

OpenBudgets.eu: Fighting Corruption with Fiscal Transparency

Project Number: 645833

Start Date of Project: 01.05.2015

Duration: 30 months

Deliverable D3.3

Comparative Analysis Presentation and Creation Tools

Dissemination Level	Public
Due Date of Deliverable	Month 18
Actual Submission Date	21.02.2017
Work Package	WP3, Budget and Spending Data Visualisation and Exploration
Task	T 3.3 - Build comparative analysis presentation tools
Type	Demonstration
Approval Status	Draft
Version	0.3
Number of Pages	17
Filename	Deliverable 3.3 - Comparative Analysis Presentation and Creation Tools



Abstract: This document documents the comparative analysis presentation and creation features in the OpenBudgets.eu platform. An application has been developed, that lets users build aggregates and submit them to the available data mining services, on demand.

The information in this document reflects only the author's views and the European Community is not liable for any use that may be made of the information contained therein. The information in this document is provided "as is" without guarantee or warranty of any kind, express or implied, including but not limited to the fitness of the information for a particular purpose. The user thereof uses the information at his/ her sole risk and liability.

History

Version	Date	Reason	Revised by
0.1	11.02.2017	First contribution	Lazaros Ioannidis
0.2	19.02.2017	Indicators Use Case addition	Sotirios Karampatakis
0.3	21.02.2017	Final contribution - text structuring	Charalampos Bratsas

Author List

Organisation	Name	Contact Information
OKFGR	Charalampos Bratsas	cbratsas@okfn.gr
OKFGR	Lazaros Ioannidis	larjohn@okfn.gr
OKFGR	Sotirios Karampatakis	karampatakis@okfn.gr

Abbreviations and Acronyms

LOD	Linked Open Data
SPARQL	SPARQL Protocol And RDF Query Language
KPI	Key Performance Indicator
API	Application Programming Interface
OTA	Greek Local Authority Organizations
KAE	Greek OTA official Coded Budget Item Classification
RDF	Resource Description Framework
HTTP	Hyper-Text Transfer Protocol
SKOS	Simple Knowledge Organizations System
OS	OpenSpending
OBEU	OpenBudgets.eu

Table of Contents

[Introduction](#)

[Specification](#)

[Implementation](#)

1 Introduction

One of the most important features supported by the OpenBudgets.eu platform is the data mining and analysis capabilities. Apart from viewing and navigating fiscal dataset records, users can also build aggregates, view descriptive statistics and perform analysis tasks, like time series analysis.

In this deliverable, we have developed a software application that enables the end users of the platform to interactively construct visualizations that display the outputs of the various data mining and comparative analysis processes implemented in Work Package 2.

In addition, a fiscal indicators monitoring environment was implemented, tailored to the Greek datasets available to the project. The environment acts as a template for similar applications that can be made-to-order for local authorities and satisfy very specific monitoring requirements. This substitutes the original requirement of this deliverable for monitoring alerts, because the fiscal datasets the project deals with are not produced in a transactional manner, thus an indicator that summarizes a yearly performance is considered more suitable for the required functionality.

2 Specification

2.1 Indigo

The requirements for the software application were specified according to the original Deliverable 3.3 description and the actual characteristics of the rest of the platform, mainly the implementation of the data mining algorithms and the existing data access infrastructure, also utilized in Deliverable 3.2. Accordingly, the new application should:

1. be accessible through the Web
2. present data mining algorithms for each of the datasets in a specific store
3. allow the user to specify a subset of the dataset to submit for analysis
4. allow the user to specify an aggregation of the dataset to submit for analysis
5. offer comprehensible visualizations for each of the analysis processes
6. allow the exporting and sharing of the visualizations
7. offer a way to share the configuration of a task with other users
8. be compatible with the Babbage/Rudolf API

The fiscal indicators monitoring use case was designed to cover fiscal indicators found in greek proprietary state accounting deployments, while at the same time it is not targeted to a simple municipality but can combine and compare data from multiple datasets. In a future version, the data mining and analysis application should be able to guide the users to create such indicators combining fiscal and demographic data into shareable visualizations.

2.2 KPIs

2.2.1 What are KPIs

Key Performance Indicators, or KPIs in abbreviation, is a type of performance measurement, a set of quantifiable measures used to gauge an organization's or company's performance over time. These metrics can be used to determine an organization's progress in achieving it's strategic and operational goals and also to compare an organization's finances and performance against other organizations of the same scope¹.

On the case of Public Administration Organizations, such as Local City Councils, Municipalities, or other higher level authorities, KPIs can provide useful information about an organization's performance. We live in a period where fiscal transparency is a major issue. The overall Open Data movement has droved governments and local authorities around the world to publish their data, and especially fiscal data, in order to promote transparency. But these data, in raw format, can be difficult to handle by average people. KPIs are a compressed form of information. Huge volume of fiscal data can be summarized in a standardized way, offering a quick overview and better understanding and study over an authority's effectiveness and performance, as it is reflected by the published fiscal data.

2.2.2 Greek Municipalities Use Case

In the case of Greek Local authorities (OTA)², they are obligated by law, to publish their budgets in a standard form. A 4 level hierarchical classification exists for each level of OTA (2 levels, municipalities and Regional Units), describing different aspects of budgetary items called KAE (Coded Classification of Revenues/Expenditure). Local authorities publish their budgetary data with respect to these standards every year since 2010, and in some cases even earlier, classifying their data as extensions of KAE. Additionally, Greek Government has defined a number of KPIs, by which Local Authorities budgets are monitored for their effectiveness and performance on the context of their budgets. In general, KPIs on their simplest form are defined as divisions between different aggregates of budget data, or demographic data. We can group KPIs, used by the Greek Government in four groups:

- Per Population Income/Expense Indicators
- Correlation of Income - Expenses Indicators
- Human Resources Indicators
- Structural Indicators

¹<http://www.investopedia.com/terms/k/kpi.asp>

² http://www.ypes.gr/UserFiles/f0ff9297-f516-40ff-a70e-eca84e2ec9b9/egk08_29042013.pdf

2.2.2.1 Per Population Income/Expense Indicators

These are Indicators that are measuring different items of an organization's budget per capita. For instance Total expenses per Capita measures shows the total amount spent by the authority per capita. Investment per capita expresses the sum of investments that corresponds in each resident. It expresses that is to say the sum per resident that dedicates the Municipality on average for investments.

2.2.2.2 Correlation of Income - Expenses Indicators

Correlation Indicators are measuring the how a specific amount of a budget correlates to another. For instance, Subsidies Over Total Expenses is defined as the quotient subsidies over total expenses and denotes the percentage of the total costs of the municipality Total Expenses covered by subsidies. The higher the index value, the greater the grants participation to the expenses of a municipality and therefore the more dependent is the municipality of subsidies.

2.2.2.3 Human Resources Indicators

HR Indicators measure the percentage of participation of employment costs over total expenses or Operational Costs. The Employment Cost index in terms of costs is defined as the quotient of Employment cost over Total Expenses and expresses the cost of the total payroll staff in relation to the total expenditure. The higher the index, the smaller is the possibility of investment municipality and may be due, either to higher costs per employee, or a large number of workers. Therefore, the smaller the value of this index, the greater the possibility the municipality to invest.

2.2.2.4 Structural Indicators

These indicators are showing the percentage of participation of a budget item to the total. For instance Revenue structure shows the percentage of participation of each revenue item to the sum of all revenue. Moreover, Autonomy Indicator expresses the participation of regular revenue to the total revenue. As much as higher the index value, the greater autonomy has the municipality in budgetary matters and financial management as it relies in regularly revenues, which are predictable revenue streams. On the other hand, shrinking offs may indicate reduced investment activity, as well as investment grants are included in extraordinary income.

2.2.3 Current State

In total there are defined 28 KPIs. These KPIs are produced only for internal reasons and are not published centrally, unless individual application deployments by municipalities. An example is shown in Figure 5. A total of 15 indicators are shown in gauges or barchart. User has the option to select just the year and visualize the selected indicators for the specific municipality. Indicators can be exported in various image formats like png, pdf, svg and img but are not shareable by means of today's technology. There is not referred how these indicators are calculated, in which phase of the budget they refer and on which data it is

based. It refers only that the population is based on the latest census, thus making reproducibility difficult if not impossible.

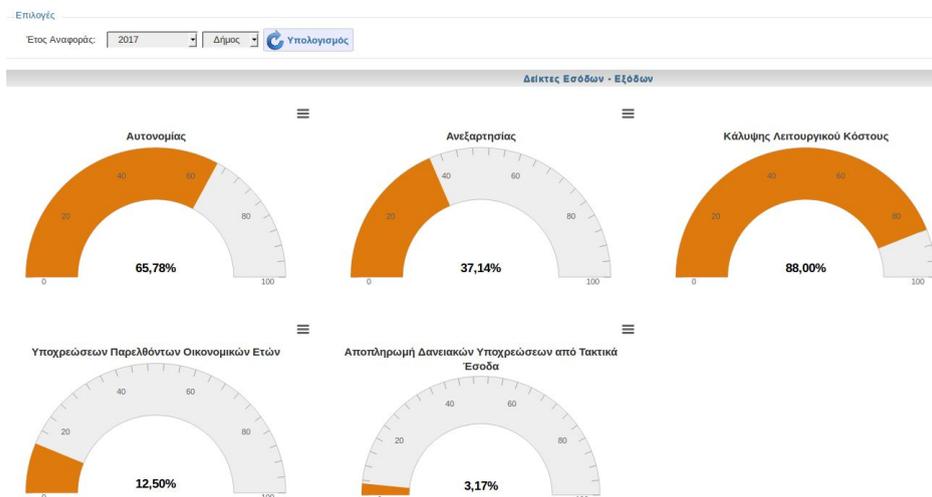


Figure 1: Example current implementation of City Dashboard for the Municipality of Veroia

3 Implementation

To satisfy the requirements, we have developed the indigo application and the fiscal indicators.

3.1 Indigo

Indigo is a web browser-based application built on the Angular 2 framework. It stands as the presentation layer between the data mining and analysis services and the end user.

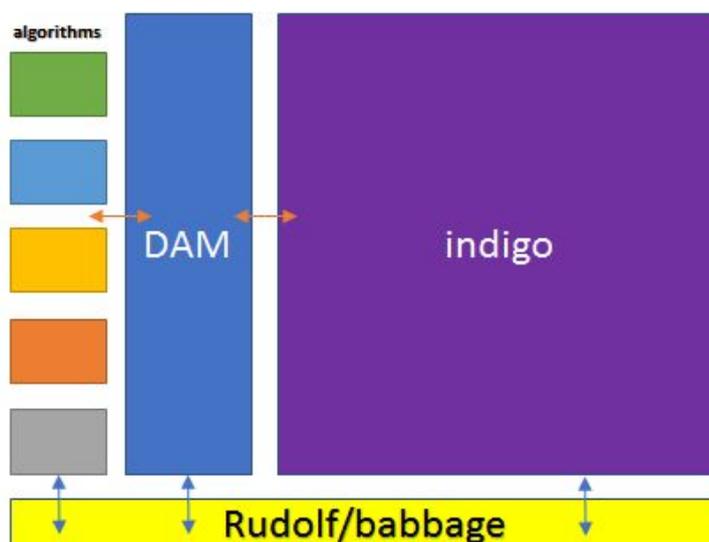


Figure 2: Indigo application integration in relation to the rest of the services

Actually, indigo communicates with two different API's. The first is the Data API, which should be a Babbage compatible API, like Rudolf, or an OpenSpending instance. This communication allows the indigo application to provide a user interface to the end users, where they can create the two types of data subsets they are familiar with, from OS Viewer: facts and aggregates. The two different subsets are valid for different algorithms, so indigo is responsible to present the suitable interface, given the algorithm selected.

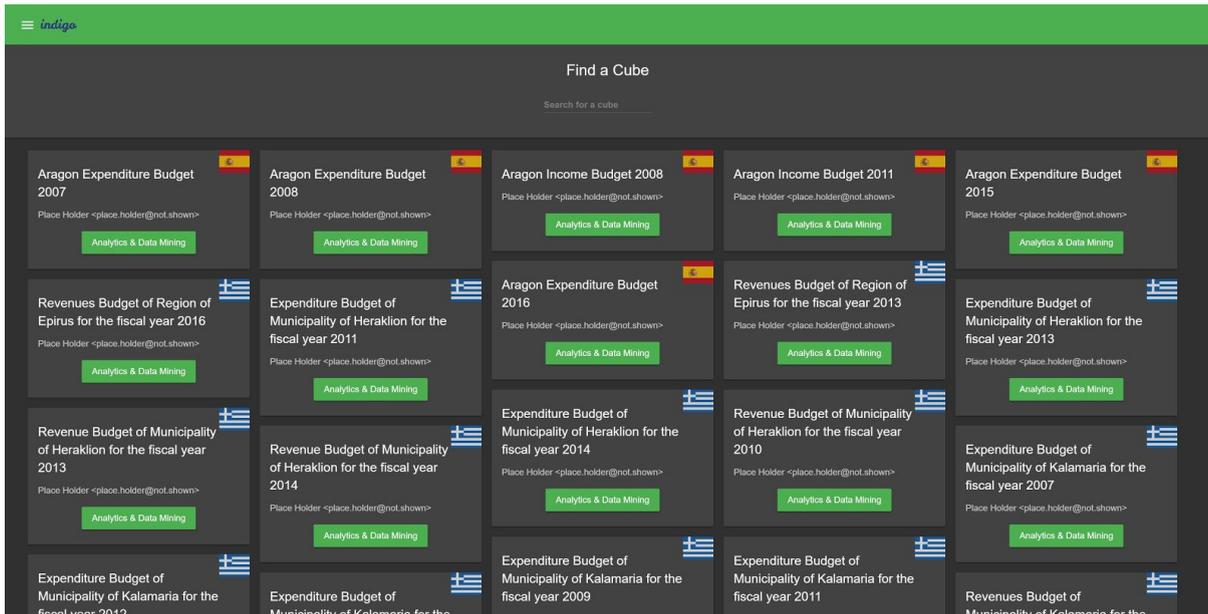


Figure 3: The users can select a dataset to apply the data mining analysis onto

Of course, the user interface is different for each dataset, depending on the dimensions and the measures it contains. This is accommodated by the model endpoint of the rudolf API, which is reflected in the user interface, providing the users with the respective options.

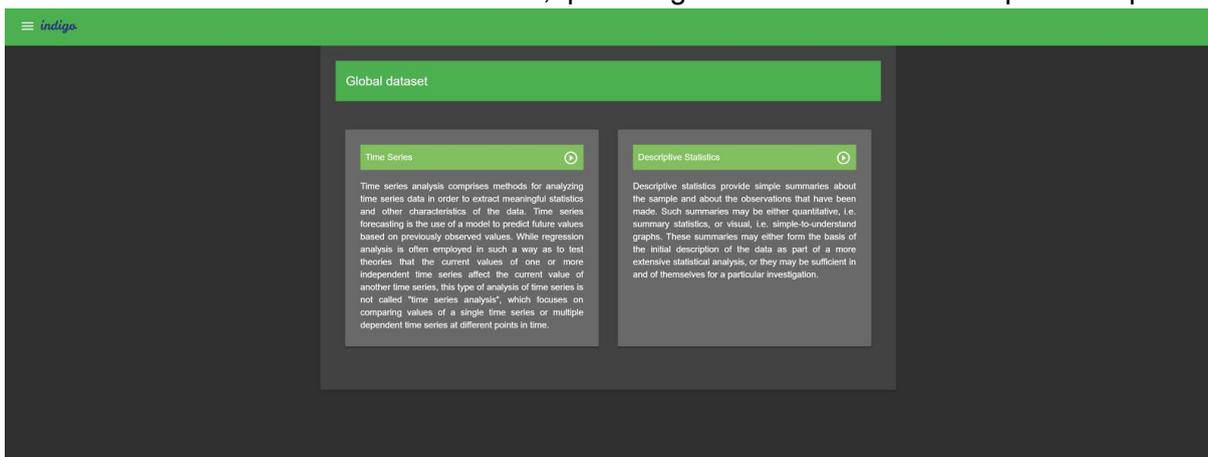


Figure 4: For a given dataset, a specific data analysis or mining algorithm can be selected

The facts or the aggregate data subsets are only one of the data mining algorithms inputs. Some algorithms might also need more input parameters in order to be executed, and the

indigo application provides the appropriate user interface for them. This is possible with the second API, the DAM. This API provides two endpoints: metadata and execution.

The metadata endpoint is responsible for providing indigo with a list of the available data mining algorithms for a specific subset of a dataset. The execution endpoint is the entry point for the actual execution of an algorithm on a specific subset of a dataset. Following this, indigo first asks for the available algorithms that can be executed onto the selected dataset. The user can then select one of the algorithms and then be presented with the facts/aggregate builder and more parameters, as described by the DAM metadata endpoint for this specific algorithm.

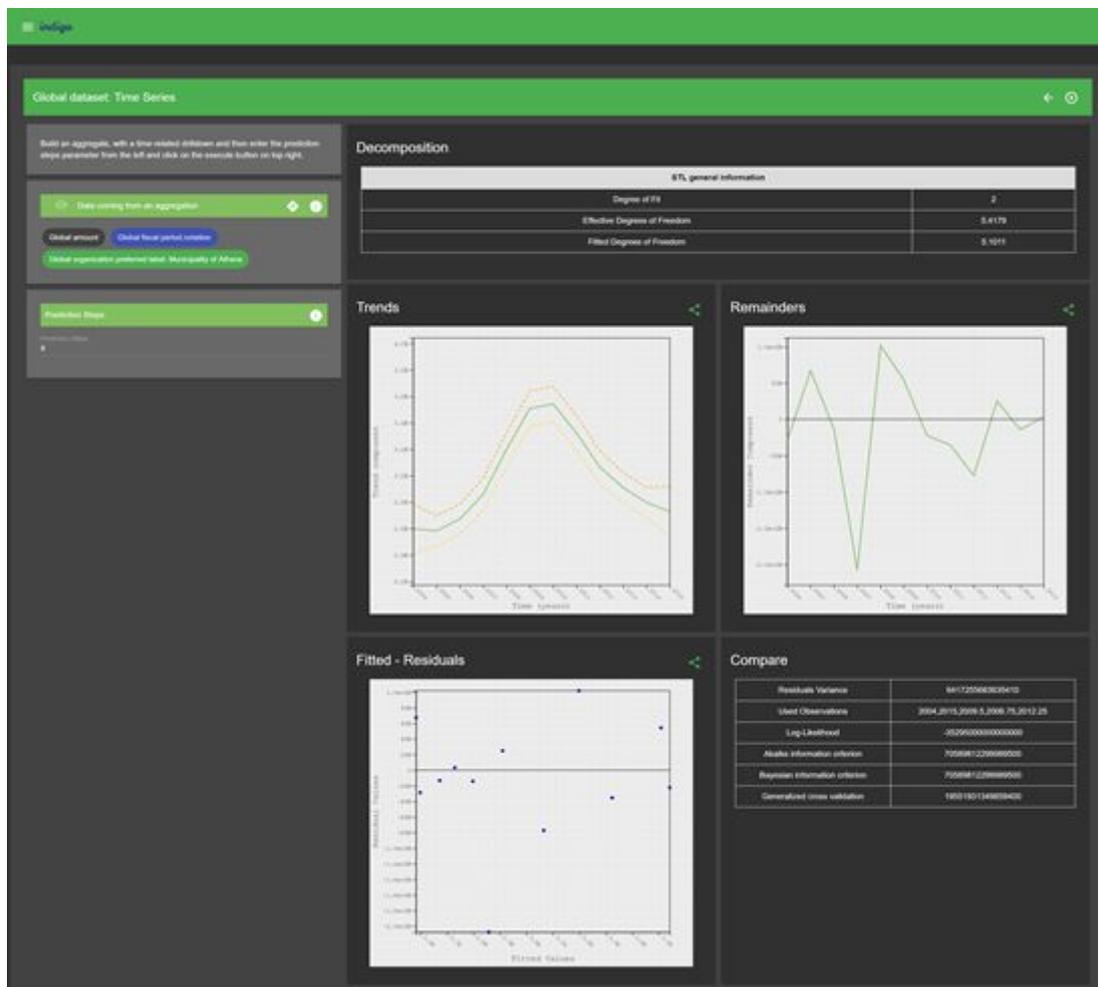


Figure 5: On the left, the user can define the inputs (included are the aggregate or facts request made to the API). The results appear on the right.

After the input parameters have been assigned values, they are sent to the DAM execution endpoint where they must be processed and forwarded to each data mining algorithm. The execution endpoint then returns the result of the execution to indigo, where the result is visualized. As the visualizations are built specifically for each algorithm, the addition of a new algorithm requires updating indigo as well, to be able to visualize its results.

Each execution of an algorithm on a specific dataset can be reproduced by visiting the same URL, which contains a serialization of the input parameters. This serialization can be used for demonstration purposes and also to waive the burden of building the analysis task from scratch. Pre-configured tasks can be an easy entry point for new users, so that they don't have to traverse the steep learning curve in order to make a couple of visualizations. Even visualizations themselves are shareable, so that a user can pick a single visualization and embed it to a social network.

3.2 Building a KPI platform integrating OpenBudgets.eu Infrastructure.

As derived from previous work completed on Work Package 1 and 2, we had transformed budgetary data from various Greek Municipalities and Regional Divisions following the OpenBudgets.eu Data Model and best practices. In particular we had converted budgetary data from 11 distinct municipalities and 2 regional divisions, for varying available years. This concluded in over 180 distinct datasets. The datasets are available on Github³ as dump files and through OpenBudgets.eu RDF Triplestore⁴. Greek Municipalities already conform to a standard formalization of their budgets as described above, so they share the same codelists for the same dimension components, allowing a level of comparison between them.

We developed an application to manage and visualize KPIs as they are defined by the Greek Government. The application consists of a frontend for the visualization of KPIs and comparative analysis, an Administration Panel to build and define KPIs and an PHP API for as a backend. The backend communicates with a remote SPARQL Endpoint, in order to fetch data and metadata for presentation on the frontend. The API has the ability to retrieve data from the LOD cloud (DBpedia or other), such as Municipality's population, label and geo-information (polygons, longitude and latitude) where needed.

The Admin can use the Administration Panel to define Groups, Aggregators and Indicators. Groups are groups of Indicators as defined above. Aggregators are classes that are used to evaluate aggregations over the datasets. The common case is that an admin define an aggregator title and description and then can choose a codelist, a skos:ConceptScheme, used as a codelist for a specific dimension of the datasets. Then, he/she can define which concepts are included on the aggregate and which are excluded. For instance including concepts with notations "1" and "2" and excluding "221" will result to include all concepts which are on the same branch with "1" and "2" and below, excluding those on "221" and below. This is better shown on Figure 6.

³ <https://github.com/openbudgets/datasets>

⁴ <http://data.openbudgets.eu/sparql>

Edit Aggregator 15

Code	<input type="text" value="total_income"/>
EN Title	<input type="text" value="Total Income"/>
EN Description	<input type="text" value="Total Income"/>
EL Title	<input type="text" value="Συνολικά Έσοδα"/>
EL Description	<input type="text" value="Συνολικά Έσοδα"/>
Included	<input type="text" value="0.1.2.3.4.5"/>
Excluded	<input type="text" value=""/>
Codelist	<input type="text" value="http://data.openbudgets.eu/resource/codelist/k"/>

Figure 6: Defining an aggregator

Data from the LOD cloud can be defined also as aggregators. User has to define the property path that will be used and the endpoint to query. Using HTTP calls was an option but result returned may vary (invalid serializations, wrong redirects, long files etc) and in some cases there is not such an option. So we chose to create SPARQL queries to the remote endpoints.

Indicators can be defined as well from the Admin Panel. Admin has to define indicator title, description, the group the indicator belongs. The Enabled option publishes the indicator on the main dashboard. The Reversed option defines the direction of an indicator. There are indicators where a high value has a positive meaning whereas the opposite in others. For instance, having a high value in Investments per Resident is considered positive. On contrast, Taxes and Fees per Resident high value has a negative meaning. Next, Admin has to define the type of the Indicator between “Percent” and “Numeric”, as a helper for the visualizations. The first will render a gauge on dashboard, whilst the latter will render a number suffixed with the currency.

Figure 7: Indicator Builder Form

We defined all of the indicators described by the Greek Government. Users have access to a main dashboard, showing only published Indicators. Admin can define the default or the only Municipality, budget phase and year shown. Users can select parameters from a minimal form and update the form.

Figure 8: Dashboard Form

Each indicator is embeddable and shareable through social media using the according menu buttons. Each Indicator is coupled with it’s definition as a helper for a better understanding of what the specific indicator express.



Figure 9: (from left to right) a) gauge view , b) numeric view, c) Indicator description

An interesting feature is the two action buttons on the bottom left corner. The yearly change action button leads to a view where the specific indicator for the selected municipality and phase is shown on a bar chart showing the indicators value for each year available. User can select different type of visualization from three types of charts, barchart (default), line chart and scatter plot. This allows the user to monitor the evolution of the indicator through the pass of time for the available years. Users can add extra datasets on the chart by changing parameters from the form above. Users can select different organization, budget phase or Indicator to display on the same visualization.



Figure 10: Yearly change and comparison between municipalities. The extremely low values for the years of 2004 and 2005 indicate missing data or incomplete datasets.

The second action button, “Compare” by default leads to a page where a barchart is presented, having as fixed dimensions the year, the budget phase and the indicator. The free dimension is organizations so user can compare the indicator value against other available organizations on the Endpoint. The behaviour is fully configurable, allowing the user to change fixed dimensions and their values. Thus, users can have various comparative visualizations.

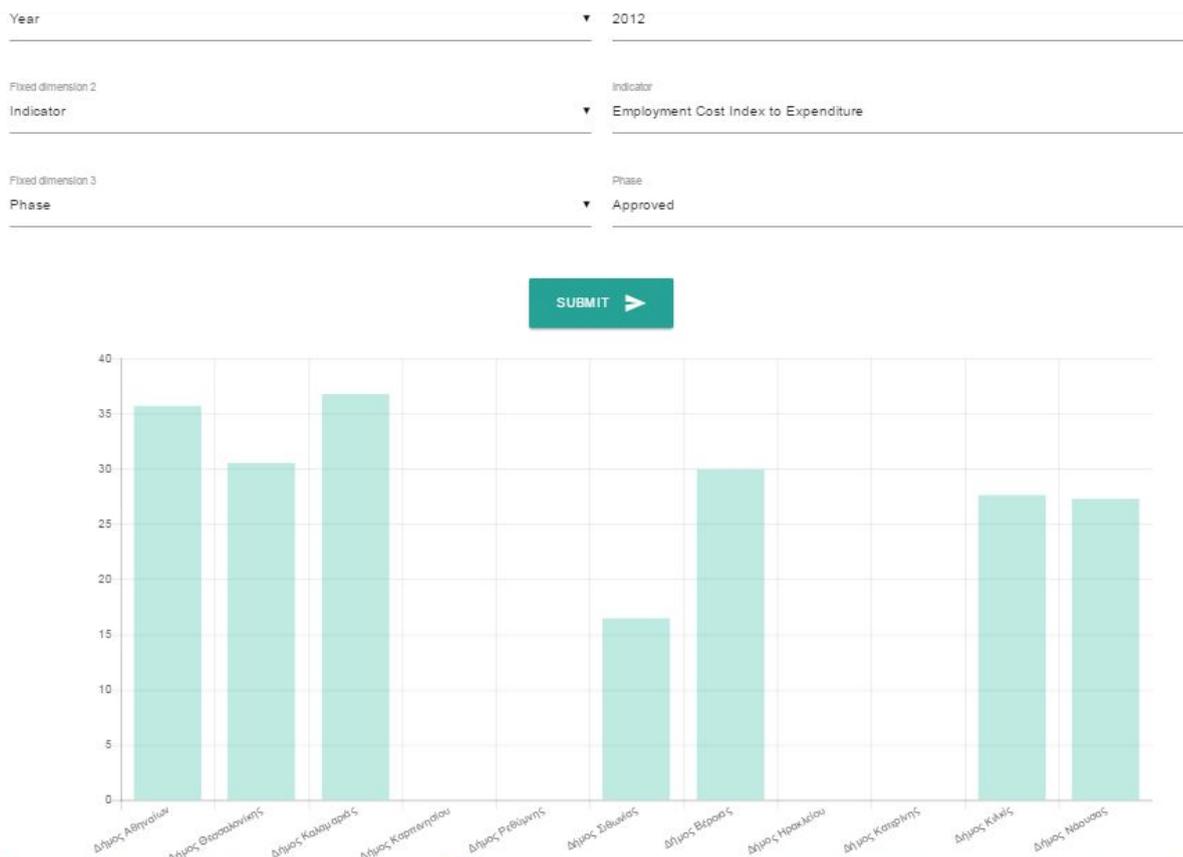


Figure 11: Comparison View. User can select fixed dimensions on the form on the top. The remaining dimension will be the free dimension.

Another interesting view is the Radar Dashboard. Indicators are grouped by the group they belong. Groups with 4 members and below are shown as barcharts, whilst the rest as radar charts. This restriction is configurable through the Admin Panel. In the case of Greek Municipalities there are four groups defined. Two are shown as radar charts and two as barcharts. This view allows for a quick overview of the indicators. User can update this view as well, including additional organizations, phases and years where available. All visualizations are embeddable and shareable allowing dissemination through social media and other channels. Users can download the data used for the calculations in different formats (CSV, JSON, RDF) or forward to the according dataset on the OpenSpending Viewer to explore different aspects of the dataset and get benefit by the services offered by the OpenBudgets.eu platform.

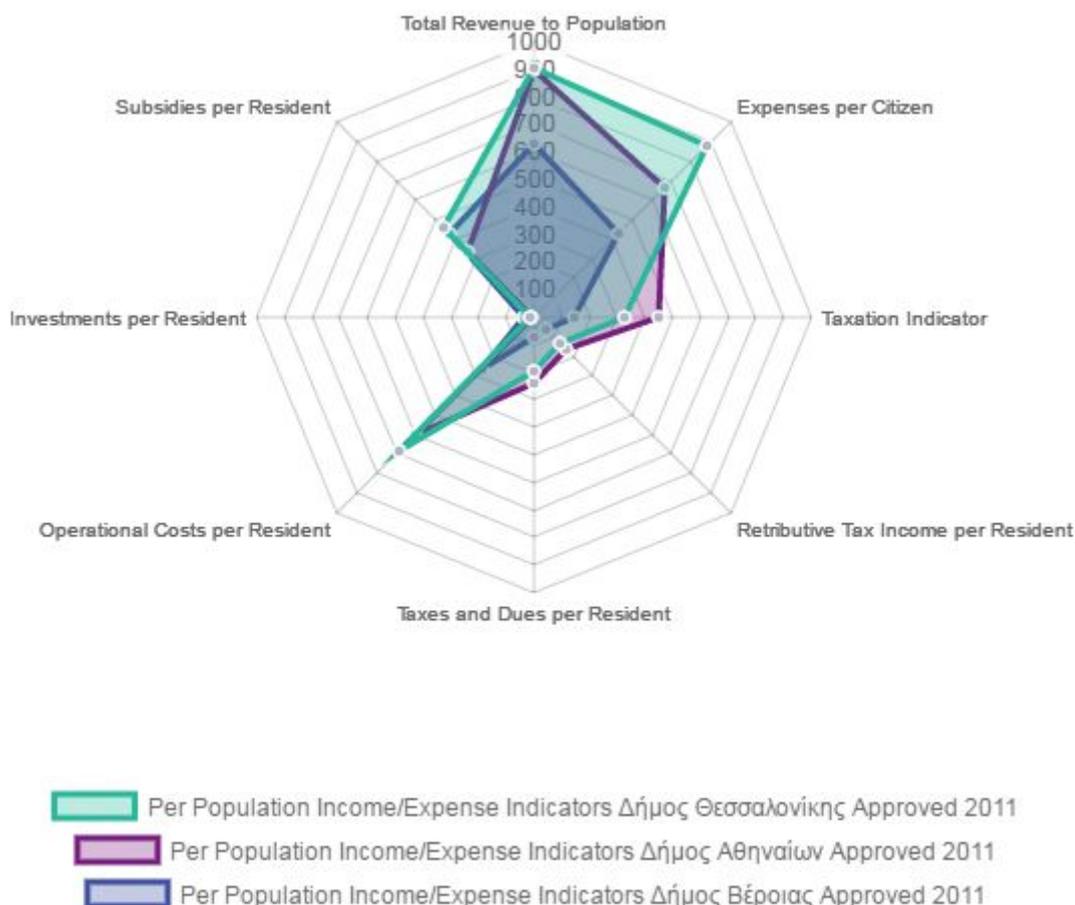


Figure 12: Radar chart showing the Per population group for three different municipalities for the same year and budget phase.

3.3 Extend to support municipalities of other countries

A critical point on the implementation of this application is the use of common codelists amongst the different Data Cubes. All use common dimension codelists for four cube dimensions. The `obeu-dimension:fiscalYear`, `obeu-dimension:administrativeClassification`, `obeu-dimension:economicClassification` and `obeu-dimension:budgetPhase`. Fiscal year dimension and Budget Phase are used as filters to select specific slices of the datasets. The fact that `obeu-dimension:economicClassification` uses common hierarchical codelist (KAE codelist), described using the SKOS vocabulary as reported on D1.7, makes possible the comparison. Aggregators are built using members of this codelist and taking advantage of the hierarchical structure to summarize individual observations. In order to extend this functionality to support comparison between municipalities in different countries, a common codelist should be used, or mappings between them. A mapping of at least the top and the first children would allow aggregations across different municipalities. Another solution would be to extend aggregators to support multiple codelists. Towards this direction, we will describe KPIs and it's elements as RDF entities.

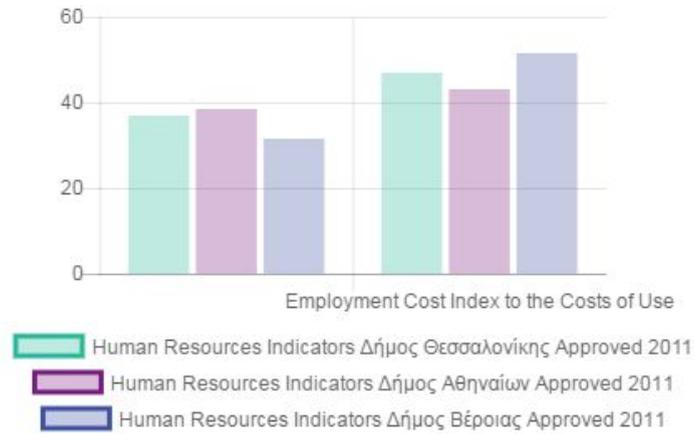


Figure 13: Bar Chart showing the HR Indicators Group for three different municipalities.

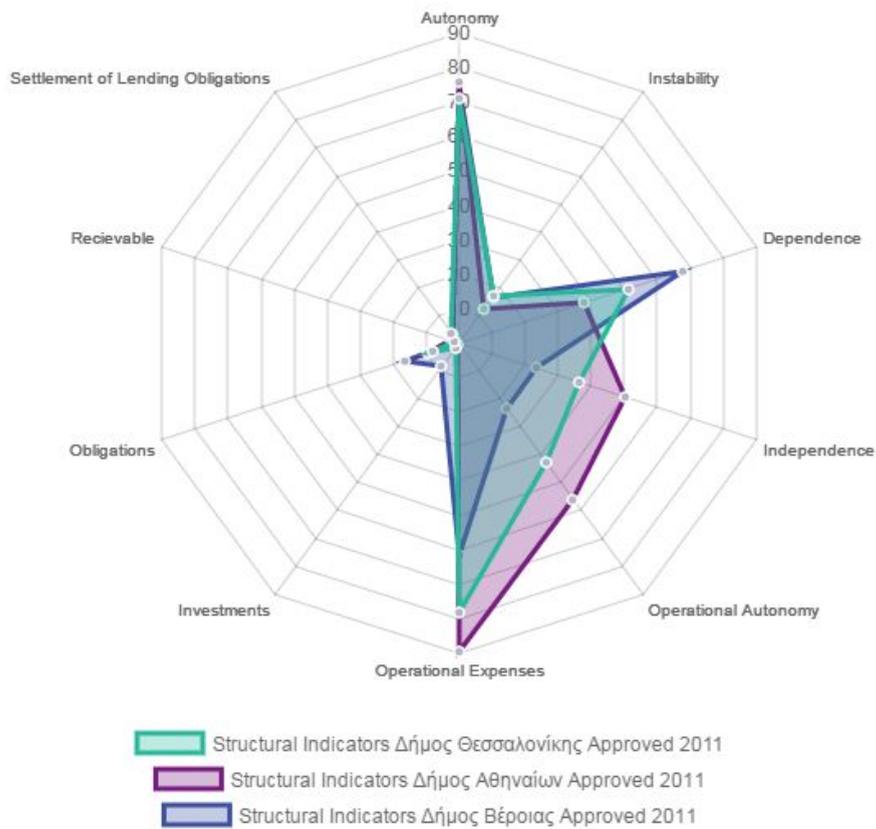


Figure 14: Radar chart showing Structural Indicators group for the same year and phase of three different municipalities