Chapter 1
Linked Open Data for Public Procurement

Vojtěch Svátek, Jindřich Mynarz, Krzysztof Węcel, Jakub Klímek, Tomáš Knap, Martin Nečaský

Abstract Public procurement is an area that could largely benefit from linked open data technology. The respective use case of the LOD2 project covered several aspects of applying linked data on public contracts: ontological modeling of relevant concepts (Public Contracts Ontology), data extraction from existing semi-structured and structured sources, support for matchmaking the demand and supply on the procurement market, and aggregate analytics. The last two, end-user oriented, functionalities are framed by a specifically designed (prototype) web application.

1.1 Public Procurement Domain

Among the various types of information produced by governmental institutions as open data, as obliged by the law, are descriptions of *public contracts*, both at the level of *requests for tenders* (RFT, also 'calls for bids' or the like)—open invitations of suppliers to respond to a defined need (usually involving precise parameters of the

Vojtěch Svátek
University of Economics, Prague, Czech Republic, e-mail: Svatek@vse.cz

Jindřich Mynarz
University of Economics, Prague, Czech Republic, e-mail: jindrich.mynarz@vse.cz

Krzysztof Węcel
I2G, Poland, e-mail: krzysztof.wecel@i2g.pl

Jakub Klímek
Czech Technical University & University of Economics, Prague, Czech Republic, e-mail: klimek@ksi.mff.cuni.cz

Tomáš Knap
Charles University & University of Economics, Prague, Czech Republic, e-mail: knap@ksi.mff.cuni.cz

Martin Nečaský
Charles University, Prague, Czech Republic, e-mail: necasky@ksi.mff.cuni.cz

required product/s or service/s)—and at the level of *awarded contract* (revealing the identity of the contractor and the final price). The whole process is typically denoted as public/government *procurement*. The domain of public procurement forms a fundamental part of modern economies, as it typically accounts for tens of percents of gross domestic product.[1] Consequently, due to the volume of spending flows in public procurement it is a domain where innovation can have significant impact. Open disclosure of public procurement data also improves the transparency of spending in the public sector.[2]

An interesting aspect of public contracts from the point of view of the semantic web is the fact that they unify two different spheres: that of *public* needs and that of *commercial* offers. They thus represent an ideal meeting place for data models, methodologies and information sources that have been (often) independently designed within the two sectors. Furthermore, the complex life cycle of public contracts gives ample space for applying diverse methods of data analytics, ranging from simple aggregate statistics to analyses over complex alignments of individual items. On the other hand, using linked data technology is beneficial for the public contract area since it allows, among other, to increase interoperability across various formats and applications, and even across human language barriers, since linked data identifiers and vocabularies are language-independent.

As three major views of the e-procurement domain we can see those of *domain concepts*, *data* and *user scenarios*. Plausible and comprehensive *conceptualization* of the domain is a prerequisite for correct design of computerized support as well as for ensuring data interoperability. Management of the large amounts of *data* produced in the procurement domain has to take into account its varying provenance and possibility of duplicities and random errors. Finally, the activities of *users*, i.e, both contract authorities and bidders/suppliers, along the different phases of the public contract lifecycle, have to be distinguished. Linked data technology provides a rich inventory of tools and techniques supporting these views. The last, user-oriented view is least specific of the three; typically, the user front-end does not differ much from other types of (web-based) applications, except that some functionality, such as autocompletion of user input, exhibits online integration to external linked data repositories.

Public procurement domain has already been addressed by projects stemming from the semantic web field. The most notable ones are probably LOTED[3] and MOLDEAS [1]. LOTED focused on extraction of data from a single procurement source, simple statistical aggregations over a SPARQL endpoint and, most recently, legal ontology modeling [4]. MOLDEAS, in turn, primarily addressed the matchmaking task, using sophisticated computational techniques such as spreading activation [2] and RDFized classifications. However, the effort undertaken in the LOD2 project is unique by systematically addressing many phases of procurement linked data processing (from domain modeling through multi-way data extraction, trans-

[1] For example, as of 2010 it makes up for 17.3 % of the EU's GDP [7].

[2] See, e.g., `http://stopsecretcontracts.org/`

[3] `http://loted.eu/`

formation and interlinking, to matchmaking and analytics) as well as both EU-level and national sources with diverse structure.

The chapter structure follows the above views of public procurement. First, the *Public Contract Ontology* (PCO) is presented, as a backbone of the subsequent efforts. Then we review the original public contract data sources that have been addressed in our project, and describe the process of their *extraction*, *cleaning* and *linking*. Finally, the end user's view, in different business scenarios, supported by a *Public Contract Filing Application* (PCFA for short) is presented. It is further divided into the *matchmaking* functionality and the *analytic* functionality (the full integration of the latter only being in progress at the time of writing the chapter).

1.2 Public Contracts Ontology

The ontology developed within the use case covers information related to public contracts that is published by contracting authorities during the procurement process. We built the ontology on the basis of analysis of the existing public procurement portals (especially TED[4] as the European one, but also national ones) and information published by contracting authorities on these portals. We did not consider all information but primarily the information relevant for matching public contracts with potential suppliers. Therefore, for the most part we consider the information that is produced in the tendering phase (description of the public contract, received tenders and the eventually accepted tender). From the evaluation phase we consider the actual final price for the contract, as its modeling is identical to that of estimated (in the contract notice) as well as agreed price; no further complexity is thus added to the ontology by including it.

1.2.1 Ontologies Reused by the PCO

Reusing existing, established ontologies when building one's own ontology is crucial for achieving interoperability on the semantic web, since applications capable of working with the original ontologies can then also process the reused elements (and even their derivatives, such as subclasses and subproperties) in the new ontology as well. The PCO reuses the following models:

- GoodRelations Ontology[5] (`gr` prefix) – to model organizations and price specifications
- VCard Ontology[6] (`vcard` prefix) – to express contact information

[4] http://ted.europa.eu/TED/

[5] http://purl.org/goodrelations/v1#

[6] http://www.w3.org/2006/vcard/ns#

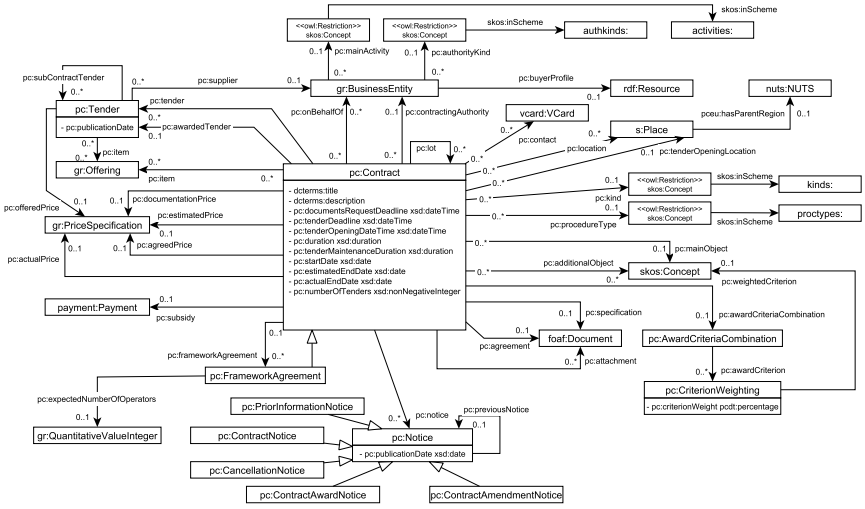

Fig. 1.1 Public Contracts Ontology – UML Class Diagram

- Payments Ontology[7] (`payment` prefix) – to express subsidies
- Dublin Core[8] (`dcterms` prefix) – to express descriptive metadata (e.g., title, description)
- Simple Knowledge Organization System (SKOS)[9] (`skos` prefix) – to express code lists and classifications
- Friend-of-a-friend Ontology (FOAF)[10] (`foaf` prefix) – to express agents, especially persons and relationships between them
- schema.org[11] (`s` prefix) – to express locations and other generic kinds of entities
- Asset Description Metadata Schema (ADMS)[12] (`adms` prefix) – to express identifiers.

1.2.2 Core Concepts of the PCO

Figure 1.1 depicts the ontology in the form of a UML class diagram. The core concept of the ontology is that of *public contract* represented by the class `pc:Contract`. We understand a public contract as a single object that groups pieces of information

[7] http://reference.data.gov.uk/def/payment#
[8] http://purl.org/dc/terms/
[9] http://www.w3.org/2004/02/skos/core#
[10] http://xmlns.com/foaf/0.1/
[11] http://schema.org/
[12] http://www.w3.org/ns/adms#

related to the contract. These pieces of information gradually arise during the public procurement process. They are published by contracting authorities on various public procurement portals in the form of different kinds of notification documents, e.g., *call for tenders* (sometimes also called *contract notice*), *contract award notice*, *contract cancellation notice*, or the like. Another important concept of the ontology are *business entities*, i.e., in this context, contracting authorities and suppliers. Business entities are not represented via a new class in the ontology; we rather reuse the class `gr:BusinessEntity` from the GoodRelations ontology.

1.2.2.1 Tendering Phase Modeling

In this phase the contracting authority publishes initial information about a public contract. This includes basic information, e.g., contract title, description, and the reference number assigned to the contract by the authority (which is usually unique only in the scope of the authority itself). If the contract is too large or too complex, the authority can split it into two or more sub-contracts, which are called *lots*. Each lot is a contract on its own but it is a part of its parent contract. In the ontology, lots are represented using the class `pc:Contract`. A lot is associated with its superior contract via the property `pc:lot`.

The authority also publishes basic information about itself, e.g., its legal name, official number or contact information. An important piece of information is the so-called *buyer profile*, which is a web page where the contracting authority publishes information about its public contracts.

Moreover, the authority publishes various requirements and restrictions on the contract. The restrictions include the specification of the kind and objective areas of the contract (Supplies, Works and Services), deadline for tenders, time to realize the contract (in a form of estimated end date or required duration in days or months), estimated price and non-structured specification documents. The objective areas restrict potential suppliers to only those operating in the objective areas. The authority also specifies the procedure by which an awarded tender will be selected from the received tenders. We consider the following procedure types in the core, which are primarily defined by the EU legislation but can be applied world-wide: Open, Restricted, Accelerated Restricted, Negotiated, Accelerated Negotiated, and Competitive Dialogue.

The requirements include two kinds of information. First, the authority specifies the items (i.e., products or services) that constitute the contract. The ontology reuses the class `gr:Offering` to represent items. Basically, the items are characterized by their name, description and price, but other kinds of characteristics can be used as well, which we however do not cover in the domain model (e.g., references to various product or service classification schemes). Second, it specifies a combination of award criteria. The class `pc:AwardCriteriaCombination` is the representation of this feature in the ontology. For each award criterion, expressed in the ontology using class `pc:WeightedCriterion`, a textual description (or, we can say, name) and weight percentage of the criterion is specified. Usually, a spe-

cific combination is distinguished. For instance, it may specify that tenders are only compared on the basis of the price offered and that the tender offering the lowest price has to be selected.

After the authority receives the tenders, it publishes either the number of tenders or details of each particular tender received. For each tender, details about the tendering supplier and the offered price are published. A tender may also comprise information about particular items (similarly to contract items, i.e. products or services) offered. Tenders are represented in the ontology using a class called `pc:Tender`. Then, according to the award criteria, the authority selects and awards the best tender. In this stage, the authority publishes the date of the award and marks the selected tender as the awarded tender.

During the tendering phase, the contract can be cancelled by the authority. If so, the authority publishes the date of cancellation.

1.2.2.2 Pre-realization, Realization and Evaluation Phase Modeling

In the pre-realization phase, the contracting authority signs an agreement with the supplier and publishes the agreement on the Web. The agreement is a non-structured text document. The ontology reuses the class `foaf:Document` to represent unstructured textual documents. We only consider one particular structured information published – the price agreed by both the authority and supplier. The agreed price should be the same as the price offered in the awarded tender but it can differ in some specific cases.

After the realization, the authority evaluates how the supplier fulfilled the requirements. This includes various sub-processes that we do not cover in our analysis. We are only interested in the actual end date and the actual final price of the contract. Moreover, the authority could cover the price of the contract or its part by subsidy from an external source (e.g., EU structural funds). Subsidies are usually provided in the form of one or more payments to the authority.

1.3 Procurement Data Extraction and Pre-processing

Although procurement data are published in a certain form in most countries in the world, we focused on three groups of sources:

1. The European TED (Tenders Electronic Daily) portal,[13] which contains data from a number of countries (thus allowing for cross-country comparisons, as shown in [8]), although only a subset of these (typically for contracts above a certain price level).
2. The Czech and Polish procurement data portals; the lead partners in the procurement linked data activity of the LOD2 project are based in these two coun-

[13] `http://ted.europa.eu/TED/`

tries, and therefore have both good contacts to the national publishing agencies, knowledge of the local regulations, and fluency in the languages in which the unstructured part of the data is written.
3. US and UK procurement data portals, as these are the countries where the open government publishing campaign started first and therefore even the procurement data sources are likely to be found sufficiently rich and well curated.

Regarding the source format of the data, the TED and Czech data were initially only available as HTML, and only at a later phase became also published in XML. In contrast, the Polish, US and UK data have been available in XML from the beginning. Data extraction (and RDFization) methods for both formats have therefore been investigated.

1.3.1 Data Extraction from HTML

TED and the Czech national portal ISZVUS (later renamed to Public Procurement Bulletin) had been the prime targets in the initial phase of the project. At that time, only the HTML pages were available for these resources. In Fig. 1.3.1, we can see two HTML fragments with information about one lot; we can demonstrate different flavors of HTML-based data extraction on them. Both contain red-labelled sections numbered 1 to 4 (the related properties are in Table 1.1).

The left side of Figure 1.3.1 depicts a fragment of a TED HTML document. The data is stored in `div` elements combined with additional textual information. Section 1 of the document contains combined information about the lot ID and the lot name, so it is necessary to split these properties. Section 2 only contains one property, with a textual label that has to be removed. In the sections 3 and 4 the fields are separated by `br` tags combined with additional labels.

In contrast, the data in ISVZUS is strictly structured using `input` elements with unique `id` attributes (see the right side of Figure 1.3.1), which allows to access the data fields without any additional transformation.

#	PCO property	#	PCO property
1	`dc:title` + `adms:identifier`	3	`pc:supplier`
2	`pc:numberOfTenders`	4	`pc:offeredPrice`

Table 1.1 PCO property mapping to HTML fragments

Technologically, the extraction was based on CSS selectors, and (where the CSS selectors did not suffice) pseudo-selectors[14] allowing to search for elements containing a defined substring. In some cases the element content had to be modified or shortened, which led us to applying regular expressions.

[14] Provided by the JSoup library `http://jsoup.org/`.

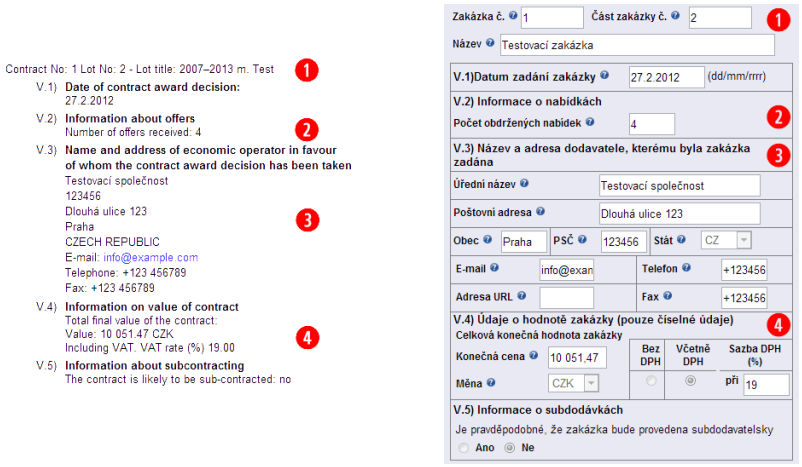

Fig. 1.2 TED fragment (left) and ISVZUS fragment (right)

The HTML-based ETL activities for both resources were later suspended when the future availability of full data in XML (rather than mere HTML) was announced. The processing was resumed in Spring 2014 based on XML dumps (and, for the Czech data, also an XML-based SOAP API), which are more reliable than data obtained via information extraction from semi-structured text embedded in HTML.

1.3.2 Data Extraction from Structured Formats

The extraction from strucured formats, namely, XML and CSV, started at different phases of the project and was carried out by different groups, therefore the used technology slightly varied. The first XML data addressed was the (British) Contracts Finder,[15] for which a standalone XSLT script for transforming all fields to RDF triples was developed in early 2012. Later, however, the main focus was on the European (TED), Czech, Polish, and also U.S. data (to have an extra-European source for comparison).

1.3.2.1 TED Data

In March 2014 the Publications Office of the EU opened access to the data from TED and ceased to charge licensing fees for data access. Current public notices for the past month are available to download for registered users of the TED portal and also via an FTP server. Archived notices dating back to 2010 can be obtained in

[15] http://contractsfinder.businesslink.gov.uk

monthly data exports. Data is published in 3 formats, including a plain-text one and 2 XML formats.

We created an XSL transformation script to convert the TED data into RDF. Using this XSLT script we performed a bulk extraction of the TED archival data via the Valiant tool[16]) from the LOD2 Stack. In parallel, using the UnifiedViews ETL framework,[17] we set up an automatic, continuously running extraction of the increments in TED data. In the further treatment of the extracted RDF data we focused on deduplication and fusion of business entities participating in the EU public procurement market, in order to provide a more integrated view on the dataset.

1.3.2.2 Czech Data

We developed an extractor data processing unit[18] for the UnifiedViews ETL framework, which is capable of incremental extraction of data from the *Czech public procurement register*[19] using its SOAP API. During the time we discussed the possibility of publishing raw open data in bulk with the company running the register. As a result of these discussions we were provided with an XML dump of historical data from the register to be used for research purposes. Combining the historical data dump with the access to current data via the SOAP API we were able to reconstruct the complete dataset of public contracts from the registry converted to RDF.

The second source of Czech public procurement data that we processed was a set of *profile feeds* of individual contracting authorities. As per the amendments in the Czech public procurement law, public sector bodies involved in public procurement are required to publish their own XML feed of data about public contracts they issue, including both public notices and award information. The set of public contracts that are published on profile feeds is a superset of what is available via the central Czech public procurement registry because the feeds also cover some lower price public contracts, which are not required to be published in the central register. The content of these feeds mostly mirrors the content of the central register, although for individual public contracts it is less comprehensive. While the data from the register is richer and more descriptive, the profile feeds contain information about *unsuccessful tenders*, which is missing from the register that only reveal information about winning tenders. We deem having data about both successful and unsuccessful tenders as vital in several analytical tasks over public procurement data, which is one of the reasons why we have invested effort into acquiring the data from feeds of contracting authorities. Since early autumn 2013 we have been scraping an HTML list of URLs of profile feeds and periodically convert each feed's XML into RDF using an ETL pipeline developed using the UnifiedViews framework. By using code-based URIs the data is linked to several external datasets. Company identifiers connect it

[16] https://github.com/bertvannuffelen/valiant

[17] https://github.com/UnifiedViews/Core

[18] https://github.com/opendatacz/VVZ_extractor

[19] http://vestnikverejnychzakazek.cz/

to the Czech business register[20] that we also periodically convert to RDF. Common Procurement Vocabulary (CPV) codes[21] link it to the RDF version of CPV that we produced.

1.3.2.3 Polish Data

Public procurement data is published by The Public Procurement Office (*Urząd Zamówień Publicznych*[22]) in the Public Procurement Bulletin (*Biuletyn Zamówień Publicznych* – BZP[23]).

There are several means to access the data: browsing the BZP portal, subscription mechanism with some restricted number of criteria, and the download of XML files, which we employed in the RDFization. The structure of XML is basically flat: even though some attributes can be grouped that are put on the same level. This has implications for the parsing and conversion mechanisms. On the one hand, no subset of XML data can be selected for further processing. On the other hand, the extraction expressions as well as XML paths are shorter. Conversion of XML files containing notices about public contracts has been carried out by means of Tripliser.[24] The RDFization had to overcome some issues in the XML structure, such as the use of consecutive numbers for elements describing the individual suppliers (in Polish ''wykonawca') awarded the different lots of a contract: `wykonawca_0`, `wykonawca_1`, `wykonawca_2` and so on. We also had to write our own extension functions for Tripliser allowing us to generate new identifiers for addresses, as data structures, from their parts: locality, postal code and street.

Automatic *linking*, using Silk[25] as one of the LOD2 stack[26] tools, was carried out for the problem of mapping the contact information of a given contracting authority or supplier to a classification of Polish territorial units called TERYT.[27]

1.3.2.4 U.S. Data

The dataset was created by combining data from two principal sources, which provide complementary views of procurement data. The two sources in question

---

[20] `http://www.czso.cz/eng/redakce.nsf/i/business_register`

[21] By its definition from `http://simap.europa.eu/codes-and-nomenclatures/codes-cpv/codes-cpv_en.htm`, "CPV establishes a single classification system for public procurement aimed at standardising the references used by contracting authorities and entities to describe the subject of procurement contracts."

[22] `http://uzp.gov.pl`

[23] `http://uzp.gov.pl/BZP/`

[24] A Java library and command-line tool for creating triple graphs from XML, `https://github.com/daverog/Tripliser`.

[25] See Chapter 1.3 of this book.

[26] `http://stack.linkeddata.org`

[27] `http://teryt.stat.gov.pl/`

are USASpending.gov[28] and Federal Business Opportunities (FBO).[29] USASpending.gov offers a database of government expenditures, including *awarded* public contracts, for which it records, e.g., the numbers of bidders. On the other hand, FBO publishes public notices for *ongoing* calls for tenders. Once public notice's deadline for tender submission passes, the final number of bidders should be published along with other information about contract award in USASpending.gov. Unfortunately, these two sources do not publish enough data about public contracts to pair the equivalent instances reliably. While the same contract identifiers are used in some cases, most of the published contracts lack identifying information necessary for deduplication. Combination of data from the two sources thus only yields a small subset of public contracts that could be merged provided they are equipped with strong identifiers, such as URIs. USASpending.gov provides data downloads in several structured data formats.We used the CSV dumps, which we converted to RDF using SPARQL mapping[30] executed by tarql.[31] Data dump from FBO is available in XML as part of the Data.gov initiative.[32] To convert the data to RDF we created an XSLT stylesheet that outputs RDF/XML.[33] As an additional dataset used by both USASpending.gov and FBO, we converted the FAR Product and Service Codes[34] to RDF using LODRefine.[35] Data resulting from transformation to RDF was interlinked both internally and with external datasets. Internal linking was done in order to fuse equivalent instances of public contracts and business entities (both contracting authorities and bidders). Deduplication was performed using data processing unit for UnifiedViews that wraps the Silk link discovery framework.[36] The output links were merged using the data fusion component of UnifiedViews.[37] Links to external resources were created either by using code-based URI templates in transformation to RDF or by instance matching based on converted data. The use of codes as strong identifiers enabled automatic generation of links to FAR codes and North American Industry Classification System 2012,[38] two controlled vocabularies used to express objects and kinds of public contracts. Instance matching was applied to discover links to DBpedia[39] and OpenCorporates.[40] Links to DBpedia were created for populated places referred to from postal addresses in the U.S. procurement dataset. In this case, the employed Silk linkage rule was based on comparison of

[28] http://usaspending.gov/

[29] https://www.fbo.gov/

[30] https://github.com/opendatacz/USASpending2RDF

[31] https://github.com/cygri/tarql

[32] ftp://ftp.fbo.gov/datagov/

[33] https://github.com/opendatacz/FBO2RDF

[34] http://www.acquisition.gov/

[35] http://code.zemanta.com/sparkica/

[36] http://wifo5-03.informatik.uni-mannheim.de/bizer/silk/

[37] Developed previously for ODCleanStore, the predecessor of UnifiedViews [5].

[38] http://www.census.gov/eos/www/naics/index.html

[39] http://dbpedia.org

[40] https://opencorporates.com/

normalized ZIP codes and pre-filtering possible matches by transitively-expanded Wikipedia category for populated places in the U.S. Furthermore, OpenCorporates was used as target for linking the bidding companies. The task was carried out using the batch reconciliation API of OpenCorporates via interface in LODRefine. The links were established based on pre-filtering by jurisdiction and fuzzy matching on normalized legal name, with which the company is registered in the respective jurisdiction. In all cases of instance matching the samples of resulting links were verified by manual scrutiny, in order to estimate the linking accuracy.

1.4 LOD-Enabled Public Contract Matchmaking

1.4.1 Public Contracts Filing Application

Seeking the best possible match is an activity repeatedly undertaken by both sides of the procurement market: *buyers* (contracting authorities) and *suppliers* (bidders). Since most of the underlying computation, and even a considerable part of the GUI components, are similar for both cases, it makes sense to provide a single, web-based interface for them, which can be characterized as a *content-management system* for public contracts (and associated tenders). We denote the prototype developed[41] simply as *Public Contract Filing Application* (PCFA). Its features are (partly dynamically) derived from the Public Contracts Ontology.

1.4.1.1 Buyer's and Supplier's View

The *buyers* can use PCFA to create and manage their calls for tenders, publish them when they are ready, and wait for tenders from the candidate suppliers, as seen in Fig. 1.3. PCFA allows the buyer to compare the proposed call for tenders with other public contracts (published by the same buyer or by others) using the matchmaking functionality. The buyers can thus take into account, e.g., the cost or duration of similar contracts in the past, and adjust the proposed price or time schedule accordingly.

PCFA can help the buyers even further by allowing them to find suitable suppliers for their proposed call for tenders, as seen in Fig. 1.4. They can thus explicitly invite such potential suppliers via email and thus increase the competition for the given call. A published call for tenders can also be withdrawn; if the buyer later publishes a withdrawn call again, it is published as a new call, which ensures that the whole history of the procurement process is recorded properly. When the deadline for tenders passes, the buyers can easily reject the tenders that do not meet the criteria of the call, and they can also award the tender that offers the best conditions.

[41] The code of the PCFA is maintained at https://github.com/opendatacz/pc-filing-app.

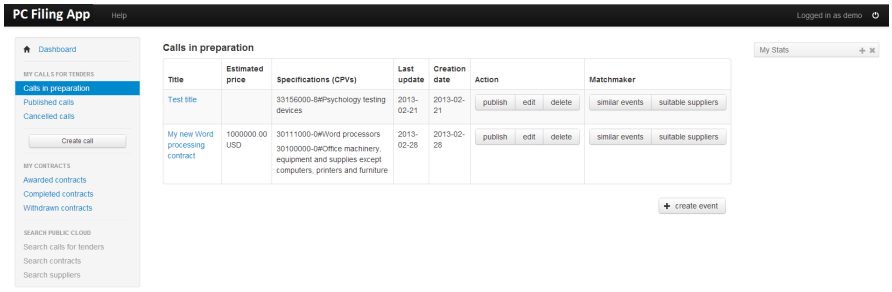

Fig. 1.3 Open calls for tenders from a buyer's perspective

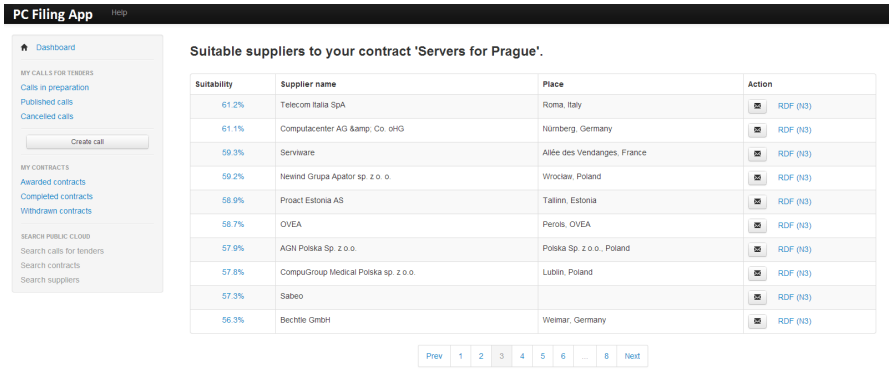

Fig. 1.4 Suitable suppliers for a call for tenders

Finally, when the public contract has been fulfilled, the buyer can briefly evaluate the progress and the outcome of the contract and record it in the application. This typically amounts to providing the actual price of the contract (which may be significantly higher than the price proposed in the awarded tender), the actual date when the contract was finished, and the overall experience of the buyer with this particular supplier. This information can later help the same buyer or other buyers in the future.

The interested *suppliers* can see the open calls for tenders suitable for them. This functionality is again provided by the matchmaking module. As mentioned above, a potential supplier can also be invited, by a contracting authority, to submit a tender to a certain call. The supplier can see and manage the received invitations in the PCFA as well. The invitations can be rejected or accepted; the latter automatically creates a tender. Alternatively, suppliers can find an open call for tenders on their own, and create a tender for it. The tender can be edited by the supplier and, when ready, submitted to the contracting authority. The suppliers are always informed by email when some update takes place for the call for tenders for which they submitted a tender.

1.4.1.2 Application Architecture

The *architecture* of the PCFA is modular, as can be seen in Fig. 1.5. Each module consists of both the client and the server side, which gives the developers freedom in what their module can do, making the application extensible. All the modules are based on a module template, which contains the code for user and context management as well as for *quad store*[42] access, so that the developer can focus on the added functionality. The modules use a shared relational database, which contains user information, user preferences and stored files, and can be used for caching of results of more complex SPARQL queries for faster response time when, for example, paging through a longer list of results. The public procurement data itself is stored in two instances of a quad store. The public quad store contains published data accessible to everyone (part of the LOD cloud). The private quad store contains unpublished data for each user of the application and for application management.

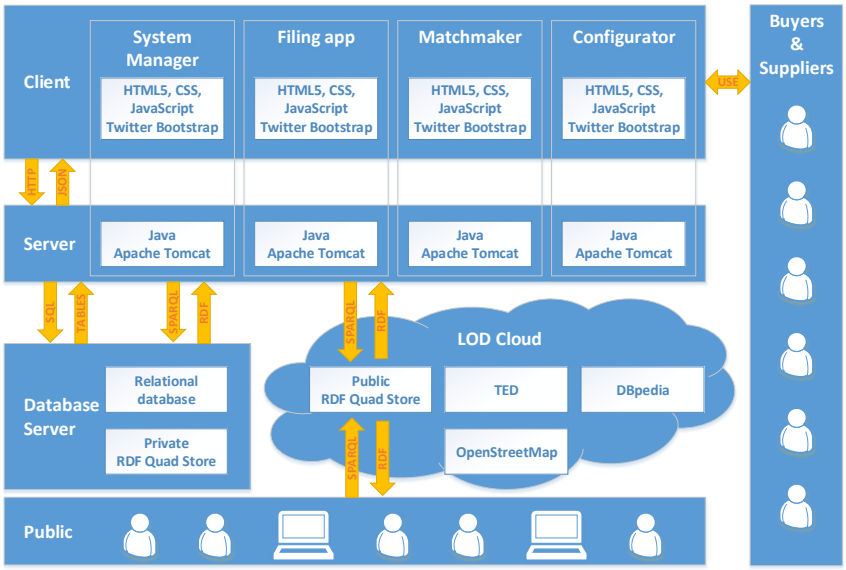

Fig. 1.5 PCFA architecture

The current implementation consists of the following modules. The system manager module handles registrations, logging in and out, and user preferences management. The filing module implements the lifecycles of calls for tenders, tenders and invitations to tenders. The matchmaking module implements the functionality behind the search for similar calls for tenders and suitable suppliers for contract-

[42] A database which stores RDF quads - subject, predicate, object and named graph, e.g., Openlink Virtuoso or Jena Fuseki.

ing authorities (buyers) and suitable open calls for tenders for suppliers. Finally, the configurations module allows the users to specify a more detailed configuration for individual types of public procurement (cars, IT, etc.).

There are two separate quad stores used in the application.[43] There is a *private* space for each user, where unpublished calls for tenders, tenders themselves and invitations for suppliers to submit a tender to a call are kept. This quad store is not accessible from the internet and is managed solely by the PCFA, which makes the private data secure. Additionally, there is the public space quad store, as part of the LOD cloud, where all the published information is kept and where also the calls for tenders to be compared by the matchmaker reside. This quad store is accessible to everyone on the internet for querying and downloading.

1.4.2 Matchmaking Functionality Internals

The core operation upon which others are based is the discovery of *contracts similar to a given contract*. To accomplish that, we first filter all contracts based on the similarity of CPV codes according to the hierarchical tree. Then we refine these results by applying additional comparers, specialized, e.g., in:

1. Tender deadlines: the shorter the interval between the two tender deadlines, the higher the similarity
2. Publication dates: the shorter the interval between the two public contract publication dates, the higher the similarity
3. Geographical distance: we measure the distance between the places where the public contracts were (or are to be) executed. For this purpose, the addresses are automatically converted to geo-coordinates.
4. Textual similarity: we compare the titles of contracts using the SoftTFIDF[3] algorithm.

The overall match score is then a weighted sum of the scores returned by all comparers.

When looking for *suitable suppliers* for a given call for tenders, the above approach is used in order to filter suppliers that have been previously awarded a contract similar to the current one. Similarly, when looking for *suitable calls for tenders* from the point of view of a supplier, the information (including CPV codes) from the supplier's profile is assembled into a virtual tender, which is matched against the open calls for tenders.

[43] They can be possibly replaced with two named graphs within a single quad store, each with a separate setting of access rights.

1.5 Aggregated Analysis of Procurement Linked Data

1.5.1 Analysis Scenarios

The data on public contracts, in combination with external data retrieved from the linked data cloud, can be submitted to aggregated analysis. The beneficiaries of such analysis can be:

- *Journalists and NGOs*: the data may help them reveal corruption and clientelism in public sector.
- *Official government bodies*: both specific supervisory bodies that address the issues of transparency and fair competition and statistical offices that collect data as part of aggregated information on the national economy.
- *Bidders*: analysing the previous successful and unsuccessful tenders may be helpful when preparing a new one; in long term, the companies may also actively plan their bidding strategies based on procurement market trends (revealed by automated analysis).
- *Contracting authorities*: they want to understand the supply side in order to know how to formulate the contract conditions, in view of successful matchmaking. Good progress of a future contract may be derived from previous experience with certain bidders. An additional goal may be to attract an adequate number of bidders; excessively many bidders bring large overheads to the awarding process, while too low a number may reduce competition (and, under some circumstances, even lead to contract canceling by a supervisory body, due to an anti-monopoly action).

1.5.2 Analytical Methods

A straightforward approach to aggregated analysis is via *summary tables and charts* expressing the relationship between, e.g., the number of contracting authorities, contractors, contracts, tenders, lots, or geographical localities. The value of contracts can be calculated as a sum or average per authority, contractor, region, kind of delivery, classification of goods etc. Charts can be generated for presentation of these statistics split by various dimensions (e.g. bar charts) or showing the evolution (e.g. line charts, timeline). The geographical dimension is best presented on maps: detailed data can be shown as points on the map, e.g., pointers with shaded tooltips on OpenStreetMap. The data for such analysis are normally provided by SPARQL SELECT queries, which allow to both retrieve the data and perform basic aggregation operations.

More sophisticated analysis can be provided by *data mining tools*, which automatically interrelate multiple views on data, often based on contingency table. As an example, see a fragment of analysis of U.S. procurement data with respect to the impact various attributes of a contract notice may have on the subsequent number

of tenders (Fig. 1.6). The association rules listed in the table fragment regard both a

1	38	8.753	**mainObject**(*Research and Development in the Physical, Engineer*) & **populationDensityPerKm2**(>=*1700*) >+< **Tenders**(>=*50*)
2	39	7.562	**mainObject**(*Research and Development in the Physical, Engineer*) & **populationDensityPerKm2**(>=*1700*) >+< **Tenders**(>=*30*)
3	36	7.306	**mainObject**(*Research and Development in the Physical, Engineer*) & **populationDensityPerKm2**(>=*1600*) >+< **Tenders**(>=*50*)
4	32	7.190	**mainObject**(*Research and Development in the Physical, Engineer*) & **populationDensityPerKm2**(*<1100;1600*)) >+< **Tenders**(>=*50*)
5	29	7.049	**mainObject**(*Research and Development in the Physical, Engineer*) & **populationDensityPerKm2**(*<1000;1500*)) >+< **Tenders**(>=*50*)
6	37	6.490	**mainObject**(*Research and Development in the Physical, Engineer*) & **populationDensityPerKm2**(>=*1600*) >+< **Tenders**(>=*30*)
7	24	6.481	**mainObject**(*Research and Development in the Physical, Engineer*) & **populationDensityPerKm2**(*<800;1300*)) >+< **Tenders**(>=*50*)
8	33	6.064	**mainObject**(*Research and Development in the Physical, Engineer*) & **populationDensityPerKm2**(*<1100;1600*)) >+< **Tenders**(>=*30*)
9	1	6.022	**mainObject**(*Research and Development in the Physical, Engineer*) >+< **Tenders**(*N/A*)
10	26	5.944	**mainObject**(*Research and Development in the Physical, Engineer*) & **populationDensityPerKm2**(*<900;1400*)) >+< **Tenders**(>=*50*)
11	2	5.882	**mainObject**(*Research and Development in the Physical, Engineer*) >+< **Tenders**(>=*100+*)
12	31	5.866	**mainObject**(*Research and Development in the Physical, Engineer*) & **populationDensityPerKm2**(*<1100;1500*)) >+< **Tenders**(>=*30*)
13	30	5.741	**mainObject**(*Research and Development in the Physical, Engineer*) & **populationDensityPerKm2**(*<1000;1500*)) >+< **Tenders**(>=*30*)
14	28	5.695	**mainObject**(*Research and Development in the Physical, Engineer*) & **populationDensityPerKm2**(*<1000;1400*)) >+< **Tenders**(>=*30*)
15	3	5.571	**mainObject**(*Research and Development in the Physical, Engineer*) >+< **Tenders**(>=*50*)
16	18	5.494	**mainObject**(*Research and Development in the Physical, Engineer*) & **populationDensityPerKm2**(*<400;900*)) >+< **Tenders**(>=*50*)
17	16	5.388	**mainObject**(*Research and Development in the Physical, Engineer*) & **populationDensityPerKm2**(*<400;700*)) >+< **Tenders**(>=*30*)
18	17	5.235	**mainObject**(*Research and Development in the Physical, Engineer*) & **populationDensityPerKm2**(*<400;800*)) >+< **Tenders**(>=*30*)
19	19	5.210	**mainObject**(*Research and Development in the Physical, Engineer*) & **populationDensityPerKm2**(*<400;900*)) >+< **Tenders**(>=*30*)
20	20	5.080	**mainObject**(*Research and Development in the Physical, Engineer*) & **populationDensityPerKm2**(*<500;900*)) >+< **Tenders**(>=*30*)
21	25	5.049	**mainObject**(*Research and Development in the Physical, Engineer*) & **populationDensityPerKm2**(*<800;1300*)) >+< **Tenders**(>=*30*)
22	34	5.047	**mainObject**(*Research and Development in the Physical, Engineer*) & **populationDensityPerKm2**(*<1200;1700*)) >+< **Tenders**(>=*30*)
23	35	4.965	**mainObject**(*Research and Development in the Physical, Engineer*) & **populationDensityPerKm2**(*<1300;2000*)) >+< **Tenders**(>=*30*)
24	14	4.923	**mainObject**(*Research and Development in the Physical, Engineer*) & **populationDensityPerKm2**(*<300;800*)) >+< **Tenders**(>=*50*)
25	4	4.916	**mainObject**(*Research and Development in the Physical, Engineer*) >+< **Tenders**(>=*30*)

Fig. 1.6 Discovered factors correlated with number of tenders

CPV code of the contract object (`mainObject` attribute), originating from one of the core procurement dataset, and the population density attribute, originating from DBpedia. It indicates that contracts for 'Research and Development in the Physical, Engineering, and Life Sciences' in localities with higher population density tend to attract a high number of tenders (as higher interval values for the former mostly coincide with higher values for the latter, in the individual rules).

1.5.3 Integration of Analytical Functionality into PCFA

Although the central role in the PCFA scenarios is reserved to matchmaking, there are also reserved slots for invocation of analytical features. Since this part of implementation has been ongoing till the final months of the project, it is not yet functional at the time of completing this chapter of the book. The analytical functionality will be at the disposal of the buyer (contracting authority), and will amount to:

- *Interactively exploring*, in graphical form,the linked data about
  - the current notice
  - a (matching) historical notice/contract
  - a relevant supplier, including its contracts.
- Viewing *suggested values* for the remaining pieces of contract notice information based on the already provided ones. The values will be provided by an inductively trained recommender.
- Getting an estimate of the *number of bidders* for (as complete as possible) contract notice information. For this, a predictive ordinal classifier will be developed.

When the integration of analytical functionality has been completed, usability testing by several contract authorities' representatives will take place.

1.6 Conclusions

The chapter outlined some of the promises and intricacies of using linked open data in the area of public procurement. It went through the different, yet interrelated partial tasks: data extraction and publishing (leveraging on Public Contracts Ontology as domain-specific, externally interlinked vocabulary), buyer/supplier matchmaking, and aggregated analytics.

Despite the numerous technical difficulties, especially as regards the coverage and quality of existing open data sources, it is clear that handling procurement data in RDF format and linking them to other (government, geographic, commercial, encyclopedic, etc.) data opens novel avenues for their matchmaking and aggregate analytics. The use of common data format (RDF), as well as common domain vocabulary (PCO) and classifications (such as CPV and TERYT) allow for integration of *external data*; furthermore, as the data are separated from their initial applications, they can be consumed by *third-party* applications originally developed for matchmaking over other procurement datasets. The often implicit model of legacy data can also be compared with a carefully crafted ontological domain model and *ambiguities* can be discovered. Finally, the data itself potentially becomes *cleaner* during the extraction and transformation process, so, even if some of the analytic tools require it to be downgraded back to simpler formats such as CSV, its value may be higher than the initial one.

Future work in this field will most likely concentrate on the last two tasks (matchmaking and analytics), however, with implication on extraction and publishing, too. Namely, precise matchmaking will require RDFization and publication of further information, some of which (such as detailed specifications of procured goods or services) will have to be extracted from *free text*. Exploitation of *product ontologies* such as those developed in the OPDM project[44] could then be beneficial. The analytic functionality should more systematically exploit external linked data as predictive/descriptive features [6]. Given the size and heterogeneity of the LOD cloud, smart methods of *incremental data crawling* rather than plain SPARQL queries should however be employed.

Finally, while the current research has been focused on the primary intended users of the PCFA, i.e. contract authorities and (to lesser extent) bidders, the remaining stakeholders should not be forgotten. While the generic features of contracts, products/services and bidders, captured by the *generalized* features (such as price intervals, geographic regions or broad categories of products) in data mining results, are important for these parties, directly participatint in the matchmaking process, there are also *NGOs and supervisory bodies* that primarily seek *concrete*

[44] http://www.ebusiness-unibw.org/ontologies/opdm/

corruption cases. To assist them, graph data mining methods should be adapted to the current state of linked data cloud, so as to detect, in particular, instances of suspicious patterns over the (RDF) graph representing organizations (contract authorities, bidders and others), contracts and people engaged in them.

References

1. Jose María Alvarez and José Emilio Labra. Semantic methods for reusing linking open data of the european public procurement notices. In *ESWC PhD Symposium*, 2011.
2. Jose María Alvarez, José Emilio Labra, Ramón Calmeau, Ángel Marín, and Jose Luis Marín. Innovative services to ease the access to the public procurement notices using linking open data and advanced methods based on semantics. In *5th International Conference on Methodologies, Technologies and enabling eGovernment Tools*, 2011.
3. William W. Cohen, Pradeep Ravikumar, and Stephen E. Fienberg. A comparison of string distance metrics for name-matching tasks. pages 73–78, 2003.
4. Isabella Distinto, Mathieu d'Aquin, and Enrico Motta. LOTED2: an Ontology of European Public Procurement Notices. Under review for *Semantic Web – Interoperability, Usability, Applicability*, IOS Press, 2014.
5. Jan Michelfeit and Tomas Knap. Linked data fusion in ODCleanStore. In Birte Glimm and David Huynh, editors, *International Semantic Web Conference (Posters and Demos)*, volume 914 of *CEUR Workshop Proceedings*. CEUR-WS.org, 2012.
6. Heiko Paulheim. Exploiting linked open data as background knowledge in data mining. In Claudia d'Amato, Petr Berka, Vojtech Svátek, and Krzysztof Wecel, editors, *DMoLD*, volume 1082 of *CEUR Workshop Proceedings*. CEUR-WS.org, 2013.
7. Siemens. Study on the evaluation of the action plan for the implementation of the legal framework for electronic procurement (phase ii): Analysis, assessment and recommendations. Technical report, July 2010.
8. Francesco Valle, Mathieu d'Aquin, Tommaso Di Noia, and Enrico Motta. LOTED: Exploiting Linked Data in Analyzing European Procurement Notices. In *1st Workshop on Knowledge Injection into and Extraction from Linked Data collocated with EKAW'10*, Madrid, Spain, 2010. CEUR-WS.org.