



SC6DI06692 - D8.12.1

Piloting the Core Public Service Vocabulary

“Describe your public service once to publish on multiple Government Access Portals”

JOINING UP GOVERNMENTS



Document Metadata

Property	Value
Release date	2013-05-02
Status	<i>Final draft</i>
Version	0.07
Authors	<i>Phil Archer – W3C Stijn Goedertier – PwC EU Services Nicolas Loozen – PwC EU Services Nikolaos Loutas – PwC EU Services</i>
Reviewed by	<i>Pieter Breyne – PwC EU Services</i>
Approved by	

Disclaimer

This report was prepared for the ISA programme by PwC EU Services.

The views expressed in this draft specification are purely those of the authors and may not, in any circumstances, be interpreted as stating an official position of the European Commission.

The European Commission does not guarantee the accuracy of the information included in this study, nor does it accept any responsibility for any use thereof.

Reference herein to any specific products, specifications, process, or service by trade name, trademark, manufacturer, or otherwise, does not necessarily constitute or imply its endorsement, recommendation, or favouring by the European Commission.

All care has been taken by the author to ensure that s/he has obtained, where necessary, permission to use any parts of manuscripts including illustrations, maps, and graphs, on which intellectual property rights already exist from the titular holder(s) of such rights or from her/his or their legal representative.

Table of Contents

TABLE OF CONTENTS.....	2
LIST OF FIGURES.....	3
LIST OF TABLES	3
1 INTRODUCTION	4
1.1 GLOSSARY	4
1.2 CORE PUBLIC SERVICE VOCABULARY.....	5
1.3 BUSINESS NEED	7
1.4 USE CASE: ENABLE THE CROSS-BORDER SEARCH FOR PUBLIC SERVICES.....	7
1.5 STAKEHOLDERS	9
2 URI SETS FOR SERVICE DATA	10
3 MAPPING EXERCISE.....	3
3.1 SPOCS NEXT-GENERATION POSC SERVICE DATA.....	3
3.2 FLEMISH INTERGOVERNMENTAL PRODUCT AND SERVICE CATALOGUE (IDPC).....	8
3.3 IRISH PUBLIC SERVICE DESCRIPTIONS	2
4 LINKED DATA INFRASTRUCTURE	5
4.1 USING REFINE RDF TO CREATE SERVICE DESCRIPTIONS FROM A SPREADSHEET.....	5
1.4.1 <i>Using the Spreadsheet template</i>	<i>5</i>
1.4.2 <i>Import Your Spreadsheet into Google Refine</i>	<i>6</i>
1.4.3 <i>Add the Pre-defined Mappings</i>	<i>6</i>
1.4.4 <i>Export the RDF.....</i>	<i>8</i>
4.2 USING XSLT TO TRANSFORM XML DESCRIPTIONS INTO CPSV RDF	10
4.3 USING OPENLINK VIRTUOSO AS A TRIPLE STORE AND SPARQL ENDPOINT	10
4.4 USE CASE – QUERYING THE TRIPLE STORE	11
4.4.5 <i>List all public services described in the data</i>	<i>11</i>
4.4.6 <i>Finding Services of a Given Type in a Given Country.....</i>	<i>13</i>
4.5 USE CASE – LOOKING UP A SERVICE DESCRIPTION	16
5 CONCLUSIONS AND FUTURE WORK	20
5.1 KNOWN LIMITATIONS.....	20
REFERENCES.....	22
ACKNOWLEDGEMENTS	24

List of Figures

Figure 1 The UML Diagram for the Core Public Service Vocabulary.....	5
Figure 2 Use case: enable the cross-border search for public services	8
Figure 3 – The 10 Dos and DONTs for persistent URIs [PURI].....	1
Figure 4 Partial screenshot of the ESD Toolkit description of the Architect Registration Service Type.....	1
Figure 5 Part of the RDF returned from the same URI when RDF/XML is requested	1
Figure 6 The UML Diagram Showing the IPDC Data Model.....	1
Figure 7 Partial screenshot of the non-technical spreadsheet template	3
Figure 8 Partial screenshot of spreadsheet template filled in with data from the SPOCS deliverable	5
Figure 9 The Google Refine Configuration	6
Figure 10 Editing the RDF Skeleton.....	7
Figure 11 Applying the Operation History	8
Figure 12 Exporting the RDF.....	9
Figure 13 – Representation of Virtuoso’s Linked Data system architecture	11
Figure 14 Searching for 'Architect' in the ESD Toolkit Service List [SL4]	14
Figure 15 ESD's Linked Data Information Concerning Architect Registration	15
Figure 16 Data returned to an RDF client dereferencing http://cpsv.testproject.eu/id/mlt/PublicService/RealEstateAgencyLicence	18
Figure 17 Web page seen in a regular browser when visiting http://cpsv.testproject.eu/id/mlt/PublicService/RealEstateAgencyLicence	19

List of Tables

Table 1 – Glossary	4
Table 2 – Stakeholders and Roles	9

1 Introduction

This document describes the results of an exercise piloting the Core Public Service Vocabulary.

The pilot is expected to have the following outcomes:

- It will demonstrate that the Core Public Service can be used as a foundational RDF Vocabulary to **homogenise public service data** that originates from local, regional, and national e-Government portals;
- It will demonstrate that the definition of uniform **HTTP URI sets** for public services facilitates information management;
- It will demonstrate that the use of **common controlled vocabularies** can greatly facilitate the search for public services, especially in a multilingual context;
- It will demonstrate that a **linked data infrastructure** can provide access to *homogenised, linked, and enriched* public service data using standard Web-based interfaces (such as HTTP and SPARQL) and Web-based languages (such as HTML and RDF), on top of existing relational/spatial database systems;
- It will demonstrate that the use of standard Web interfaces (such as HTTP(S) and SPARQL) can **simplify the use of public service data** for humans and machines.

1.1 Glossary

The table below contains an overview of terms and acronyms used throughout this document.

Table 1 – Glossary

Term / Acronym	Description
ISA Programme	The Interoperability Solutions for European Public Administrations Programme of the European Union
LSP	Large-scale pilot
RDF	Resource Description Framework
PoS	Point of Single Contact
SKOS	Simple Knowledge Organization System – RDF Vocabulary for the representation of key reference data such as code lists, and taxonomies.
SPARQL	SPARQL Query Language for RDF
URI	Uniform Resource Identifier
URI Set	A collection of reference data published using URIs, about a single concept, governed from a single source.
URL	Uniform Resource Locator

1.2 Core Public Service Vocabulary

Like any core vocabulary, the CPSV does not, and cannot, provide *all* the terms necessary to describe *every* public service in *all* contexts. Rather, it provides a foundation that, when used, provides a common interoperability layer.

The vocabulary is summarised in Figure 1 and in the following text which also appears in the specification document itself.

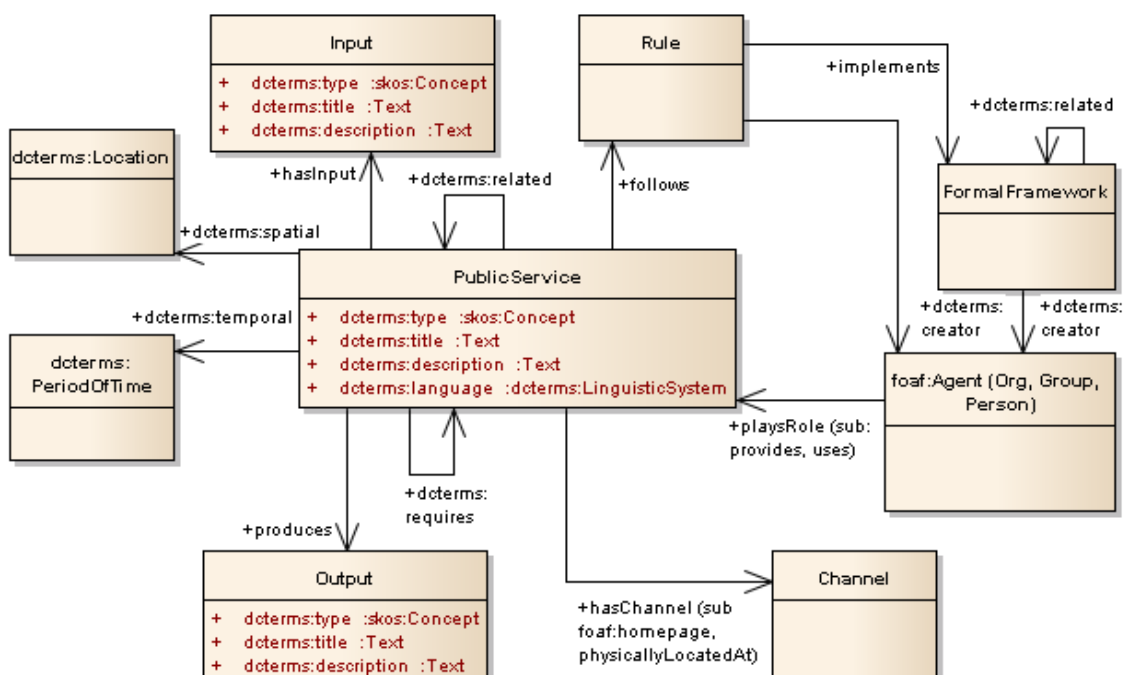


Figure 1 The UML Diagram for the Core Public Service Vocabulary

At the heart of the model is the public service itself. This will very likely have a name, a description and, in many cases, will be of a specific type. For greatest interoperability, service types should be given as values from a list such as the service list used in many EU countries [SL4]. The service is likely to be available through multiple channels including a Web site, one or more physical locations and so on. The generic `hasChannel` property links the service to any such `Channel`. CPSV asserts that the well known `foaf:homepage` property is a sub property of `hasChannel` and mints a further sub property `physicallyAvailableAt` which links a service to a `dcterms:Location`. Details of the location(s) can be given using the Location Core Vocabulary [Location] or similar. A service will often be made available in multiple languages that can be specified using `dcterms:language`.

A service will usually require some sort of input. In the case of issuing a driving licence this will

be evidence that driving test has been passed; many services will require some sort of proof of ID and so on. Likewise, the output will vary depending on the specific service but there will usually be a document or other artefact that is the output. This is not the same as the *outcome*. Drawing on the definitions used in StratML [StratML], if the service controls all of the necessary inputs and processes, the desired result is an *output*. Likewise, the GEA Public Service Model, distinguishes between public service outcome, output and effect [LOU, PER]. For example, a driving licence is an output. The outcome (or effect in GEA) is that the new licence holder can drive a vehicle on the public highway. How they do that, which vehicle they drive etc. is beyond the service's remit.

Public services are regulated by a set of rules. These will typically be set by a single organisation and will implement a combination of legislation and policy, i.e. the rules will sit within a Formal Framework that may be decided at any level from local to supranational by any number of bodies. It is not the task of the CPSV to model detailed relationships between policies and legislation, however, `dcterms:related` may be used to link such items and it is noteworthy that controlled service type lists are themselves likely to offer hints and links to relevant documentation that empowers or requires the provision of the service. The creator(s) of the rules and formal framework are the bodies responsible for their creation, not the individuals who wrote them. It is also notable that the Rule and Formal Framework classes are both subclasses of the FRBR class Expression [FRBR].

An individual service may be related to another in some way, in which case the two services can be linked using `dcterms:related`. If the relationship is such that one service *requires* another then the `dcterms:requires` relationship should be used. The `dcterms:Agent` class represents any individual, group or organisation that plays any role in the service. These include but are not limited to:

- the public administration responsible for providing the service;
- the public administration that defines the rules that regulate the service;
- the organisation(s) that deliver the service on behalf of the responsible public body;
- the public body responsible for passing the legislation or setting the policy or policies from which the rules are derived;
- the person, organisation or group that uses the service.

The basic roles are `provides` and `uses` and specific object properties are provided for these as shortcuts. However there are any number of roles that may be played in the provision or use of a service. Therefore a `hasRole` super property is provided.

Details of the specific role played by an Agent can be provided using the Role and Membership classes defined in the Organization Ontology [ORG] which in turn derived them from FOAF

[FOAF]. It allows for the roles to be defined separately from the agents that fulfil those roles and for any number of agents to be associated with any number of roles.

Finally the service is likely to be available within a defined area and/or time frame. These limits are recorded using the Dublin Core terms `dcterms:spatial` and `dcterms:temporal` together with their respective classes.

1.3 Business need

The decision to develop the Core Public Service Vocabulary followed an extensive survey of public administrations across Europe that indicated a need for it. Several portals exist through which services can be discovered but there is little commonality between them in the way the services are described.

This picture has been borne out by the pilot study which allowed an analysis of the situation in Flanders. Despite its relatively small size, the region has 308 municipalities¹. Many of these administrations act as points through which essentially the same set of services are available leading to as many descriptions of the same services as there are administrations. Furthermore, many phone calls received from members of the public seeking services were received by the wrong administration - the service being sought was most often provided by a different tier of government.

This inefficiency - duplicate descriptions of the same services, lack of readily available information about the public administration responsible for a given service - can be greatly improved using machine readable descriptions that can be created once and re-used any number of times.

1.4 Use case: enable the cross-border search for public services

The business need outlined above is derived in part from the fact that services tend to be defined at a high level, by a combination of legislation and government policy, but implemented multiple times at a local level. Governments use local, regional, and national access portals to give businesses, citizens, and public administrations basic information about their public services. Unfortunately, the basic information coming from the centre is often duplicated, unstructured, and not machine-readable. This fragmentation makes it difficult for citizens, businesses, and public administrations to find information about the public service that they need. It also leads to situations where the same information about a public service is re-created in different applications or by different governments. The scope for errors and mismatches is immediately obvious.

The Core Public Service Vocabulary allows public administrations to describe their service only once using a standard, extensible, and machine-readable vocabulary and make these

¹ <http://www.vlaanderen.be/nl/ontdek-vlaanderen/provincies-en-gemeenten>

descriptions re-used on many governmental access portals. This pilot implementation shows how this can be done using a Linked Data infrastructure for a small sample of public service descriptions originating from various European public administrations.

The requirements for registering as an Architect, for example, will be the same in different locations in a given country as they are defined centrally. Therefore a basic use case within a country can be broken down into:

1. What is the service I need in this country?
2. What are the requirements for me to comply with the law?
3. Where can access the service, online? via a hotline? perhaps at a local office?

The enquiry can come from anywhere of course and so 'local office' might not even be in the relevant country. For example, if a Portuguese tourist loses a passport in Estonia, the rules for acquiring a new one will apply across Portugal but the local office is likely to be the Portuguese embassy in Tallinn.

A comprehensive set of use cases is included in the CPSV specification itself, including that shown in Figure 2. The pilot will demonstrate how the Core Public Service vocabulary could support the cross-border search for public services by public administrations, businesses and citizens.

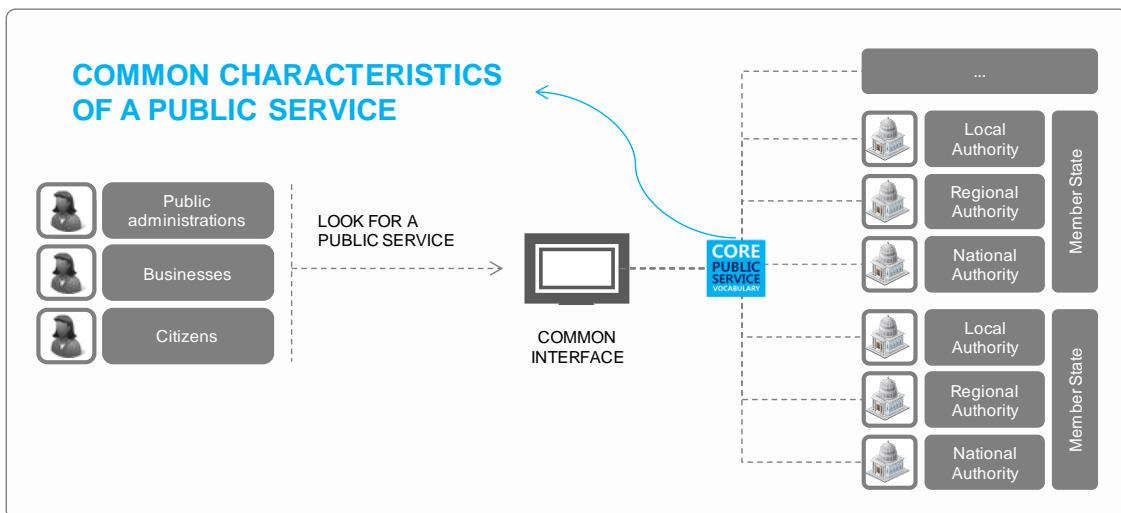


Figure 2 Use case: enable the cross-border search for public services

1.5 Stakeholders

The table below lists the stakeholders involved in this pilot.

Table 2 – Stakeholders and Roles

Stakeholder	Description
<u>e-CODEX</u> consortium	<p>Large scale pilot on eJustice that aims at improving the cross-border exchange of information in legal proceedings - where citizens, businesses and governments are involved - in a safe, accessible and sustainable way.</p> <p>Role in pilot: Invited to share descriptions of services piloted.</p>
<u>ePSOS</u> consortium	<p>Large scale pilot on eHealth that aims at making it easier for people to receive medical assistance anywhere in the EU by removing linguistic, administrative and technical barriers.</p> <p>Role in pilot: Invited to share descriptions of services piloted.</p>
<u>e-SENS</u> consortium	<p>Large scale pilot on Electronic Simple European Networked Services that aims to develop an infrastructure for interoperable public services in Europe.</p> <p>Role in pilot: Invited to share descriptions of services (to be) piloted.</p>
<u>ISA Programme</u>	<p>The Interoperability Solutions for European Public Administrations Programme of the European Union</p> <p>Role in pilot: Initiated and funds the work.</p>
<u>PEPPOL</u> consortium	<p>Large scale pilot on eProcurement that aims to make it easier for companies to bid for public sector contracts anywhere in the EU in a simpler and more efficient way.</p> <p>Role in pilot: Invited to share descriptions of services (to be) piloted.</p>
<u>SPOCS</u> consortium	<p>Large scale pilot in the context of the Services Directive that aims at setting up the next-generation national points of single contacts, i.e. "one-stop shops" to facilitate contacts between businesses and national public administrations and support the completion of administrative procedures online;</p> <p>Role in pilot: Provided descriptions of services piloted.</p>
<u>STORK</u> consortium	<p>Large scale pilot on e-Identity that aims at making the cross-border use of electronic identification possible, allowing citizens and businesses to access eGovernment services in any Member State they live in or travel to.</p> <p>Role in pilot: Invited to share descriptions of services (to be) piloted.</p>
<u>CORVE</u>	<p>CORVE, the Flemish e-Government unit.</p> <p>Role in the pilot: provided a sample of 9 service descriptions from their Inter-governmental product and services catalogue (IDPC).</p>

2 URI sets for service data

A recurring theme is the re-use of open data, that is: data created in one context is re-used in another. The most highly re-usable data of all is data published as URI sets that provides a set of identifiers for basic infrastructure like buildings, roads, agencies, communities etc. Alongside that it is also necessary to be able to identify common ideas and concepts and again, URI sets are the most flexible and powerful means of doing this.

Unlike any other identifiers, there is a global infrastructure in place so that HTTP URIs can be dereferenced, i.e looked up. A human can put a URI straight into a browser and receive an HTML page in return that either *is* the identified item if it's a digital resource or is a *description* of the item if it is a physical resource. Exactly the same identifier can be used by a computer program to fetch machine readable data and the number of formats in which the data can be returned is not bounded by any current technology. HTTP was designed to be completely agnostic about the types of document it handles and servers can easily be configured to return any number of different formats.

HTTP supports the active provision of data in a technology-neutral manner. URIs can persist and still be useful even if an individual technology becomes obsolete. If HTTP itself becomes obsolete, as it surely will one distant day, the identifier is no less useful than any other that can't be dereferenced such as a UUID.

These ideas and more are discussed in detail in the 2012 study 10 Rules for Persistent URIs [PURI] developed by the ISA Programme, which is summarised in Figure 3.

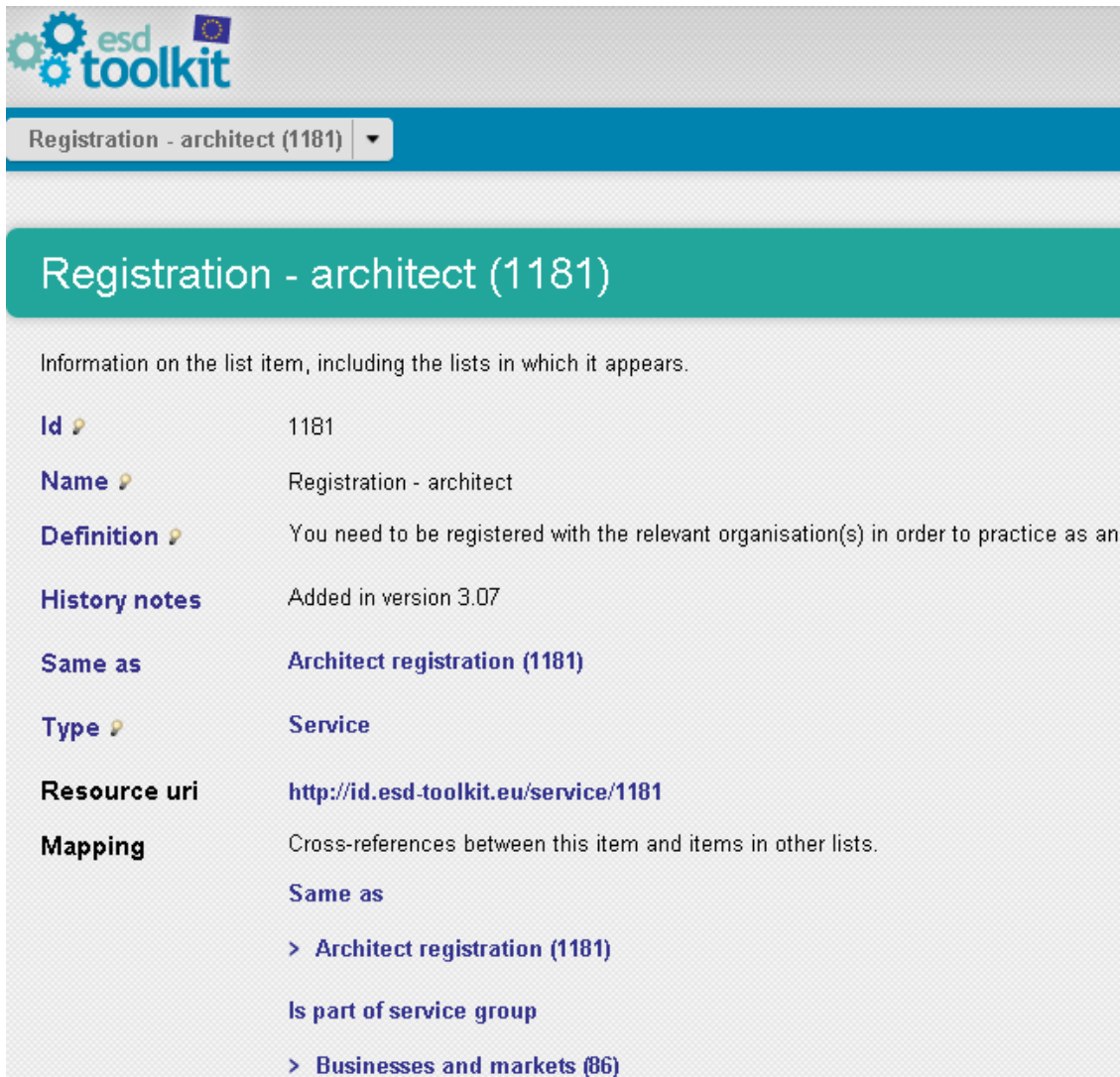
In the case of the current pilot, URI sets are useful for describing various elements:

- the legislation that requires or empowers the provision of the service
- the policies that describe how a particular service will be run, i.e. the implements the legislation;
- the locations, Web pages etc. through which the service is accessible;
- the country in which the service operates;
- the type of public service.

The last item on that list is particularly important for cross border searches for public services. The ESD Toolkit Service List [SL4] is used in many countries to provide a referenceable list of public service types and, among other means, is made available as a URI set. For example: <http://id.esd-toolkit.eu/service/1181>. This identifies the specific service type of architect registration (architects need to be formally registered to operate in many countries). Put into a normal Web browser it will return a Web page that describes the service type (Figure 4). Deference **the same URI** using software that accepts RDF/XML and machine readable data is returned (Figure 5).



Figure 3 – The 10 Dos and DONTs for persistent URIs [PURI]



Information on the list item, including the lists in which it appears.

Id	1181
Name	Registration - architect
Definition	You need to be registered with the relevant organisation(s) in order to practice as an
History notes	Added in version 3.07
Same as	Architect registration (1181)
Type	Service
Resource uri	http://id.esd-toolkit.eu/service/1181
Mapping	Cross-references between this item and items in other lists.
	<p>Same as</p> <ul style="list-style-type: none"> > Architect registration (1181) <p>Is part of service group</p> <ul style="list-style-type: none"> > Businesses and markets (86)

Figure 4 Partial screenshot of the ESD Toolkit description of the Architect Registration Service Type

```
<defesdorguk:ServiceClass rdf:about="http://id.esd-toolkit.eu/service/1181">
  <dcterms:identifier>1181</dcterms:identifier>
  <skos:prefLabel>Registration - architect</skos:prefLabel>
  <skos:historyNote>Added in version 3.07</skos:historyNote>
  <skos:definition>You need to be registered with the relevant
    organisation(s) in order to practice as an architect</skos:definition>
</defesdorguk:ServiceClass>
</rdf:RDF>
```

Figure 5 Part of the RDF returned from the same URI when RDF/XML is requested



It's worth noticing that the URI in this example has been *designed* for persistence. Look closely at the URI itself:

```
http://id.esd-toolkit.eu/service/1181
```

This can be broken down as:

```
http://{type}.esd-toolkit.eu/{concept}/{reference}
```

This is a slight variation on the pattern defined in Designing URI Sets for the UK Public Sector [UKURI]. In brief, the `{type}` in this example is `id` which generally means that the URI identifies a non-information resource, i.e. something that cannot be delivered electronically, like a building or a person (other typical values are `doc` and `ref`). The `{concept}` in this case is a service and the `{reference}` is the specific service identified, the code for which is 1181. Again, these ideas are discussed further in 10 Rules for Persistent URIs [PURI].

3 Mapping exercise

This section contains a description of the sample datasets and a mapping to the Public Service RDF Vocabulary v0.3, released for public review. The provided sample dataset contains information about the following public services:

- Next-generation points of single contact services in Malta, Portugal, Lithuania piloted by the SPOCS LSP [SPOCS-D5.13];
- Flemish Intergovernmental Product and Service Catalogue (IDPC);
- Irish Public Service Descriptions.

3.1 SPOCS next-generation PoSC service data

The pilot will describe four services piloted by the SPOCS LSP:

- **Lithuania – Permission for Temporary Provision of Architect’s Services:** The procedure of the recognition of professional qualifications of Architect pursuant to the occasional or temporary provision of architectural services in the Republic of Lithuania
- **Portugal – Travel Agent Registration Service:** The Travel Agent Registration Service enables any travel agent to register, validate and certify its activities by using the official TA service provided by the CA (Turismo de Portugal), in order to perform cross-border activities in Portuguese territory.
- **Portugal – Tourism Entertainment Companies Registration Service:** The TEC Service enables any Tourism Entertainment Company to register, validate and certificate their activities by using the official CA (Turismo de Portugal) TEC service, in order to perform cross-border activities in Portuguese territory.
- **Malta – Obtaining a real estate licence:** Obtaining information and applying for a licence to start up in Malta: The service will be available on PSC portal www.gov.mt The service provider can find information about establishing a business and needs to register in order to initiate the procedure.
- **Slovenia – Temporary entry in the Slovene directory of architects:** Attestation about the temporary entry in the directory of architects from the Slovenian Chamber of Architecture and Spatial Planning of Slovenia.

The SPOCS deliverable that includes these service descriptions [SPOCS-D5.13] covers a substantial piece of work but for our purposes, the important aspect is how the authors described a set of public services. This was done using a single set of attributes for each service. From our perspective, this is a descriptive vocabulary and so the task is to see how the SPOCS vocabulary maps to the CPSV.

One immediate difference is that in the SPOCS case, all terms describe the service – it's a 2 dimensional model. In the CPSV, there are multiple classes and relationships. In following table we map the SPOCS terms to the CPSV with notes where relevant.

SPOCS attribute	SPOCS definition	CPSV term	Notes
Name	The official name of the service.	skos:prefLabel	The assumption in making these mappings is that the short name will be the common name. ORG uses skos:preflabel for official but usually unused names, skos:altLabel for others. CPSV uses dcterms:title as the single means of providing a service name and makes no mention of the SKOS terms suggested here.
ShortName	The unofficial short name of the service.	dcterms:title	
Abbreviation	The abbreviation of the service.	skos:altLabel	
ServiceOutline	The ServiceOutline is a short description of the service giving an overview about the scope and purpose of the service.	None	A sub property of dcterms:description could be defined for this if desired.
ServiceDescription	The ServiceDescription gives a comprehensive description of the purpose and scope of the service.	dcterms:description	
ServiceConsumer	ServiceConsumers gives an overview who is the audience or the target group of the offered service. It also defines the scope of the service.	foaf:Agent, linked using cpsv:uses	A sub class of foaf:Agent is likely to be used, such as foaf:Person, and it may be appropriate to define a sub property of cpsv:uses. CPSV does NOT include scope in this class.
ProcedureDescription	ProcedureDescription is describing the activities of the procedure of the service. This is the textual description how the procedure for this service will be enforced including the process steps of the customer and the responsible CA.	cpsv:Rule linked from the service using cpsv:follows	
ProcedureModel	The structured description or modelled view of the procedure. Based on the textual description the ProcedureModel defines in a specific modelling notation the process, communication and information flow of the procedure.	Not mapped.	It is notable that this attribute is only used once in the SPOCS examples and that the value given is simply 'Malta Tourism Authority' which doesn't match the definition given here.
ServiceAgency	The agency that is providing the service	These two attributes	

	<p>operationally (CA) to the customer (SP). The ServiceAgency is the executor of the procedure and contact for specific questions regarding the service.</p>	<p>are not included in the CPSV but could readily be modelled using <code>cpsv:playsRole</code> or a newly minted subproperty thereof. <code>ORG</code> provides mechanisms to model more specific relationships.</p>	
ServiceIntermediary	<p>ServiceIntermediaries are broker or one-stopshops regarding the service (e.g. citizen bureaus). They are primary contacts for the communication with the customer, but not responsible for detailed questions.</p>		
DescriptionOwner	<p>The organisation that is responsible for describing the service.</p>	Not mapped	This would be covered using the provenance vocabulary [PROV-O]
ServiceOwner	<p>The agency that is responsible for the service.</p>	<p><code>foaf:Agent</code>, linked from the service using <code>cpsv:provides</code></p>	
ContactPerson	<p>A physical person the user can contact</p>	not mapped	Such a person would be described as a property of the service provider using a suitable vocabulary such as <code>vCard</code> .
Fees	<p>The fees are describing all dues and rates that have to be paid for using the service. As some services have fees that are dependent to individual aspects of the service it is just a textual description. The description includes also the payment methods and conditions.</p>	<p><code>cpsv:Input</code> (linked from the service using <code>cpsv:hasInput</code>).</p>	<p>A fee is a kind of input and this would be described using <code>dcterms:title</code> and <code>dcterms:description</code>. Further details of the fees might be given as part of the Rule(s). <code>dcterms:type</code> can be used to link to a <code>skos:Concept</code> so that the specific Input is declared to be of type 'Fee.'</p>
Respites	<p>Respites are describing the time limits and durations regarding the procedure of the service. The description details for each time limit or duration the conditions for start and/or end.</p>	<p><code>dcterms:PeriodOfTime</code> linked from the service using <code>dcterms:temporal</code></p>	<p>CPSV uses Dublin Core terms to describe the temporal limits of a service.</p>
LegalBasis	<p>LegalBasis describes what laws and legal or administrative regulations are related to the service. The text</p>	<p><code>cpsv:FormalFramework</code> linked from a Rule</p>	<p>In the CPSV, legislation should be identified with a URI and then the textual description applied. The</p>

	can include links to the legal text.	using cpsv:implements	forthcoming European Legislation Identifiers can be used directly for this purpose. The concept of SPOCS LegalBasis is however identical to the CPSV Formal Framework.
Forms	Forms are structured documents for receiving information from the customer. Forms contains descriptions what forms need to be filled applying for the service. The text can include links to online forms or external eServices.	cpsv:Input (linked from the service using cpsv:hasInput).	
NeededDocuments	NeededDocuments describes what documents are necessary during the different steps of the application procedure. Information has to be structured to show when and under which conditions documents have to be provided by the customer. NeededDocuments can be related to a DocumentType.	cpsv:Input (linked from the service using cpsv:hasInput).	
PreConditions	PreCondition describes which conditions need to be fulfilled for starting with the application procedure.	cpsv:Rule linked from the service using cpsv:follows	Descriptions of this type would be a Rule – something that describes how a service is implemented.
PostConditions	PostCondition describes which conditions need to be fulfilled at the end of the application procedure.		
FurtherInformation	FurtherInformation links to external web pages or online documents providing information related to the service.	foaf:homepage, rdfs:seeAlso etc.	This is a common requirement when describing resources and many well known properties exist for this purpose.
ServiceOutput	ServiceOutput defines the outcome of the service (product or status) and thus the result of the procedure.	cpsv:Output linked from the service using cpsv:produces	
ID	The identifier (ID) is the unique element. It is valid within a namespace (incl. country code and	The URI of the service	This is a fundamental aspect of RDF.

	business domain) and a version number.		
DescriptionDateTime	DescriptionDateTime defines when the description of the service was provided.	Not mapped	This would be covered using the provenance vocabulary [PROV-O]
Status	Status describes the status of the service.	Not mapped	It is interesting to note that SPOCS offers a set of 4 possible values for this: prepared, active, depreciated, terminated. adms:status could readily be used to provide this data about a service.
ValidityStart	ValidityStart defines the starting date where the service is available.	dcterms:PeriodOfTime linked from the service using dcterms:temporal	CPSV uses Dublin Core terms to describe the temporal limits of a service.
ValidityEnd	ValidityEnd defines the ending date where the service is not available anymore.		
Territory	Territory is the area where the service is available and legally usable.	dcterms:Location linked from the service using dcterms:spatial	CPSV uses Dublin Core terms to describe the geographic limits of a service.
Synonyms	Synonyms are different wordings for the same meaning of the service.	Not mapped directly.	It is notable that CPSV strongly urges service descriptions to provide a service type using a controlled vocabulary. Use of such lists, which should be multilingual, will fulfil the role foreseen for SPOCS' Synonyms.
Keywords	Keywords are wordings that are relevant to the service.	Not mapped	dcat:keyword is available for use in this manner.
BusinessEvent	A certain stage in the business lifecycle with which a bundle of services is associated with. The BusinessEvent represents a certain group of needs of the SPs related to their activities.	Not mapped	
Profession	A profession describes the related business domains for the service. It is directly related to the scope description.	Not mapped	This kind of information can be included as part of a cpsv:Rule.

In section 4.1 we describe an Excel spreadsheet that was created for use when creating service description using the CPSV. The above mapping provides the means to transfer data from the SPOCS service descriptions to that spreadsheet as we will see below.

3.2 Flemish Intergovernmental Product and Service Catalogue (IDPC)

The Coördinatieceel Vlaams e-government, Corve, is close to completing a catalogue of public services available in Flanders. Their work references SPOCS² and at the time of this pilot study a number of service descriptions have already been made available as XML files in a test environment. Full details of the project are available at <http://www.corve.be/projecten/lokaal/IPDC/> and the UML diagram of their data model is included as Figure 6.

What we're concerned with specifically is how well that work matches the CPSV. As with the SPOCS case, the IPDC provides a series of attributes of the service itself, most of the classes are actually enumerations, i.e. controlled lists of string values, so it's a largely 2-dimensional model. Mapping from that model to the CPSV - which is multidimensional - therefore requires more than simple term substitution. Also, the pilot is concerned with trialling the exposure of service descriptions as linked data so there is a transformation to be made from mostly flat data into the graph structure used by RDF.

² <http://www.corve.be/docs/IPDC/IPDCInformatieanalyse.pdf>

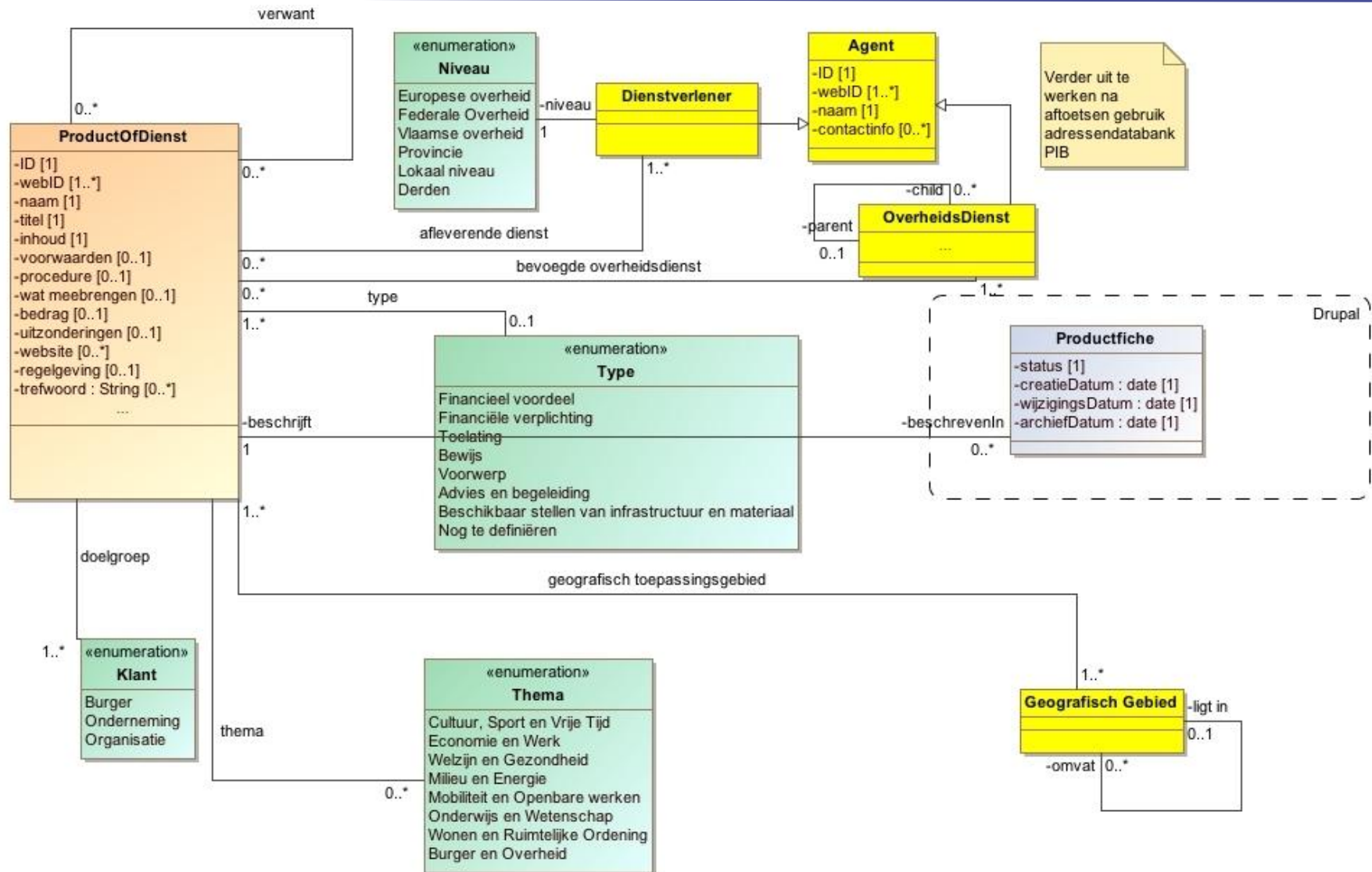


Figure 6 The UML Diagram Showing the IPDC Data Model

The following table summarises the mapping used.

IPDC term	CPSV term	Notes
naam	rdfs:label	Not part of the CPSV but easily included
titel	dcterms:title	
inhoud	dcterms:description	
voorwaarden	cpsv:Rule with a dcterms:title of 'voorwaarden' (NL) and 'eligibility conditions' (EN). The type of Rule, 'Precondition', is given as the appropriate URI from a SKOS concept scheme.	The eligibility conditions described for each service in the IPDC catalogue are rendered as a Rule with the text provided as a dcterms:description of the Rule.
procedure	cpsv:Rule with a dcterms:title of 'procedure' (in both NL and EN). The type of Rule, 'Process', is given as the appropriate URI from a SKOS concept scheme.	As above, the text describing the process is provided as a dcterms:description of the Rule.
watMeebrengen	cpsv:Input with a dcterms:title of ' watMeebrengen' (NL) and ' what to bring along' (EN).	The details of the inputs required are provided as the value of dcterms:description of the cpsv:Input.
bedrag	cpsv:Input with a dcterms:type of 'Fee' and dcterms:title of 'bedrag' (NL) and 'fee' (EN). The type of Input, 'Fee', is given as the appropriate URI from a SKOS concept scheme.	The actual fee is given as the value of a dcterms:description of the Input.
uitzonderingen	cpsv:Rule with a dcterms:title of 'uitzonderingen' (NL) and 'exceptions' (EN). The type of Rule, 'Exceptions', is given as the appropriate URI from a SKOS concept scheme.	Details of the exceptions are given as the dcterms:description of the Rule.
regelgeving	cpsv:implements with a dcterms:title of ' regelgeving' (NL) and 'regulations' (EN).	This aspect of the mapping is problematic - see text.

links	foaf:homepage	
bevoegdeOverheidsdiensten	Not mapped	Can be included in CPSV-based models using the <code>cpsv:playsRole</code> property
afleverendeDiensten	<code>cpsv:provides</code>	
doelgroepen	Not mapped	
geografischeToepassingsgebieden	<code>dcterms:spatial</code> linked to a <code>dcterms:Location</code>	
productType	<code>dcterms:type</code>	See notes
trefwoorden	Not mapped in CPSV	<code>dcat:keyword</code> (or <code>radion:keyword</code>) may be used.
themas	Not mapped in CPSV	<code>dcterms:subject</code> may be used to for this.
defaultThema	Not mapped in CPSV	
verwanteProducten	<code>dcterms:related</code>	

The mapping above is largely straightforward. Where terms are used by IPDC that are not used in the CPSV, these can be encoded as the data modeller sees fit. The point of a core vocabulary is that it encourages the use of common terms for the most common data points but does not limit what can be expressed using other vocabularies. There is one aspect that does highlight a clash however.

The Core Public Service vocabulary makes a distinction between Rules and Formal Frameworks. The latter will be legislation or other such legal instruments (whether regional, national or European). Rules are typically set by the authority responsible for the provision of the service which implement the legislation. Rules will include details of the procedure, commitments on service timing etc. That's why the CPSV shows a relationship of `implements` between the Rule and Formal Framework classes.

In the 2 dimensional descriptions of services provided by SPOCS and in the IPDC, this relationship is not readily identifiable. Therefore to say in the table above that the `regelgeving` property maps to `cpsv:implements` is a simplification since the subject of the `implements` property must always be a Rule, not a service.

3.3 Irish Public Service Descriptions

The team at DERI is routinely involved with open data work in the Irish public sector and is an internationally recognised centre of excellence. As such, they were in a position to provide

descriptions of three public services which they did using a spreadsheet template provided as part of this pilot. Section 1.4.1 describes one of two spreadsheet templates that were created - one that assumes knowledge of the CPSV and familiarity with linked data vocabularies in general. A second template was created for non-technical users. This [non-technical spreadsheet](#) (Figure 7) was used by DERI.

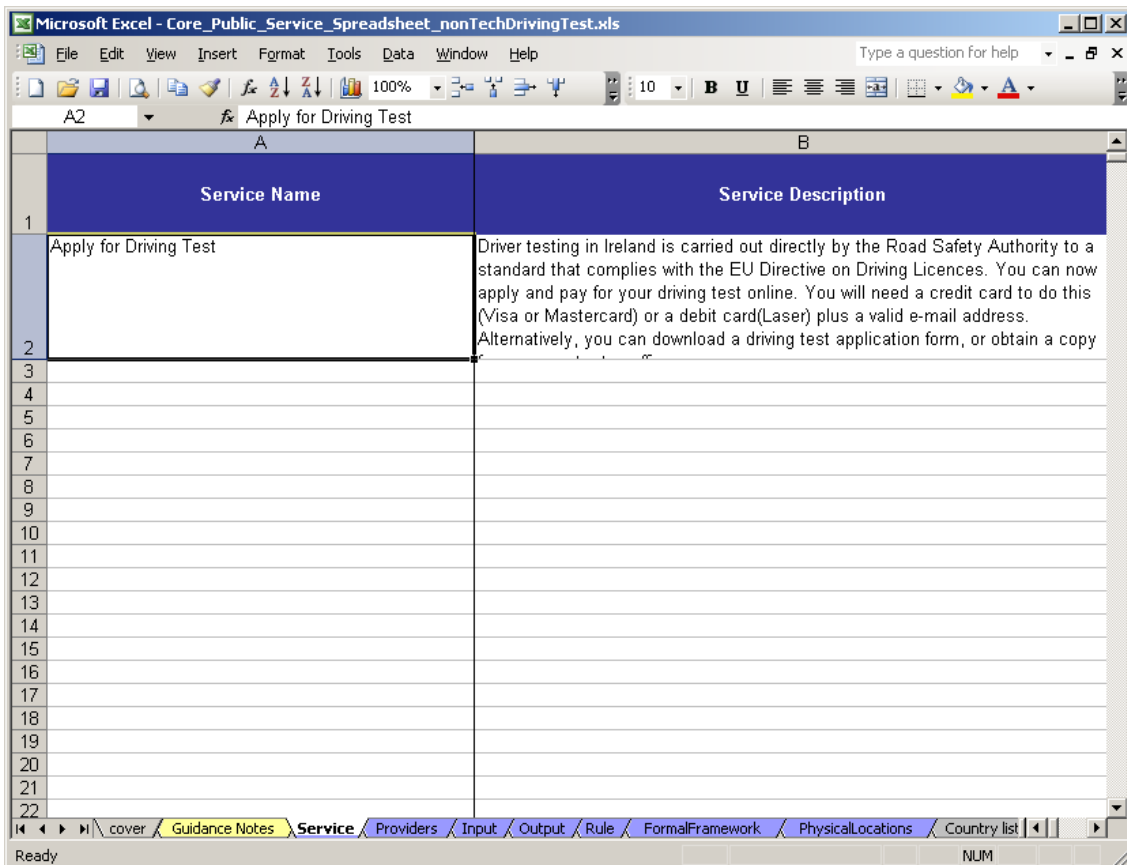


Figure 7 Partial screenshot of the non-technical spreadsheet template

The template attempts to guide users through the process and to automate certain aspects. For example, when a user adds text to the Inputs column on the first tab (worksheet), it is automatically copied to column A on the Inputs tab. Further details can then be provided *about the input*. This eliminates the need to enter the same data multiple times and therefore minimises the risk of making errors when doing so. We say minimise the risk, not eliminate the risk, since the spreadsheet was not protected and therefore it was possible to simply overwrite the formulae. This happened in the data received from DERI.

An important point to note is that DERI is not responsible for the actual service, that is, they're working at a distance from the service itself and so only know what they can discover publicly. This distance is what lies behind, for example, the value they give for the 'URL of formal document(s) describing how the service operates, rights and responsibilities etc.' namely http://europa.eu/rapid/press-release_MEMO-13-10_en.htm - which points to a press release. We will return to this point in the conclusion.



The data provided by DERI in the non-technical spreadsheet was then easily transferred to the more technical spreadsheet from which the RDF was generated as described in the following section.

4 Linked data infrastructure

This section contains a description of the publicly accessible linked data infrastructure that has been set up in the context of this pilot to illustrate how the business need use case identified in Section 1 can be met in practice.

4.1 Using Refine RDF to create service descriptions from a spreadsheet

This section shows how we generated RDF descriptions of public services from a spreadsheet using the [RDF extension](#) to [Google Refine](#) released by [DERI's Fadi Maali](#) and [Richard Cyganiak](#).

This can be done in four steps:

1. Use the [spreadsheet template](#) to describe your semantic standard;
2. Import your spreadsheet in [Google Refine](#);
3. Add the pre-defined mappings using the Google Refine [RDF extension](#) and CPSV [skeleton](#); and
4. Export to RDF using the Google Refine RDF extension.

1.4.1 Using the Spreadsheet template

The [spreadsheet template](#) contains all concepts and fields in the Core Public Service Vocabulary. To describe your public service fill in as much of the spreadsheet as possible.

	PublicService_URI	dcterms:title	dcterms:description	dcterms:type
1	/id/ltu/PublicService/ArchitectRegistration	Provision of Architectural Services on a Temporary Basis	The procedure of the recognition of professional qualifications of Architect pursuant to the occasional or temporary provision of architectural services in the Republic of Lithuania. Authorisation is needed for Architects from other EU member states for the provision of architectural Services on a temporary or occasional basis. Upon successful completion of the requirements, a list of Architects who can provide Architectural Services on a temporary or occasional basis will be publicly available on the site of Ministry of Environment of the Republic of Lithuania http://www.am.lt/VI/index.php#/1457 http://www.verslovertai.lt/en/regulations/provision-of-architectural-services-on-a-temporary-or-occasional-basis-in-lithuania/160585/ 	http://id.esd-toolkit.eu/service/1181
2	/id/ltu/PublicService/ArchitectRegistration			
3				
4	/id/ltu/PublicService/ArchitectRegistration			
5	/id/ltu/PublicService/ArchitectRegistration			
6	/id/ltu/PublicService/ArchitectRegistration			
7	/id/prt/PublicService/TravelAgentRegistration	Travel Agent Registration Service	The TA Service enables any travel agent to register, validate and certificate its activities by using the official TA service provided by the CA (Turismo de Portugal), in order to perform cross-border activities in Portuguese territory.	

Figure 8 Partial screenshot of spreadsheet template filled in with data from the SPOCS deliverable

The spreadsheet has been designed for use by individuals who are comfortable with linked data principles (see below for an alternative approach for less technical people). There are several tabs in the spreadsheet, one for each of the classes in the CPSV. In each case, the left hand column is a URI of the relevant class. Relative URIs are acceptable as these will be resolved against the base URI (see step 3 below).

The spreadsheet also contains a number of sheets with pre-defined controlled vocabularies, such as the ESD Toolkit Service List and various types of input and output. These sheets help you to select the right codes to use, but you should not update them.

1.4.2 Import Your Spreadsheet into Google Refine

Install [Google Refine](#) and the [RDF extension](#) on your system following the [installation guide](#). Create a new project in Google Refine and import the spreadsheet you created in step 1. The screenshot below indicates the desired configuration. In brief: check the boxes related to each of the coloured tabs in the spreadsheet and uncheck any other boxes except 'Parse next.'

is:title	dcterms:description	dcterms:type	foaf:homepage	cpsv:physicallyAvailableAt	dcterms:requ
on of ctural is on a rary or onal n ia	The procedure of the recognition of professional qualifications of Architect pursuant to the occasional or temporary provision of architectural services in the Republic of Lithuania. Authorisation is needed for Architects from	http://id.esd-toolkit.eu/service/1181	http://www.businessgateway.lt		

Worksheets to Import	<input type="checkbox"/> Ignore first	0 line(s) at beginning of file	<input type="checkbox"/> Store bla
<input type="checkbox"/> cover 32 rows	<input checked="" type="checkbox"/> Parse next	1 line(s) as column headers	<input type="checkbox"/> Store bla
<input checked="" type="checkbox"/> cpsv_PublicService 63 rows	<input type="checkbox"/> Discard initial	0 row(s) of data	<input type="checkbox"/> Store file (file nam in each r
<input checked="" type="checkbox"/> foaf_Agent 21 rows	<input type="checkbox"/> Load at most	0 row(s) of data	
<input checked="" type="checkbox"/> cpsv_Input 45 rows			
<input checked="" type="checkbox"/> cpsv_Output 10 rows			
<input checked="" type="checkbox"/> cpsv_Rule 32 rows			
<input checked="" type="checkbox"/> cpsv_FormalFramework 9 rows			
<input checked="" type="checkbox"/> dcterms_Location 12 rows			
<input checked="" type="checkbox"/> locn_Address 30 rows			
<input type="checkbox"/> ServiceType 1371 rows			
<input type="checkbox"/> InputType 12 rows			

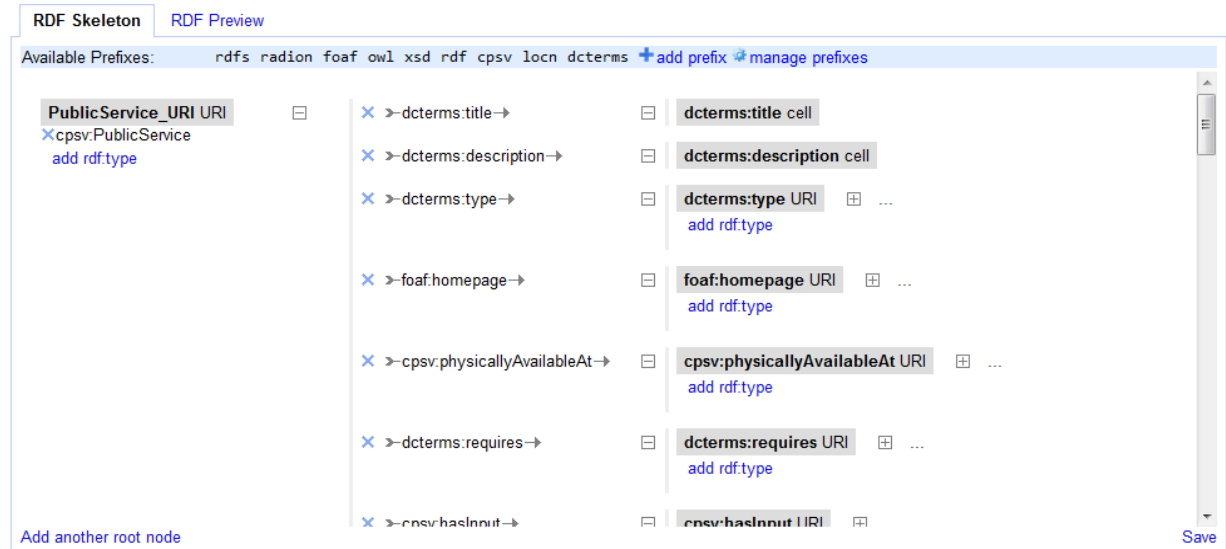
Figure 9 The Google Refine Configuration

1.4.3 Add the Pre-defined Mappings

The Google Refine RDF extension allows you to determine the intended structure of the RDF data by drawing a template graph. You can manually edit the RDF Skeleton by clicking "RDF" > "Edit RDF Skeleton..." in Refine.

The RDF schema alignment skeleton below specifies how the RDF data that will get generated from your grid-shaped data. The cells in each record of your data will get placed into nodes within the skeleton. Configure the skeleton by specifying which column to substitute into which node.

Base URI: <http://cpsv.testproject.eu/spocs> [edit](#)



The screenshot shows the 'RDF Skeleton' editor interface. At the top, there are tabs for 'RDF Skeleton' and 'RDF Preview'. Below the tabs, there is a header for 'Available Prefixes' with a list of prefixes: rdfs, radion, foaf, owl, xsd, rdf, cpsv, locn, dcterms, and buttons for '+ add prefix' and 'manage prefixes'. The main area is divided into two columns. The left column contains root nodes: 'PublicService_URI URI' (with a minus icon), 'cpsv:PublicService' (with a plus icon and 'add rdf.type' link), and 'add rdf.type'. The right column contains property nodes: 'dcterms:title cell', 'dcterms:description cell', 'dcterms:type URI' (with a plus icon and 'add rdf.type' link), 'foaf:homepage URI' (with a plus icon and 'add rdf.type' link), 'cpsv:physicallyAvailableAt URI' (with a plus icon and 'add rdf.type' link), 'dcterms:requires URI' (with a plus icon and 'add rdf.type' link), and 'cpsv:hasInput URI' (with a plus icon). In the center, there are several mappings: '>dcterms:title->', '>dcterms:description->', '>dcterms:type->', '>foaf:homepage->', '>cpsv:physicallyAvailableAt->', '>dcterms:requires->', and '>cpsv:hasInput->'. At the bottom left, there is a button 'Add another root node'. At the bottom right, there is a 'Save' button.

Figure 10 Editing the RDF Skeleton

To save you the trouble of adding a mapping from the spreadsheet to the CPSV, you can re-apply the operation history used in the pilot. The operation history contains all operations previously carried out to build the RDF Skeleton for the spreadsheet template. To apply the operation history:

- Click “Undo/Redo” > “Apply...”
- Paste the Operation History from [this text file](#) (the skeleton)
- Click “Perform Operations”.
- Click “RDF” > “Edit RDF Skeleton...” and verify that the RDF Skeleton has been created from the Operation History.

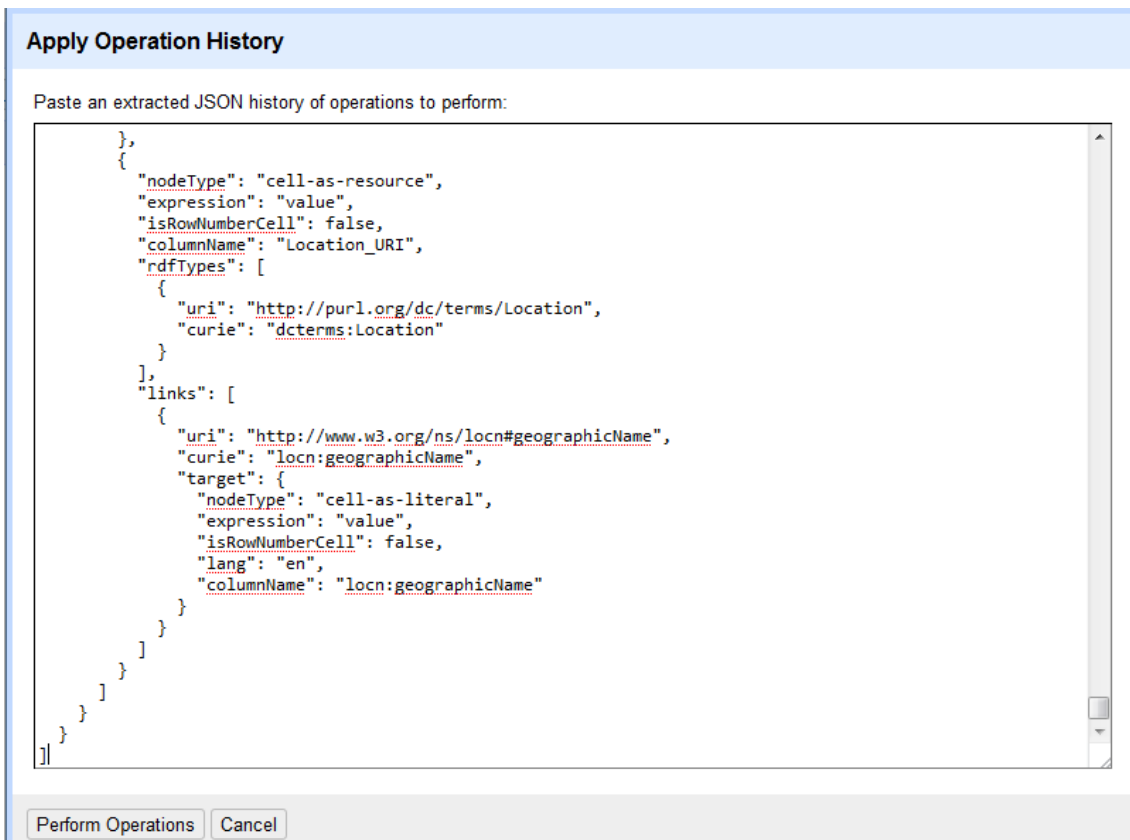


Figure 11 Applying the Operation History

1.4.4 Export the RDF

The final stage is to export the RDF that will have been generated by the tool. To do this, click the Export button on Refine's top right hand corner and select either RDF/XML or Turtle.

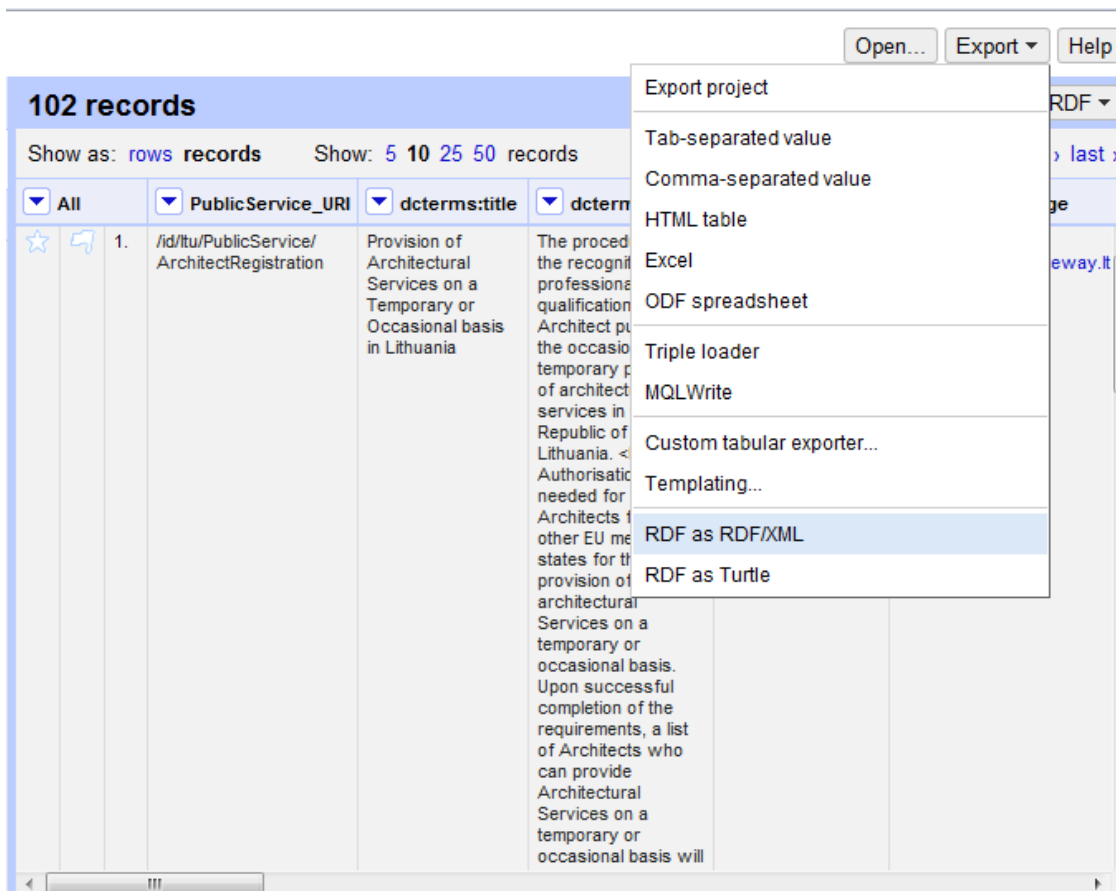


Figure 12 Exporting the RDF

RDF/XML and Turtle can both be validated using Joshua Tauberer's RDF Validator and Converter³. RDF/XML can be validated and visualised using the W3C Validator⁴.

As we noted in section 1.4.1, the spreadsheet is intended for use by individuals who are familiar with the concepts around linked data. An [alternative spreadsheet](#) has also been prepared that makes a lower demand on the user. This spreadsheet automatically takes text entered in the first worksheet and copies it where relevant in the others. Users are not asked to enter any URIs except the URL of the service homepage. Following the kind of approach taken in another recent pilot that focussed on the Organisation Ontology/RegOrg⁵, this spreadsheet template assumes a more powerful RDF Skeleton that will auto-generate URIs based on the input text. European Public Administrations wishing to pursue this approach are invited to contact the authors for more information and assistance.

³ <http://www.rdfabout.com/demo/validator/>

⁴ <http://www.w3.org/RDF/Validator/>

⁵ http://joinup.ec.europa.eu/asset/core_business/document/organization-ontology-pilot-linking-public-sectors-organisational-data

4.2 Using XSLT to transform XML descriptions into CPSV RDF

In section 3.2 we considered the mapping between the terms used in the Flemish IPDC (Interbestuurlijke Producten en Dienstencatalogus) service descriptions and the CPSV. At the time of writing a small number of service descriptions have been made available as discrete XML instances. As part of this pilot, an XSLT⁶ was prepared that automatically transforms those descriptions into the Core Public Service vocabulary (CPSV RDF representation). The XSLT can be run using an XSLT processor such as Apache XALAN, XT, or Saxon. Alternatively, it can be run in the Opera Web browser⁷. To do this, take any of the XML instances considered to the pilot, and the XSLT, which are available online at https://joinup.ec.europa.eu/svn/core_public_service/pilot/IPDC/. Insert

```
<?xml-stylesheet type="text/xsl" href="IPDC-2-CPSV.xsl"?>
```

Just below the opening `<?xml version="1.0" encoding="utf-8"?>`, save the XML file and then open it in Opera. The transformed content - the RDF - will be shown.

Using the XML data from the IPDC does have some drawbacks. In some cases the mapping is an easy 1 - 1 match, but in many others, the XSLT is forced to create a blank node in the RDF graph and then just give the text as a description of that node. There is no option within this process to create and use URIs for individual instances of the Rule class, agents involved in providing or using the service etc. Transforming data from one model to another usually causes some loss of data integrity and this is the case here too. Nevertheless, the ease with which the data can be transformed is notable.

4.3 Using OpenLink Virtuoso as a triple store and SPARQL endpoint

We have set up a Linked Data Infrastructure using the open-source version of **OpenLink Virtuoso** [Virtuoso]. Figure 13 contains a simplified representation of the relevant parts of Virtuoso's system architecture that were used in this pilot.

⁶ The XSLT script is available here
https://joinup.ec.europa.eu/svn/core_public_service/pilot/IPDC/

⁷ <http://www.opera.com/>

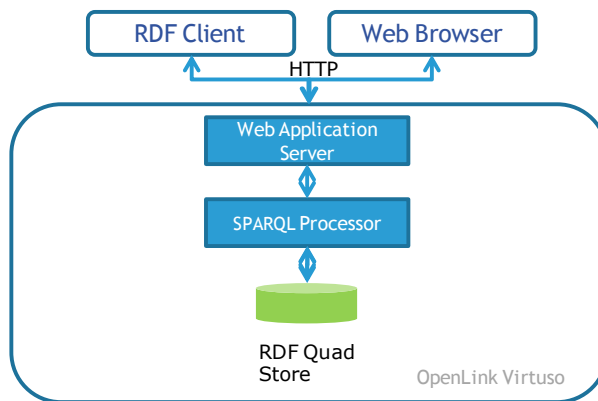


Figure 13 – Representation of Virtuoso’s Linked Data system architecture

4.4 Use case – Querying the triple store

Loading the test data into a triple store and providing a SPARQL endpoint means that any number of queries can be made against the data. Some examples follow. In each case, the query can be simply copied and pasted into the query text box at <http://cpsv.testproject.eu/sparql>, then press '.Run Query' to get the output shown.

4.4.5 List all public services described in the data

This simple query will list all the services described in the triple store and give their titles (which may be present but are not assumed to be).

```

PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX cpsv: <http://purl.org/vocab/cpsv#>
PREFIX dcterms: <http://purl.org/dc/terms/>

SELECT DISTINCT ?s ?serviceTitle
FROM <http://cpsv.testproject.eu/CPSV>

WHERE {?s rdf:type cpsv:PublicService .
  OPTIONAL {?s dcterms:title ?serviceTitle }
}
  
```

Running this query yields the output shown below

s	serviceTitle
http://cpsv.testproject.eu/id/ltu/PublicService/ArchitectRegistration	"Provision of Architectural Services on a Temporary or Occasional basis in Lithuania"@en
http://cpsv.testproject.eu/id/prt/PublicService/TravelAgentRegistration	"Travel Agent Registration Service"@en
http://cpsv.testproject.eu/id/prt/PublicService/TECRegistration	"Tourism Entertainment

	Companies Registration Service"@en
http://cpsv.testproject.eu/id/mlt/PublicService/RealEstateAgencyLicence	"Obtaining information and applying for a real estate licence to startup a real estate agency in Malta"@en
http://cpsv.testproject.eu/id/svn/PublicService/ArchitectRegistration	"Temporary entry in the Slovene directory of architects"@en
http://cpsv.testproject.eu/id/bel/PublicService/632	"Arbeidskaart voor loontrekkenden"@nl
http://cpsv.testproject.eu/id/bel/PublicService/117	"Premie 50+"@nl
http://cpsv.testproject.eu/id/bel/PublicService/159	"Eurovignet"@nl
http://cpsv.testproject.eu/id/bel/PublicService/279	"Bodemattest"@nl
http://cpsv.testproject.eu/id/bel/PublicService/291	"Milieuvergunning"@nl
http://cpsv.testproject.eu/id/bel/PublicService/301	"Bedrijfstoeslag voor de landbouw"@nl
http://cpsv.testproject.eu/id/bel/PublicService/346	"Stedenbouwkundige vergunning (bouwvergunning)"@nl
http://cpsv.testproject.eu/id/bel/PublicService/347	"Logiespremie"@nl
http://cpsv.testproject.eu/id/bel/PublicService/415	"Aanmoedigingspremie voor tijdscrediet of loopbaanonderbreking"@nl
http://cpsv.testproject.eu/id/bel/PublicService/263	"Opleidingscheques voor werknemers"@nl
http://cpsv.testproject.eu/id/bel/PublicService/586	"Test: Vergunning tot het rijden in een voetgangerszone"@nl
http://cpsv.testproject.eu/id/mlt/PublicService/CateringEstablishmentLicence	"Obtaining information and applying for a licence to startup a catering establishment in Malta"@en
http://cpsv.testproject.eu/id/bel/PublicService/34	"Eco-efficiëntiescan"@nl
http://cpsv.testproject.eu/id/bel/PublicService/306	"Softwarepakket voor het berekenen van minder afval, meer bedrijfsopbrengsten (MAMBO)"@nl
http://cpsv.testproject.eu/id/irl/PublicService/DrivingTest	"Apply for Driving Test"@en
http://cpsv.testproject.eu/id/irl/PublicService/EuropeanHealthInsuranceCard	"Apply for your European Health Insurance Card"@en
http://cpsv.testproject.eu/id/irl/PublicService/PublicServicesCard	"Apply for the Public Services Card"@en

The query can easily be modified to include, for example, only titles that are in Dutch:

```
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX cpsv: <http://purl.org/vocab/cpsv#>
PREFIX dcterms: <http://purl.org/dc/terms/>

SELECT DISTINCT ?s ?serviceTitle
FROM <http://cpsv.testproject.eu/CPSV>
```

```
WHERE {?s rdf:type cpsv:PublicService .
  OPTIONAL {?s dct:terms:title ?serviceTitle }
  FILTER (lang(?serviceTitle) = "nl")
```

returns

s	serviceTitle
http://cpsv.testproject.eu/id/bel/PublicService/632	"Arbeidskaart voor loontrekkenden"@nl
http://cpsv.testproject.eu/id/bel/PublicService/117	"Premie 50+"@nl
http://cpsv.testproject.eu/id/bel/PublicService/159	"Eurovignet"@nl
http://cpsv.testproject.eu/id/bel/PublicService/279	"Bodemattest"@nl
http://cpsv.testproject.eu/id/bel/PublicService/291	"Milieuvergunning"@nl
http://cpsv.testproject.eu/id/bel/PublicService/301	"Bedrijfstoelage voor de landbouw"@nl
http://cpsv.testproject.eu/id/bel/PublicService/346	"Stedenbouwkundige vergunning (bouwvergunning)"@nl
http://cpsv.testproject.eu/id/bel/PublicService/347	"Logiespremie"@nl
http://cpsv.testproject.eu/id/bel/PublicService/415	"Aanmoedigingspremie voor tijds krediet of loopbaanonderbreking"@nl
http://cpsv.testproject.eu/id/bel/PublicService/263	"Opleidingscheques voor werknemers"@nl
http://cpsv.testproject.eu/id/bel/PublicService/586	"Test: Vergunning tot het rijden in een voetgangerszone"@nl
http://cpsv.testproject.eu/id/bel/PublicService/34	"Eco-efficiëntiescan"@nl
http://cpsv.testproject.eu/id/bel/PublicService/306	"Softwarepakket voor het berekenen van minder afval, meer bedrijfsopbrengsten (MAMBO)"@nl

4.4.6 Finding Services of a Given Type in a Given Country

Perhaps more interesting is a query that returns details of services of a particular type in a particular country. The test data is small so for demonstration purposes we'll pick something that we already know is available.

Imagine we want to know how to register as an architect in Lithuania. We can look up Architect Registration in the ESD Toolkit service list [SL4].

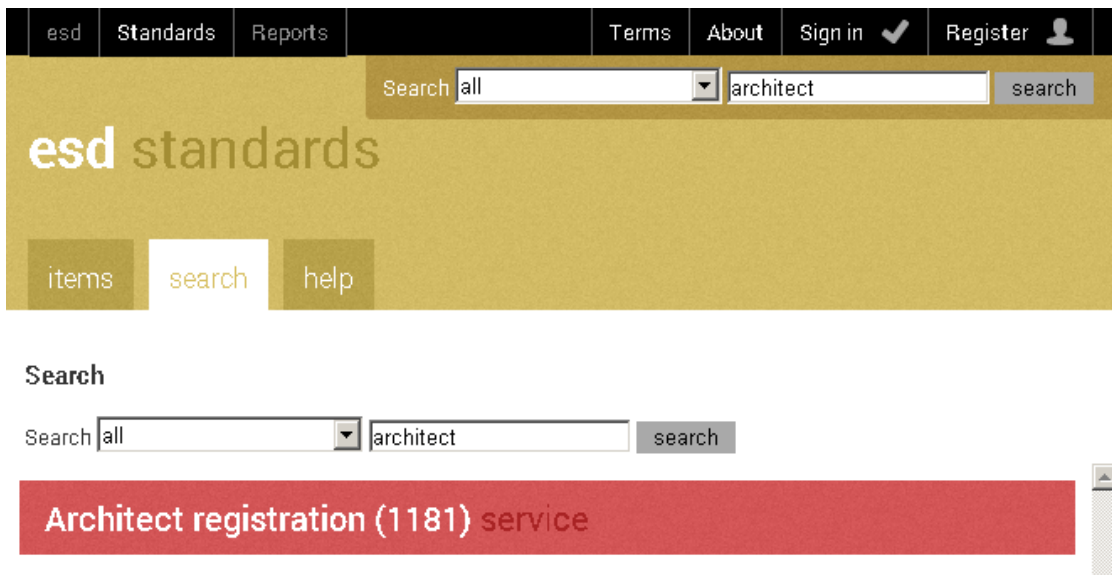


Figure 14 Searching for 'Architect' in the ESD Toolkit Service List [SL4]

Clicking through we come to the Linked data information about this service type and note its URI (Figure 15).

Armed with this information we can construct a query that looks for information about any service of this type:

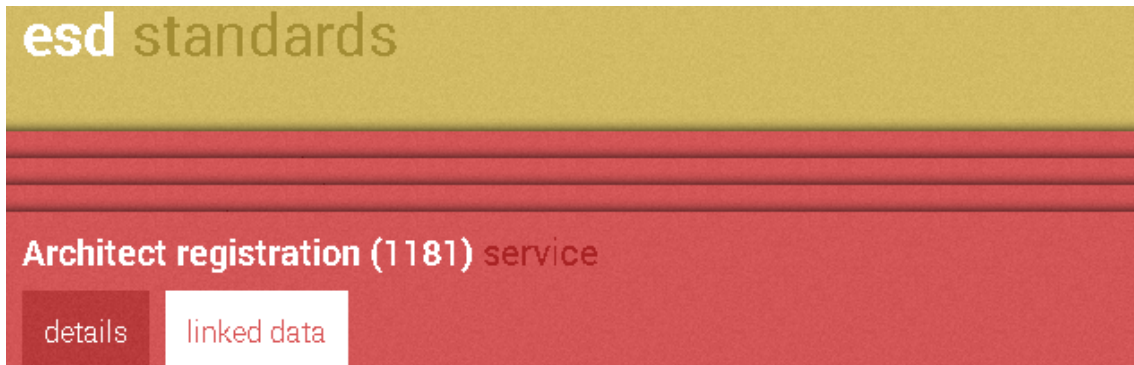
```

PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX cpsv: <http://purl.org/vocab/cpsv#>
PREFIX dcterms: <http://purl.org/dc/terms/>

SELECT DISTINCT ?s ?serviceTitle
FROM <http://cpsv.testproject.eu/CPSV>

WHERE {
    ?s dcterms:type <http://id.esd-toolkit.eu/service/1181> .
    OPTIONAL {?s dcterms:title ?serviceTitle }
}

```



Uri

<http://id.esd.org.uk/service/1181>

Property	Value
dc:identifier	1181
skos:prefLabel	Architect registration
dc:description	You need to be registered with the relevant organisation(s) in order to practice as an architect
dc:created	2013-04-03
dc:modified	2013-04-03
skos:historyNote	Added in version 3.07. Name changed to 'Architect registration' in version 4.00.

Figure 15 ESD's Linked Data Information Concerning Architect Registration

If we want to restrict our search specifically to Lithuania then we need to add in an extra term to the query. The CPSV uses `dcterms:spatial` to link to a URI that identifies a location. For European Public Administrations, the recommended URI set to use for this is the one made available by the Publications Office⁸. We can now restrict our search for architect registration services just to Lithuania thus:

```
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX cpsv: <http://purl.org/vocab/cpsv#>
PREFIX dcterms: <http://purl.org/dc/terms/>

SELECT DISTINCT ?s ?serviceTitle
FROM <http://cpsv.testproject.eu/CPSV>

WHERE {
    ?s dcterms:type <http://id.esd-toolkit.eu/service/1181> .
    ?s dcterms:spatial
    <http://publications.europa.eu/resource/authority/country/LTU>
    OPTIONAL { ?s dcterms:title ?serviceTitle }
}
```

⁸ <http://open-data.europa.eu/en/data/dataset/2nM4aG8LdHG6RBMumfkNzQ>

which, unsurprisingly within the pilot, gives us

s	serviceTitle
http://cpsv.testproject.eu/id/ltu/PublicService/ArchitectRegistration	"Provision of Architectural Services on a Temporary or Occasional basis in Lithuania"@en

A SPARQL editor at <http://cpsv.testproject.eu:8080/sparqled/sindice-editor/> can be used to create many other queries. Such tools become increasingly valuable as the amount of available data and complexity of queries increases.

4.5 Use case – Looking up a service description

By means of URL rewrite rules, Virtuoso also deploying a URI infrastructure that adheres to the *content negotiation* design principles of the HTTP protocol and related design guidelines for Linked Data URIs included in among others the following recommendations: the 2008 W3C Semantic Web Interest Group Note entitled '[Cool URIs for the Semantic Web](#)' [CoolURIs] and the 2012 study '[10 rules for persistent URIs](#)' [PURI] developed by the ISA Programme.

An essential part of HTTP URIs is that they identify abstract representations of real-world objects. For example, the following URIs identify some of the services described in earlier sections:

<http://cpsv.testproject.eu/id/mlt/PublicService/RealEstateAgencyLicence>
<http://cpsv.testproject.eu/id/mlt/PublicService/CateringEstablishmentLicence>
<http://cpsv.testproject.eu/id/bel/PublicService/34>

The HTTP protocol provides a powerful mechanism – called content negotiation – to allow Web clients (e.g. a Web browser, an RDF application, an e-Government application) for offering different formats of the same Web document known as *content negotiation*. When de-referenced, URIs that identify real world objects other than Web documents should redirect using HTTP response code 303 to Web documents that describe the object. This should be done in a consistent manner that can be written as a **URI re-write rule**, typically replacing the URI {*type*} of 'id' with 'doc.' In our example, the above mentioned URIs should de-reference to the following URLs:

<http://cpsv.testproject.eu/doc/mlt/PublicService/RealEstateAgencyLicence>
<http://cpsv.testproject.eu/doc/mlt/PublicService/CateringEstablishmentLicence>
<http://cpsv.testproject.eu/doc/bel/PublicService/34>

The following HTTP request:

```
GET /id/mlt/PublicService/RealEstateAgencyLicence HTTP/1.1
Host: cpsv.testproject.eu
Accept: text/html
```

must therefore lead to the following HTTP response with a 303 redirect response code:

```
HTTP/1.1 303 See Other
Connection: close
Content-Type: text/html; charset=UTF-8
Date: Mon, 07 Jan 2013 22:23:06 GMT
Accept-Ranges: bytes
Location: http://cpsv.testproject.eu/doc/mlt/PublicService/RealEstateAgencyLicence
Content-Length: 0
```

An RDF client dereferencing the *same* URI will retrieve the rdf document without any redirection

```
GET /id/mlt/PublicService/RealEstateAgencyLicence HTTP/1.1
Host: cpsv.testproject.eu
Accept: application/rdf+xml
```

Resulting in the desired Web document describing the address object:

```
HTTP/1.1 200 OK
Connection: Keep-Alive
Date: Mon, 07 Jan 2013 22:35:16 GMT
Accept-Ranges: bytes
Content-Type: application/rdf+xml; charset=UTF-8
Content-Length: 1821
```

Error! Reference source not found. summarises the interaction when a client that accepts TML, i.e. a normal Web browser, requests the URI of the Maltese real estate agency licence service.

Via URL rewriting rules, the interaction shown in **Error! Reference source not found.** has been implemented in the pilot's linked data infrastructure. The request for resource

```
http://cpsv.testproject.eu/id/mlt/PublicService/RealEstateAgencyLicence
```

is internally handled as a simple SPARQL query

```
DESCRIBE <http://cpsv.testproject.eu/id/mlt/PublicService/RealEstateAgencyLicence>
```

of which the result is returned as an RDF+XML file with the following content:

```
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"
  xmlns:dct="http://purl.org/dc/terms/" xmlns:cpsv="http://purl.org/vocab/cpsv#"
  xmlns:ns4="http://www.w3.org/ns/radion#">
  <rdf:Description
    rdf:about="http://cpsv.testproject.eu/id/mlt/PublicService/RealEstateAgencyLicence">
```

```
<rdf:type rdf:resource="http://purl.org/vocab/cpsv#PublicService"/>
< dct:title xml:lang="en">Obtaining information and applying for a real estate
  licence to startup a real estate agency in Malta</dct:title>
<dct:spatial
  rdf:resource="http://publications.europa.eu/resource/authority/country/MLT"/>
<cpsv:hasInput rdf:resource="http://cpsv.testproject.eu/id/mlt/Input/Fee"/>
<cpsv:hasInput
  rdf:resource="http://cpsv.testproject.eu/id/mlt/Input/EmploymentEngagementForm"/>
<cpsv:hasInput
  rdf:resource="http://cpsv.testproject.eu/id/mlt/Input/PhotoOfApplicant"/>
<cpsv:hasInput rdf:resource="http://cpsv.testproject.eu/id/mlt/Input/CopyOfIDCard"/>
<cpsv:hasInput
  rdf:resource="http://cpsv.testproject.eu/id/mlt/Input/CompetenceCertificate"/>
<cpsv:hasInput
  rdf:resource="http://cpsv.testproject.eu/id/mlt/Input/RegisterEconomicActivityForm"
  />
<cpsv:produces
  rdf:resource="http://cpsv.testproject.eu/id/mlt/Output/RealEstateAgencyLicence"/>
<cpsv:follows
  rdf:resource="http://cpsv.testproject.eu/id/mlt/Rule/RealEstateAgencyLicence"/>
<ns4:keyword xml:lang="en">licence</ns4:keyword>
<ns4:keyword xml:lang="en">real estate</ns4:keyword>
<ns4:keyword xml:lang="en">property</ns4:keyword>
</rdf:Description>

<rdf:Description rdf:about="http://www.commerce.gov.mt/org">
  <cpsv:provides
    rdf:resource="http://cpsv.testproject.eu/id/mlt/PublicService/RealEstateAgencyLicence"
    />
  <cpsv:playsRole
    rdf:resource="http://cpsv.testproject.eu/id/mlt/PublicService/RealEstateAgencyLicence"
    />
</rdf:Description>
</rdf:RDF>
```

**Figure 16 Data returned to an RDF client dereferencing
<http://cpsv.testproject.eu/id/mlt/PublicService/RealEstateAgencyLicence>**

Via the same mechanism, an HTML representation can also be returned, as shown in Figure 17



About: Obtaining information and applying for a real estate licence to startup a real estate agency in Malta

An Entity of Type : PublicService,

References

Referenced By

- dct:spatial
 - <http://publications.europa.eu/resource/authority/country/MLT>
- dct:title
 - Obtaining information and applying for a real estate licence to startup a real estate agency in Malta
- cpsv:follows
 - Real Estate Agency Rules
- cpsv:hasInput
 - <http://cpsv.testproject.eu/id/mlt/Input/Fee>
 - Application to register a commercial activity
 - Employment engagement form
 - Two passport size photos of applicant
 - Copy of ID card
 - Competence certificate/document
- cpsv:produces
 - Real Estate Agency Licence
- type
 - cpsv:PublicService
- keyword
 - real estate
 - licence
 - property

Figure 17 Web page seen in a regular browser when visiting <http://cpsv.testproject.eu/id/mlt/PublicService/RealEstateAgencyLicence>

5 Conclusions and future work

We began work on the pilot with the following aims:

- to demonstrate that the Core Public Service can be used as a foundational RDF Vocabulary to **homogenise public service data** that originates from local, regional, and national e-Government portals;
- to demonstrate that the definition of uniform **HTTP URI sets** for public services facilitates information management;
- to demonstrate that the use of **common controlled vocabularies** can greatly facilitate the search for public services, especially in a multilingual context;
- to demonstrate that a **linked data infrastructure** can provide access to *homogenised*, *linked*, and *enriched* public service data;
- to demonstrate that the use of standard Web interfaces (such as HTTP(S) and SPARQL) can **simplify the use of public service data** for humans and machines.

The first point is demonstrated by the provision of three different sets of service descriptions that were created independently. In the case of the SPOCS and IPDC data it was possible to map from the vocabulary used to the CPSV. Whilst there was some loss of data along the way, it was not substantial and the benefits of the homogenise approach readily outweigh this loss. In the case of the Irish data, the CPSV model was used from the beginning.

The usefulness of HTTP URI sets, and the use of common controlled vocabularies is demonstrated particularly in section 4.4.6. Here we were able to use pre-existing URI sets for service types and countries in SPARQL queries that returned exactly the desired information. It was also trivially easy to specify the language in which we wanted results to be returned.

The linked data infrastructure means that when two parties wish to exchange information about a particular service, all they need exchange is the URI. All the data about the service can be accessed by the receiving party easily, whether it's wanted in machine or human readable format. Likewise if a public authority wishes to publish information about its public services, it can re-use the data prepared elsewhere without having to duplicate it. This efficient method of data exchange eliminates the risk of errors through duplication and minimises the danger of the data being out of date.

5.1 Known Limitations

Although the pilot shows how to achieve the aim things using real data about real public services, it would be false to suppose that we have reached utopia. Some weaknesses are apparent.

Firstly, filling in the Excel spreadsheet based on information available from SPOCS was a time consuming, non-trivial task. In the Flemish case, an XSLT does the job; but there is a trade-off here. The SPOCS data is more faithful to the CPSV data model than the auto-generated Flemish data. This is not just because the different data models use different terms - mapping from SPOCS ServiceDescription to dcterms:description is clearly trivial - it's that the two data

sets examined take a two dimensional approach or mostly two-dimensional approach. That is, all or most of the data supplied is about a single entity which is the service. The CPSV's Rule and Formal Framework classes are hard to match automatically although a human can spot instances of those classes and link them accordingly.

The Irish data was created by individuals who were not directly involved in the service provision itself. Despite being highly expert in linked data, when faced with a request for the URL of some legislation they entered one for a press release and it seems didn't fully understand the concept of CPSV's Rule class.

As a continuation of that theme, during the pilot, we have worked with data curated by service owners according to the data model they were given. The pilot has not been able to work directly with service providers to ascertain how easy it would have been for them to prepare the data directly, i.e. using the CPSV as the original data model.

The roles played by different agents varies from one data set to another and this is, at least in part, a reflection of the whole way in which a given country's public sector operates. In particular the relationship between the organisation responsible for a service and the organisation that actually delivers it on their behalf. This was a topic of discussion during the development of the CPSV.

Overall, the CPSV provides a good framework that can be used to describe public services and, when this is done, the target benefits are available. However, it is clear that creating service descriptions really requires two kinds of expert to work together: one who understands the service and another who understands the data model.

References

- [Business] Core Business Vocabulary
http://joinup.ec.europa.eu/asset/core_business/description
- [ESD] Effective Service Delivery (ESD) standards
<http://standards.esd.org.uk/>
- [FOAF] Friend of a Friend. Brickley D., Miller L., <http://xmlns.com/foaf/spec/>
- [FRBR] Functional Requirements for Bibliographic Records, International Federation of Library Associations and Institutions (IFLA), 1998
<http://www.ifla.org/publications/functional-requirements-for-bibliographic-records>
- [ISA] Interoperability Solutions for European Public Administrations,
<http://ec.europa.eu/isa/>
- [LinkedData] Tim Berners-Lee, Linked Data Design Issues, 2006.
<http://www.w3.org/DesignIssues/LinkedData.html>
- [Location] Core Location Vocabulary
http://joinup.ec.europa.eu/asset/core_location/description
- [LOU] Loutas N., Peristeras V., Tarabanis K., The Public Service Ontology: A formal model for describing domain specific semantics. In International Journal of Metadata, Semantics and Ontologies, vol. 6 (1), p. 23-34, 2011.
- [ORG] An organization ontology, Dave Reynolds/W3C,
<http://www.w3.org/TR/vocab-org/>
- [PER] Peristeras V. and Tarabanis K. (2008) "The Governance Architecture Framework and Models", chapter in Pallab Saha (eds.) "Advances in Government Enterprise Architecture" Hershey, PA: IGI Global Information Science Reference.
- [Person] Core Person Vocabulary
http://joinup.ec.europa.eu/asset/core_person/description
- [PROV-O] PROV-O: The PROV Ontology, T Lebo, S Sahoo, D McGuinness. W3C Recommendations Track document <http://www.w3.org/TR/prov-o/>
- [PURI] D7.1.3 - Study on persistent URIs, with identification of best practices and recommendations on the topic for the MSs and the EC
<https://joinup.ec.europa.eu/community/semic/document/10-rules-persistent-uris>
- [RDFPrimer] RDF Primer <http://www.w3.org/TR/rdf-primer/>

[SPOCS-D5.13] SPOCS deliverable D5.13 Population of the Services Directory with Information Related to the New Professions

http://www.eu-spoocs.eu/index.php?option=com_processes&task=showProcess&id=18&Itemid=61

[Service] Core Public Service

https://joinup.ec.europa.eu/asset/core_public_service/description

[SL4] ESD Toolkit Service List 4, Available at:

<http://doc.esd.org.uk/ServiceList/4.00.html>

[StratML] Under development at AIIM (Association for Information and Image Management), Available at:

<http://www.aiim.org/Research-and-Publications/Standards/Committees/StratML>

[TBL] Tim Berners-Lee, Cool URIs don't change, 1998

<http://www.w3.org/Provider/Style/URI.html>

[Virtuoso] OpenLink Virtuoso

<http://docs.openlinksw.com/virtuoso/>

[UKURI] Designing URI Sets for the UK Public Sector, Davidson, P. UK Chief Technology Officer Council

https://www.gov.uk/government/uploads/system/uploads/attachment_data/file/60975/designing-uri-sets-uk-public-sector.pdf

[URLrewrite] Virtuoso Linked Data deployment Guideline

<http://virtuoso.openlinksw.com/whitepapers/VirtDeployingLinkedDataGuide.html>

Acknowledgements

Specific acknowledgement is due to:

Name	Organisation
Muriel Foulonneau	Henri Tudor Research Centre
Joseph Grech	Business First
Ana Branco	AMA - Agência para a Modernização Administrativa, IP
Nemunas Karalius	iTree company , Lithuania
Gediminas Grazis	Verslo vartai , Point of single contact in Lithuania
Adegboyega Ojo	DERI , Digital Enterprise Research Institute
Katrien De Smet	CORVE , Flemish e-Government Coordination Unit
Raf Buyle	V-ICT-OR , Flemish ICT Organisation