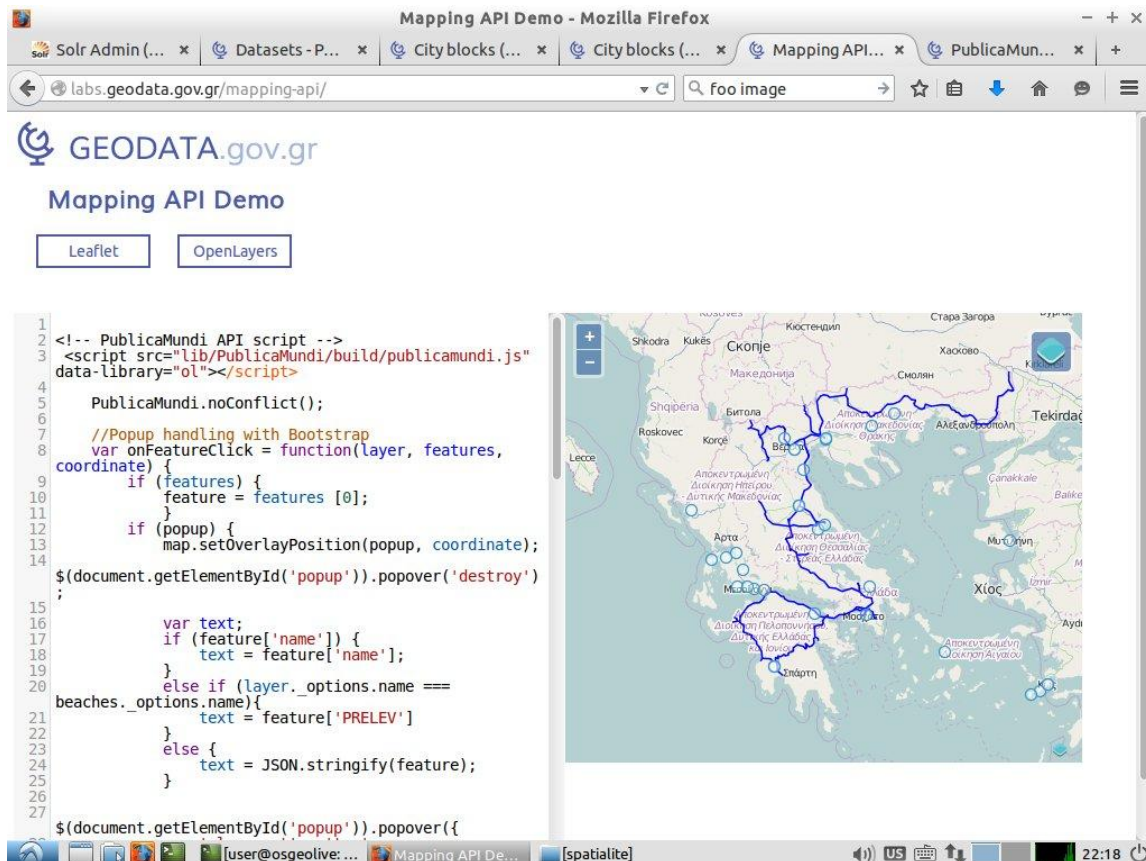


Figure 2: Mapping API demonstration page on <http://labs.geodata.gov/mapping-api> (from *PublicaMundi Workshop at FOSS4G Europe 2015 in Como, using OSGeoLive*)

All the above plugins were extended to fit our deployment needs (*thus they are forked to our GitHub account, to track our contributions and use for production deployment*). Bug fixes and generic solutions to CKAN issues were submitted back the code upstream through pull requests.



4 NEW CKAN PLUGINS

To achieve the project goals, several new CKAN plugins were developed, in order to extend CKAN into a full spatial catalogue, and implement the planned functionality of PublicaMundi.

4.1 CKANEXT-VECTORSTORER

VectorStorer is a CKAN plugin that allows users to upload vector geospatial data, as well as store and publish them through OGC services. With the vectorstorer plugin we have extended CKAN to *natively* support geospatial vector data management, by integrating PostGIS, the leading open source geospatial database. Data publishers can upload geospatial data in any format and coordinate reference system. The system automatically stores the dataset and can provide it in another data format (on-demand) or through OGC compatible services. As such, data publishers can provide any data they have at hand, without additional effort into transforming their data in specific-purpose formats. Further, as soon as the data set is published, it is automatically available for querying and visualization with no extra effort. This plugin extends the *ckanext-datastorer*, the *ckanext-datastore* and the *ckanext-spatial* plugins.

The VectorStorer plugin adds CKAN support for publishing geospatial data directly to *GeoServer and MapServer backends*. Vector data are ingested into PostGIS from within a newly implemented resource dashboard and then, additional resources are automatically created for the publisher, including OGC web services (WMS, WFS). The ingested data are available directly through the PublicaMundi Data API and the Mapping API for visualization and processing on the web.

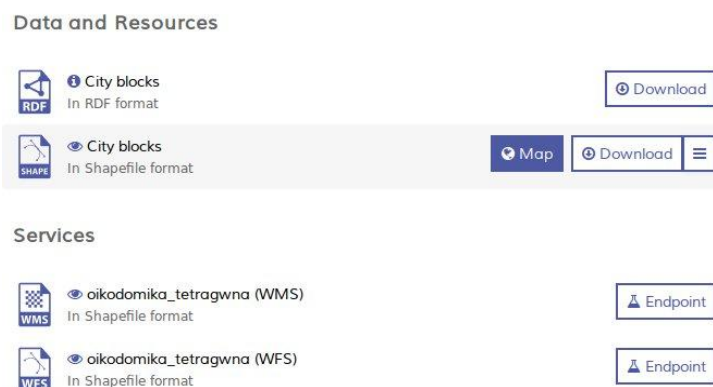


Figure 3: Automatically created OGC Services from VectorStorer in the dataset user interface.



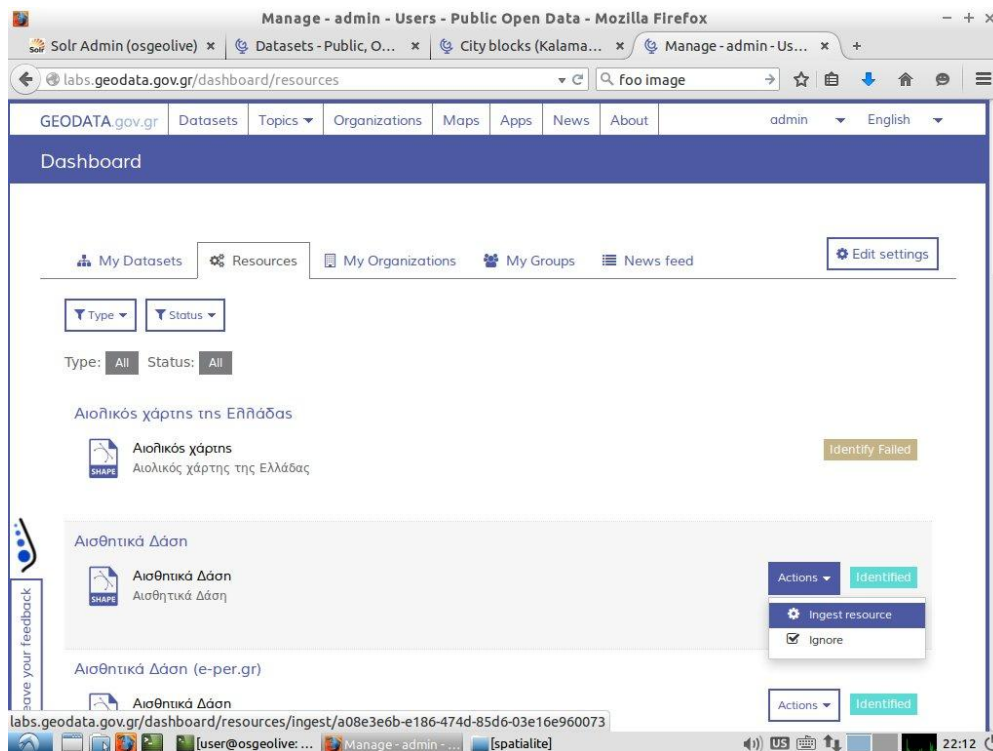


Figure 4: New CKAN administrator dashboard, for geospatial data ingestion.

The VectorStorer plugin was developed as stand-alone in the first quarter of the project but it was then merged into the ckanext-publicamundi CKAN extension.

The original source code is available here:

<https://github.com/PublicaMundi/ckanext-vectorstorer/tree/master/ckanext/vectorstorer>

and the latest version of the plugin can now be found here:

<https://github.com/PublicaMundi/ckanext-publicamundi/tree/master/ckanext/publicamundi/storers/vector>

4.2 CKANEXT-RASTERSTORER


RasterStorer is a CKAN plugin that allows raster files to be imported into a WCST service. The plugin supports any GDAL raster format (e.g., GeoTIFF, JPEG2000) and GML raster files.

The plugin was developed to extend CKAN to *natively* support raster data management, by integrating *rasdaman*, the leading big raster data analytics server. Data publishers can publish any raster data they have available, ranging from a single satellite image, to thousands of orthomaps. The raster data are efficiently managed and are available for querying and visualization with no extra effort, through the WCS, WCPS, and rasql APIs. This plugin is



seamlessly integrated into the administrative dashboard so each time a raster file is uploaded as a resource to CKAN, the data type is identified and the raster storer can ingest it directly into rasdaman.

Data and Resources


landsat_thessaloniki
 RGB composite of Landsat OLI 8

[Map](#)
[Download](#)

Services


landsat_thessaloniki (WCS)
 A WCS resource generated from landsat_thessaloniki
 [Endpoint](#)


landsat_thessaloniki (WMS)
 A WMS layer resource generated from landsat_thessaloniki
 [Endpoint](#)

Figure 5: Automatically created OGC Services from RasterStorer.

[GEODATA.gov.gr](#)
[Datasets](#)
[Topics](#)
[Organizations](#)
[Maps](#)
[Apps](#)
[News](#)
[About](#)

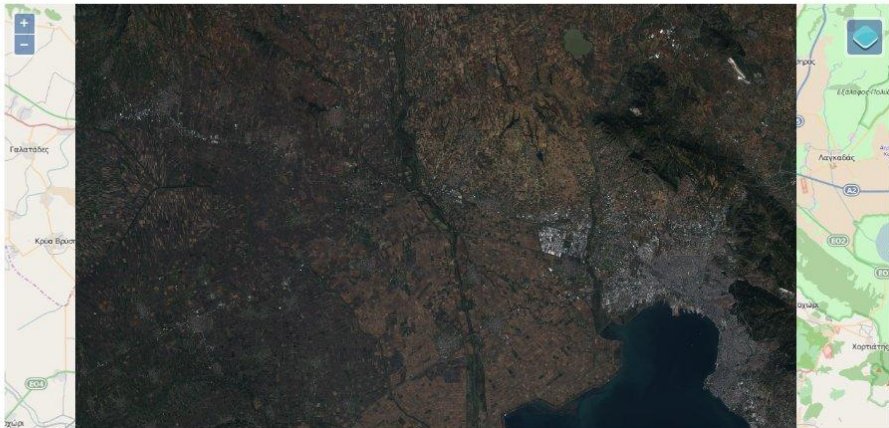
[Log in](#)
[Register](#)
[English](#)

Organizations // [landsat_thessaloniki](#)

landsat_thessaloniki

[Data API](#)
[Maps](#)
[Download](#)

URL: http://aiolos.survey.ntua.gr/tmp/raster/LC81840322014306LGN00_RGB_3857_8bit.tif
 RGB composite of Landsat OLI 8



Resources

- landsat_thessaloniki
- Thessaloniki Landsat (WCS)
- Thessaloniki Landsat (WMS)
- Share

Additional Information

Last updated	December 13, 2014
Created	December 13, 2014
Format	image/tiff

Leave your feedback

Figure 6: Preview of raster datasets directly from PublicaMundi extension.

The RasterStorer plugin was developed as stand-alone plugin in the first quarter of the project, but it was then merged into the ckanext-publicamundi CKAN extension.

The original source code is available here:



<https://github.com/PublicaMundi/ckan-rasterimport>

and the latest version of the plugin can now be found here:

<https://github.com/PublicaMundi/ckanext-publicamundi/tree/master/ckanext-publicamundi/storers/raster>

4.3 CKANEXT-WPS

Ckanext-wps is a CKAN plugin that allows users to access OGC WPS capabilities and publish execution results directly as CKAN resources.

CKAN was extended to support complex and OGC-compliant geospatial processing capabilities, by extending ZOO-WPS, the leading open source OGC WPS server. As soon as vector and raster data have been uploaded by the data publishers in the catalogue, these can be *automatically reused* through standards-compliant WPS services and publish results *as new resources* within an existing dataset.

Source code is available here:

<https://github.com/PublicaMundi/ckanext-wps>

4.4 PUBLICAMUNDI_DATASET

The publicamundi_dataset is part of the ckanext-publicamundi extension. It provides validation logic, storage logic and user interface controls for metadata described in alternative schemata (e.g., INSPIRE, ISO-19115, Dublin Core).

With this plugin, CKAN is extended with an easy to use and flexible data publishing workflow for geospatial data, supporting vector and raster data, in addition to the majority of standardized metadata schemata. Data publishers are guided step-by-step into creating metadata in their schema of choice (Figures 7,8), or importing existing metadata. The metadata they provide can then be transformed *on-the-fly* in any supported schema. If an existing data catalogue or Spatial Data Infrastructure is available (e.g. INSPIRE), the data publisher only needs to provide a simple entry point, and all available metadata are automatically harvested.

The CKAN metadata model has been extended for flexible metadata creation/editing. Publishers have several options for creating metadata aiming to minimize effort and maximize reusability. They can create metadata for several supported metadata schemata (e.g. CKAN, INSPIRE, ISO), or import existing metadata files. Following validation, metadata are



stored and can be transformed on demand in other schemata through CSW interface. This mechanism is programmatically extensible to support any other metadata schema (geospatial or not). This is a core plugin that is implemented within ckanext-publicamundi, using zope interfaces, with capabilities of automatic UI creation, validation and export of metadata files.

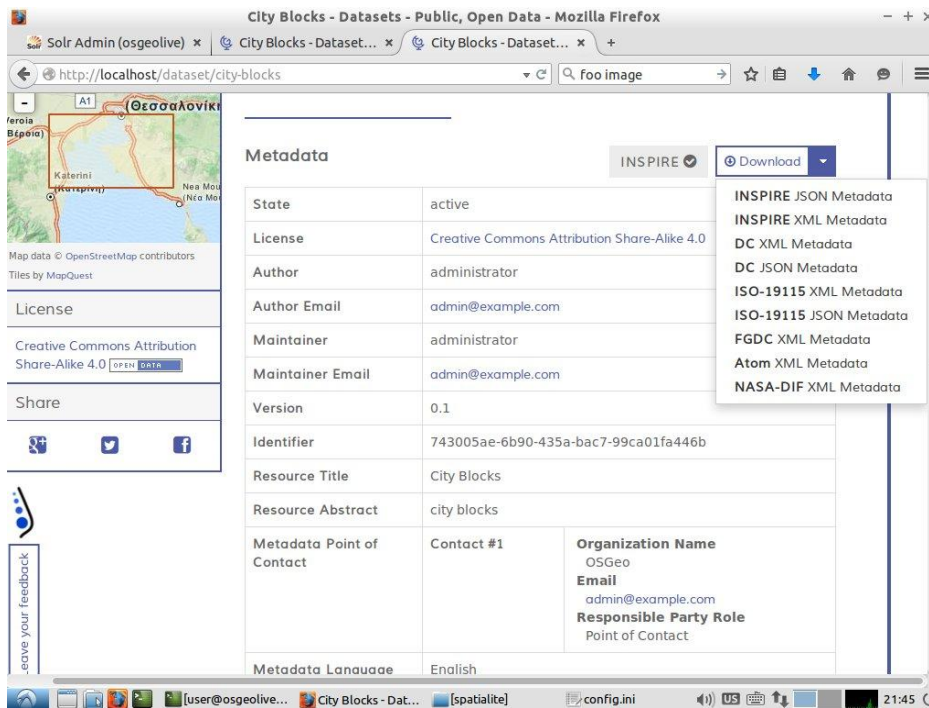


Figure 7: Metadata UI elements through the publicamundi_dataset plugin with on the fly transformation.

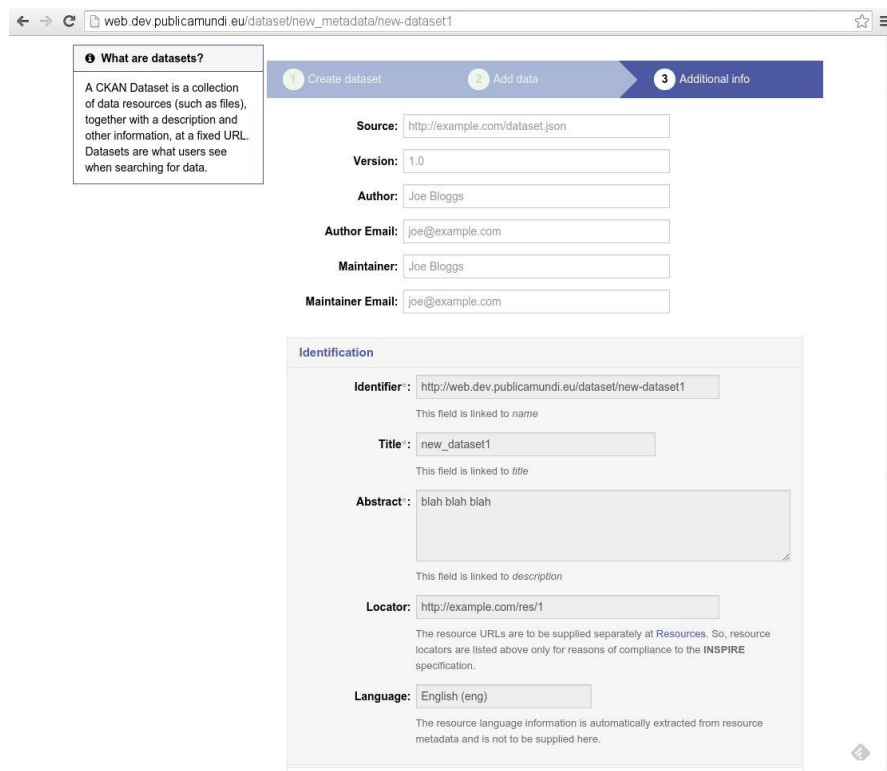


Figure 8: The PublicaMundi metadata editor.



Source code is available here:

<https://github.com/PublicaMundi/ckanext-publicamundi/tree/master/ckanext-publicamundi/lib/metadata>

Due to increased interest from the CKAN community, this plugin is currently being ported to *a separate CKAN extension (named ckanext-schematic)* to be used as stand-alone:

<https://github.com/PublicaMundi/ckanext-schematic>

4.5 PUBLICAMUNDI_PACKAGE

The `publicamundi_package` plugin provides synchronization of CKAN package metadata to other databases. This plugin is used to synchronize the CKAN database with the integrated pycsw database, thus providing a tight integration of CSW API to native CKAN datasets (*ckanext-spatial only offers support for harvested datasets*). In addition to this plugin, pycsw has been significantly extended to support OpenSearch Geo/Time specification in CKAN as well as the new CSW3.0 specification.

Source code is available here:

<https://github.com/PublicaMundi/ckanext-publicamundi/tree/master/ckanext-publicamundi/lib>

4.6 PUBLICAMUNDI_ANALYTICS

The `publicamundi_analytics` plugin provides granular geospatial analytics to CKAN that can shed light on the actual use of the geospatial services and data that is being accessed. The developed plugin displays the data analytics directly into CKAN. The analytics are gathered from a proxy that logs all access to OGC services and system APIs and processes the results into a database. The plugin then displays the results into an analytics dashboard for the system administrator.

Source code is available here:

<https://github.com/PublicaMundi/ckanext-publicamundi/tree/master/ckanext-publicamundi/analytics>

4.7 PUBLICAMUNDI_GEODATA_THEME

The `publicamundi_geodata_theme` plugin overrides the default CKAN user interface theme, providing the look and feel of the geodata.gov.gr



catalogue. As such, anyone can deploy a feature complete PublicaMundi instance, with an elegant theme exposing the full PublicaMundi-developed geospatial capabilities.

Source code is available here:

<https://github.com/PublicaMundi/ckanext-publicamundi/tree/master/ckanext/publicamundi/themes>

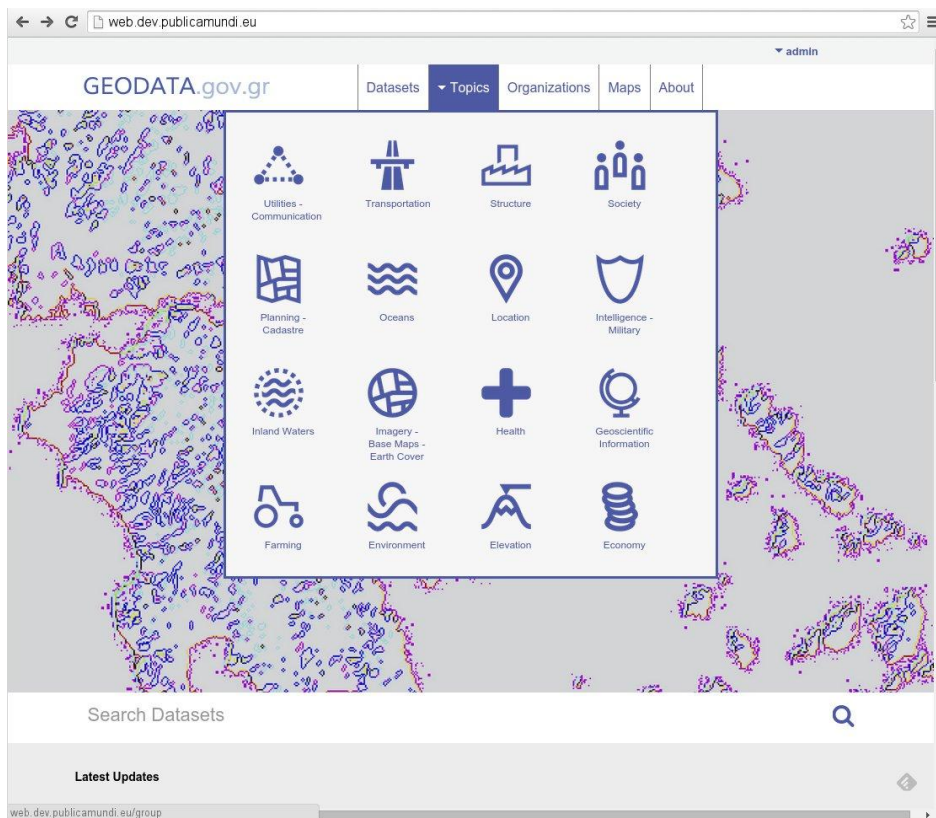


Figure 9: geodata.gov.gr theme implemented in publicamundi_geodata_theme CKAN plugin

4.8 PUBLICAMUNDI_MAPCLIENT

The publicamundi_mapclient plugin is a web based application that augments CKAN with advanced mapping features not present in the current CKAN spatial extension. It is tightly integrated with CKAN and the ckanext-publicamundi extension through the CKAN action API. The plugin provides extended support for the VectorStorer plugin by exploiting the extra metadata information and the query capabilities offered by the aforementioned extensions respectively. The MapClient plugin allows users to browse, search and preview CKAN datasets and resources in a spatial oriented perspective.

Furthermore, the MapClient plugin also includes the developed PublicaMundi Data API. This API provides to users and developers an



access point to the ingested CKAN spatial data, as well to the processing engine underneath (*GDAL, PostGIS, ZOO WPS*).

Due to increased interest from the CKAN community, in the following months the Data API code is planned to be moved and maintained into a new CKAN extension (<https://github.com/PublicaMundi/DataAPI>).

Source code for the MapClient plugin is available here:

<https://github.com/PublicaMundi/MapClient>

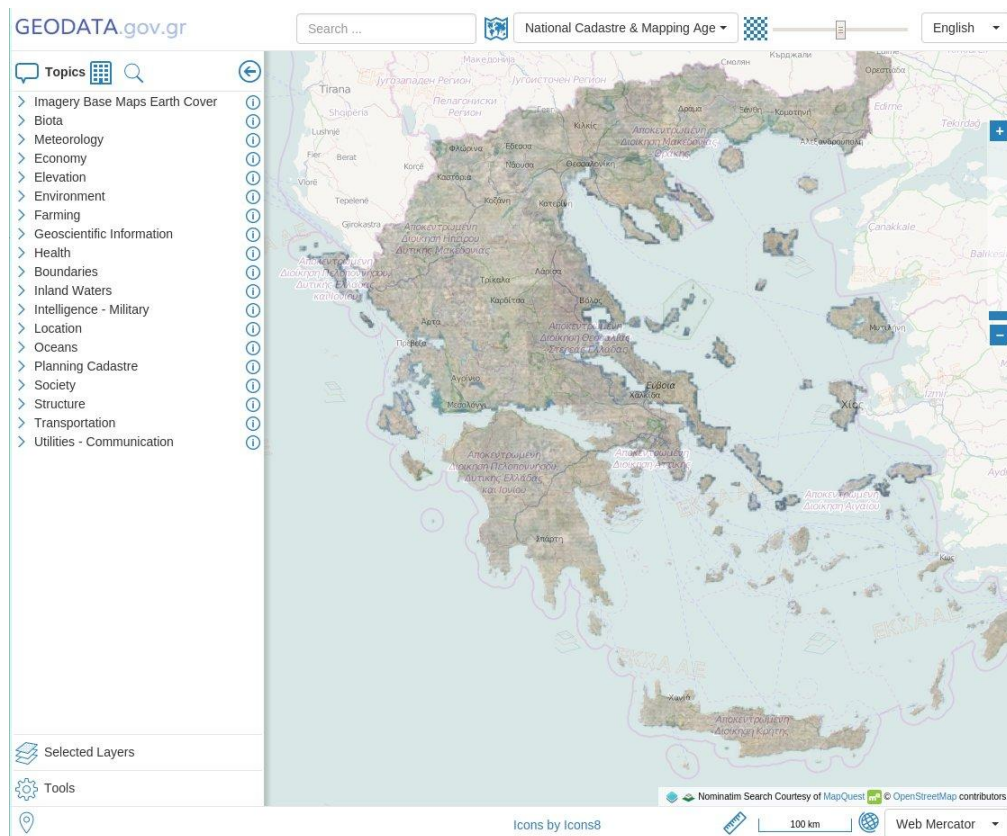


Figure 10: The MapClient CKAN plugin, adding a full map application directly into the CKAN application.

4.9 CKANEXT-MULTILINGUALITY

Ckanext-multilinguality is a CKAN plugin that allows tabular resource translation using recline. This plugin enables the addition of many supported languages in CKAN, and enables users of the catalog to provide translations for the dataset metadata as well for the actual data, if they are stored through the datastorer into the database. As such, spatial data in the attribute tables can be translated into several languages and then exported from the Data API. The user interface is presented in Figures 11-13. For the implementation of this plugin, the datastorer was modified to be able to



create *additional database tables per dataset* in order to hold the translated columns in a separate database table. This ensures that the plugin *will not break the default CKAN functionality*. This means the default language is stored in the CKAN main tables and the translations in custom database tables. The same strategy is applied to the metadata for the datasets; one language is marked as the *default* and the rest as translations. The UI lets the user to change display language. The translation process takes place one column at the time, while the user can exploit an automatic translation from an external service, from known dictionaries, or perform a manual translation.

Source code is available here:

<https://github.com/PublicaMundi/ckanext-multilinguality>

Data and Resources

aegean imports

[Manage](#)
[Data API](#)
[Download](#)

Resource language:

EN

Available languages:

EN FR DE ES

URL: <http://snf-665777.vm.okeanos.gmet.gr:5000/en/dataset/0b2ba5b8-9548-4191-ae38-2ff2c9b9eb04/resource/b3319fe6-c028-4c11-baa6-...>

From the dataset abstract

Περιέχονται τα όρια των οικοδομικών τετραγώνων (ΟΤ) του Δήμου Καλαμαριάς. Τα περιεχόμενα πολύγωνα έχουν μοναδική αρίθμηση ανά ΟΤ στη μορφή 'αριθμός τομέα-αριθμός ΟΤ'.

Source: Οικοδομικά τετράγωνα του Δήμου Καλαμαριάς

Grid 100 records << 0 - 100 >>

Search data ... Go » Filters

_id	code	type	material	cat_no	site	location	chronolo...	context	origin	chronolo...
1	1	Figurine	Faience	NMA no. ...	Perati	Mainland ...	LH IIIC1.	Tomb 30.	Egypt	18th-19th...
2	2	Figurine	Faience	NMA no. ...	Perati	Mainland ...	LH IIIC1.	Tomb 30.	Egypt	18th-19th...
3	3	Figurine	Faience	NMA no. ...	Perati	Mainland ...	LH IIIC1.	Tomb 30.	Egypt	18th-19th...
4	4	Figurines)	Faience	NMA no. ...	Perati	Mainland ...	LH IIIC1.	Tomb 30.	Egypt	18th-19th...
5	5	Figurine	Frit (Egyp...	NMA no 4...	Mycenae	Mainland ...	LH IIIB2.	Acropolis;...	Egypt	18th Dyn...
12	12	Figurine	Terracotta	lalyosos E...	lalyosos (...)	Rhodes	LH IIIA2.	New Tom...	Cyprus	LC II.
6	6	Figurine	Frit (Egyp...	Tiryns Ex...	Tiryns	Mainland ...	LH IIIA.	East of B...	Egypt	18th Dyn...

Figure 11: The Resources view in the dataset CKAN page (top) and the resource preview page with the multilingual support (bottom)





Figure 12: The user interface to trigger a dataset translation from within CKAN Resource management page. As can be seen the default language in this case is English.

Source: Οικοδομικά τετράγωνα του Δήμου Καλαμαριάς

_id	code	type	material	material-el	location	chronolo...	coi		
57	57	Bracelet ...	Bronze		aegean S...	12th cent...	Ca		
9	9	Figurine	Bronze		mos, Is...	LH IIIC (o...	He		
13	13	Statuette	Bronze		ands	LH IIIC (p...	Ar		
14	14	Statuette	Bronze		los, Isl...	LH IIIC.	Fiv		
15	15	Statuette	Bronze		Phylakopí...	Phylakopí	Melos, Isl...	LH IIIC (e...	St
16	16	Statuette	Bronze		NMA no. ...	Mycenae	Mainland ...	LH IIIB.	Hc
17	17	Statuette	Bronze		NMA no. ...	Tiryns	Mainland ...	LH III (po...	Cit

Figure 13: The recline user interface to translate a resource to another language. Here a new column is created (material-el) indicating the target language. Then the user can pick the method to translate.

4.10 CKANEXT-INTERLINKING

Ckanext-interlinking is a CKAN plugin that allows tabular resource interlinking using recline. Similarly to the previous plugin, the process of interlinking is to create a new database column (*in this case a temporary one*) and be able to provide the user with automated interlinking results. The interlinking method and the source data to interlink against, are selected from a Recline menu available to the user. The interlinking engine in this case is Lucene and the results are displayed according to their relevance (*interlinking score*). Then the user gets to decide if the result is acceptable and if not, which suggested alternative is the best for the result. Finally, the column gets interlinked and replaced in the CKAN datastore table.

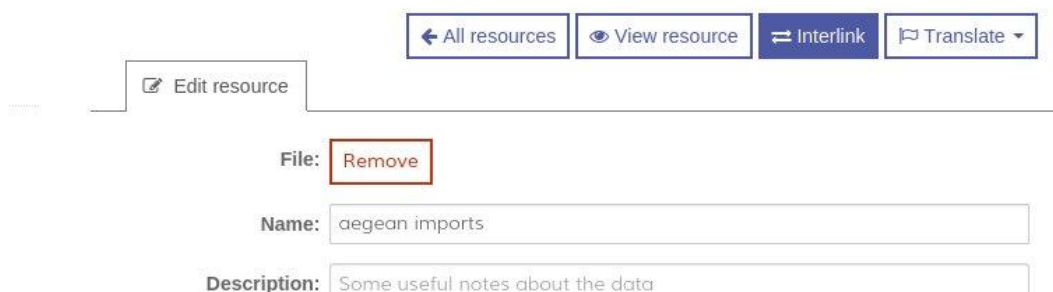


Figure 14: The user interface to trigger a dataset interlinking from within CKAN Resource management page.



calendar

View resource

Edit

Interlinking resource

URL: <http://snf-665777.vm.okeanos.grnet.gr:5000/en/dataset/cf48ac10-8c5e-4321-94fa-1c4eed54d00f/resource/365a16e9-1c61-4414-8078-...>

From the dataset abstract

Περιέχεται το γενικό πολεοδομικό σχέδιο για το δήμο Αγίου Στεφάνου όπως αυτό ορίζεται από το ΦΕΚ 561Δ/95. Η κλίμακα ψηφιοποίησης των δεδομένων είναι 1:10000.

Source: Γενικό πολεοδομικό σχέδιο δήμου Αγίου Στεφάνου2 ~1

Grid 588 records « 0 – 100 » Save Publish

_id	service_id	monday	tuesday	wednesday	thursday	friday	saturday	sunday	start
1	THΛEMA...	1	1	1	1	0	0	0	20
2	THΛEMA...	0	0	0	0	1	0	0	20
3	THΛEMA...	0	0	0	0	0	1	0	20
4	THΛEMA...	0	0	0	0	0	0	1	20
5	THΛEMA...	1	1	1	1	0	0	0	20

Figure 15: The recline user interface to interlink a resource based on another dataset.

Source code is available here:

<https://github.com/PublicaMundi/ckanext-interlinking>

