# The reuse of external existing vocabularies appearing in a catalogue, not managed by your organization. Present the pros and cons of relying on vocabularies.

Pros[[1]](#footnote-1):

* Semantic interoperability is enabled. This is well explained in the latest version of EIRA, see the Key Interoperability Enablers View: Ontologies are Key IOP Enablers in the semantic layer. For the Enablement to be activated the ABB (ontology) is to be ‘Shared and Reused’ (Sharing and Reuse are the two critical events resulting into components that can be considered Key IOP Enablers🡪The Structural Interoperability Saliency is defined in terms of ‘Sharing and Reuse’);
* Modularity is promoted in front of monolithic solutions. Modularity is the architectural approach making possible the Structural Interoperability. The more modular the more reusable;
* Openly sharing modular components promotes the standardisation of semantic artefacts. The (re)use of open standards is one important recommendation in the EIF and a critical factor for the enablement of interoperability. One major benefit of open standards is the proliferation of compatible solutions that are Vendor-independent;
* The reuse of standard vocabularies and ontologies provide the opportunity to consume existing and up-to-the-date datasets. These data are invaluable assets for the implementation of the Public Service Policy and the Digital Public Service Delivery.
* The reuse of open standard domain-agnostic semantic artefacts, e.g. ontological and linguistic resources, allows policy, business and technical implementers to focus on the domain problems to be solved. This also Reusing reduces the cost of maintenance and evolution and boosts the time-to-market;
* If the drafting of ‘the catalogue’ of reusable vocabularies is based on sound governance principles. One of such ‘sound’ governance principles should be that the vocabularies to reuse need to be developed and maintained by a stable SSO, SDO[[2]](#footnote-2) or Community ensuring the open access to their governance processes and developments. These organisations have special interest in releasing backwards compatible artefacts and solutions, which impinges directly the business and digital continuity. These principles would ensure the long-term sustainability of the Public Services and would justify the investment in their development.

Cons:

* Dependency
* Deficiencies in the documentation/formats available;
* If it is not well maintained: loss of information due to an imported ontology that is not available (validation errors);
* License unavailable;
* …

# It is not clear which is the target audience of this deliverable apart from the Semic team. In case where the target audience in broader including:

# semantic experts that want either to use as basis our core vocabularies and extend them (i.e., create their own application profiles) or use the core vocabularies in the general content of designing their own models (for example Flemish government that wants to implement the change of address procedure of SDG)

* 1. Policy makers. In that case, Cecile mentions that, “*the design approach takes for granted the existence of a core vocabulary in a reverse engineering fashion.  
     Design Approach refers to Step One and Two (where are they defined, I could only find them in the word doc). This is about the documentation of the process to create this guideline. But not about the design approach of a vocabulary.  
     There is a lack of explanation of the driving need for a (core) vocabulary. and the process for managing and using it.*” She proposes to make a chapter that documents the approach to create the guideline in the history for this document and then focus on the product that is needed to serve your target audience. Usually people are NOT interested in how you were thinking but WHETHER the guideline can HELP them and on HOW they can use it.

1. Taking under consideration the target audience, we need to clarify which document is appropriate for each profile. By the term document, I refer to the Handbook and the style guide. By the term profile, I mean semantic technology developers as well as policy makers, so that definitions should be added appropriately. Under this scope, Cecile suggests the following: “[*https://github.com/ecobosco/SEMICguidelines/wiki/Need*](https://github.com/ecobosco/SEMICguidelines/wiki/Need) *explains that the guideline is to be used to build e-government core vocabularies. It does not explain what it is about. The scope should be more clear. It should also be consistent with the target audience.  
   There is a general confusion between the narrative of the task and the narrative around the deliverable itself. The deliverable should be independent of the task that produces it. This harms the potential reach of the guidelines.* She suggests of “*providing a definition for e-Government core vocabularies. But also, I am not sure that this guideline should be limited to "core" vocabularies. What in these guidelines is specific to core and to SEMIC? Or for developing vocabularies and application profiles for use by public administrations and offering a clear approach for synthesis of various knowledge representations.".*
2. The stress on the target audience is strongly related to the maximization of its reusability by reducing the semantical incoherencies. However, reuse needs further clarification. More specifically:
   1. It is unclear what is normative information and what is non-normative information, and how the reuse should be done. In the end, one must have a clear understanding of *compliance with the core vocabulary/application profile*. This perspective is not clear in the style guide.
   2. At the level of a vocabulary (terms with definitions), the URIs are the persistent identifiers for a term, its label and its definition.
   3. It is unclear in the style guide whether additional constraints like domain and range are part of the definition or not, since according to the style guide they are in a separate file. However, questions like the following ones arise: Are they non-normative? Is rdfs:subclassOf/rdfs:subPropertyOf then also non-normative? Because it binds the term with a larger term, and thus to the domain & range constraints of that larger term.
   4. Another issue has to do with file /folder organization (splitting) in terms of modularization (or semantical perspective/scope). Since the aim is to model implementation supporting models rather than reusable assets, the current practice is to put all information in a single file taking under consideration the fact that most users in the semantic web apply cherry picking: they select parts from the data models, ignoring the scope of the machine-readable distribution. For that, the notions Core Vocabulary & application profile should be clarified, with the proposed modularization, and reuse approach. The current definitions in the text mostly illustrate the terminology, but do not define nor relate them with each other. E.g. why is Core Person a core vocabulary? And not an application profile? Why is DCAT-AP an application profile, and why is BREG DCAT-AP should not be a specialization of DCAT-AP, but a sibling? in Annex I, an draft description of the difference between a Core Vocabulary and an Application Profile is included.
3. Validation process with regards to definitional and assertional knowledge. Since,
   1. definitional knowledge is what is part of the definition (i.e., is an atomic unity and it should be reused as a whole)
   2. assertional knowledge restricts the possible worlds that can be constructed from the definitional knowledge to the cases that are valid according to the Core Vocabulary/Application Profile, that means that conformance to a Core Vocabulary/Application Profile is mainly based on the validation through the SHACL shapes.
   3. not all assertional knowledge is of the same level. There are mandatory/recommended and optional advices. Each of them can lead to one or more SHACL shapes. It is however not possible to add a uniform severity level to the messages. That depends on the usage context.

In terms of the style guide, the following concerns were expressed:

* What is definitional knowledge, and should be taken as one unity
* If another specification implements/is a specialization of the Core Vocabulary what should it reuse minimally?
* Compliance: is a specification compliant with the Core Vocabulary if it does not satisfy the rdfs:domain/rdfs:range statements?

1. Versioning

In the guideline, the URIs from the core vocabularies include version numbers. This creates a number of problems. Take the case of DCAT-AP. This means that if DCAT-AP has a new release every 6 months, then the URIs change every 6 months. The naming rules for the entities should be versionless. However artifacts such as documents and files should be versioned. Versionless URIs does not mean there is a life cycle management on the entities. That means that a definition of a term CANNOT be adapted unless it is an editorial change. If it implies a semantical change then it cannot happen. The existing term should be deprecated and replaced with a new term having the updated definition. Deprecated terms cannot be deleted for persistency reasons.

Term lifecycle management: create -> editorial changes -> deprecated.

A best practice for xml versioning can be found in <http://www.xfront.com/Versioning.pdf>

1. URIs Re-mint.

It is in general not a good idea to re-mint URIs that already exist. A recommendation to this is the note that Makx wrote some years ago to create redirections from the [data.europa.eu](http://data.europa.eu/) URIs to the [semic.eu](http://semic.eu/) ones. Also, URIs of the form [data.europa.eu/semanticassets/](http://data.europa.eu/semanticassets/).. cannot be assigned under the OP’s URI policy – and it is OP who assigns these first-level path elements; it’s not up to other DGs/services to define them. This supports the need to coordinate with the Publications Office and its practices.

1. SHACL distribution

The intend is of the SHACL distribution should be made more clear. Based on the assessment it seems it is a file documenting all constraints. Constraints like minCount 0 and maxcount \* are useless to take into a validation process because they only succeed. They never fail. So, adding them to the distribution creates only weight for the validation steps. Therefore, is the SHACL an aid for the community to help them detect errors in data exchanges or it is a machine readable description of all axioms in a specification?

Depending on the answer to the previous question, what should we do with mandatory/recommended/optional statements?

From the perspective of validation, the ISA testbed actually has machine generated messages, which are considered to be precise enough for a machine to machine data exchange. During the webinar on DCAT-AP SHACL, the community did not express the need for a SHACL distribution documenting the specification in a machine-readable way, but for a SHACL distribution that would assist them in detecting errors while being harvested by the EDP.

1. XML distribution
   1. Even though the effort to create a canonical version for an XML structure is highly appreciated, but one should be aware that this is the result of a series of choices. And how it should be used. In the XML, one incorporates the rdfs:domain/rdfs:range + SHACL rules, all in one file. So, by making this normative, the previous argumentation on the splitting in 2 definitional files is gone. So, anyone who is using the XML ecosystem is forced in practice to deal with the domain & range challenges. Note that XML distributions might severely change from one version of the specification to another. This impedance cannot be an argument to not incorporate a semantical change in a Core Vocabulary. It is up to the users of the XML structures to follow the updates.
   2. In the document there are XSD guidelines for a vocabulary and application profile. It is unclear if the XSD for an application profile based on a core vocabulary is just a further restriction. What is the relationship between them?
   3. Currently the mapping is unidirectional from Core Voc/Application Profile to XML. Is this XML reversible interpretable? Meaning, does there exist an automated transformation that would transform an XML according to these rules back to RDF. Having that, one can define compliance w.r.t. the Core voc and AP as the correct validation against the SHACL. The proposal is to make choices that would make the transition easier. E.g. use identifiers in the data (cfr codelists below).
   4. Naming conventions: in the examples the name is taken from the reference of the URI. If we reuse external terms like in DCAT-AP, the label is providing the first human interpretation, not the URI. So, it is the label that should be the start point of it all, but the definition associated with the assigned URI (either external or internal) should express the intend of that term. The match between the URI definition and the term label will decide if reuse of the external term is directly possible. This means that if the attribute term has label “explanation” and is mapped on dct:description, “explanation” should appear in the XML and not description.
2. Synergies/ Similarities/ Differences with other style guides/ Questions based on the Closure Meeting CCCEV with TOOP, e-Certis and e-Procurement stakeholders.

Taking under consideration the style guide created for the purposes of the e-procurement ontology, in the 21/7/2020 meeting with TOOP, e-Certis and e-Procurement stakeholders, *Natalie (OP) and Enric (Everis) discussed transforming the XML back to OWL (RDF). This is weird from the perspective of the Core Vocabulary as master document. We think we should not commit to ensuring that this loop: UML -> RDF -> XML -> PO OWL would work all the time. Because that would bind both the RDF and the XML to the PO setting. Even if this is possible, to a large extend, the core vocabulary should have all freedom. On a side note, OWL is serialized as RDF, why not from RDF -> PO OWL? In general, the style guide should make the Master Data management clear. Indeed, it starts from the UML as modeling environment, but I think we agree that the html document + the machine-readable distributions is the master of the specification. What is written in these documents are the rules to follow. So back to what is normative and what is not normative. There is no style guide for the human readable document, and that is a gap. Because that document will be used the most.*

1. Controlled vocabularies / Enumerations

Despite they play a large role in application profiles, no specific section has been made. In the TOOP meeting, Enric mentioned that the XMLs would include enumerations based on “strings”. Why do not use the PURIs? And ensure that an XML would share the URI instead of an embedded string.

On the quality assessment, maybe add minimal rules like:

* All artifacts should be parseable by a corresponding parser (html/RDF/XSD)

1. Impact of the style guide in already used vocabularies

The eJustice documents (see <https://github.com/SEMICeu/SDG-sandbox/issues/28>) are based on totally different XML style guides. It is unclear if an assessment of the impact of the style guide on them took place. Since eJustice XML documents are to be maintained by SEMIC then 2 collections of XML structures are present. It should be made clear if the new style guide is applicable to the eJustice documents. For the SDG, there is a risk here as if the xml message on the broker is according the new style guide (TOOP) but that the content of the documents is according to another scheme. Only in the setting where the eJustice documents are considered blackboxes for the SDG scheme this is workable.

1. Taking under consideration the following comment, to decide what to include in the style guide and what to the handbook:

*It should be structured around use cases: i.e.,*

1. *using an existing voc- how to read the specs*
2. *creating a core vocabulary*
3. *create a vocabulary using/extending a Core voc*
4. *creating an application profile of an existing vocabulary;*
5. *defining equivalences to a core voc.*

# Annex

## Annex I - Core Vocabulary and Application Profile

**Core Vocabularies are simplified, re-usable and extensible data models that capture the fundamental characteristics of an entity in a context-neutral fashion. (**[**source**](https://joinup.ec.europa.eu/collection/semantic-interoperability-community-semic/core-vocabularies#What%20are%20the%20Core%20Vocabularies)**, ISA)**

The important word here is ‘context-neutral’. In effect, a Core Vocabulary only defines the classes, their attributes and relationships between classes, without defining how those classes and properties should be used by applications. The terms in the vocabulary are like words in a dictionary: they just define what the classes and properties mean. The specification of a Core Vocabulary does not say anything about whether properties are ‘mandatory’ or ‘optional’ and do not put obligations on controlled vocabularies to be used as values for properties.

**Application Profiles are schemas which consist of data elements drawn from one or more namespaces, combined together by implementors, and optimised for a particular local application. (**[**source**](http://www.ariadne.ac.uk/issue/25/app-profiles/) **from 2000!)**

The important part here is “a particular […] application”. Application profiles define which classes and properties to use, what the cardinalities of the properties are, while respecting the overall structure of the vocabulary, in particular how classes are related. Any application profile is free to use as much of a vocabulary as it wants; there is no obligation to use all the properties that are defined in the vocabulary, nor is there a limitation on the use of any other classes and properties from other vocabularies to meet the requirements of the application. So the difference is that the Core Vocabulary is context-neutral, while the application profile is intended for a particular application context. In general, if you have a different application, you’ll need a different application profile.

A note on the ‘reuse’ of Application Profiles or building an Application Profile B that is compliant with’ another Application Profile A. This only makes sense if application B basically performs the same function as application B but needs some additional classes and/or properties. The example is a national application profile for open data portals: DCAT-AP-IT is an extension of the European DCAT-AP because it serves the same goal (aggregating and sharing metadata of public datasets) but just needs a couple of extra properties and controlled vocabularies that are only relevant for the Italian context. For the rest it complies with all the requirements (e.g. mandatory properties, cardinalities, controlled vocabularies etc.) defined for DCAT-AP. The nice thing in a Open-World, RDF-based context is that data compliant with DCAT-AP-IT can still interoperate with data compliant with the German profile DCAT-AP-DE if they are both compliant withy DCAT-AP – they will understand the information that is common, as defined by DCAT-AP, but they may not understand the specific Italian and German bits.

1. Specific comments regarding CCCEV implementation approach and assumptions

**Note that the following feedback regards only the CCCEV**

1. Design issues

On <https://github.com/SEMICeu/CCCEV/blob/CV-2.0.0/dcat-ap/2.0.0/xml/dcat-ap_semic-cbc.xsd>

* + Why DataServiceType has an attribute cbc:id and DatasetType not?
  + Where do the cbc attributes come from?

The structure should be transparently created from a specification.

For CCCEV 2.0, the way the XML handles Concepts and Values is very much focused on syntax; e.g. the differences between Concepts are expressed as differences in data types, rather than as differences in semantics. As far as we can see, this makes the RDF modelling awkward. In the past, this is the reason why we observed that, in most cases, CVs and APs start with an RDF model which is then reformulated as XSD/XML. In the case of CCCEV 2.0.0, we have the impression that it started with a model that was tailored for an XML environment and then try to reverse engineer RDF out of it. One additional difficulty is also that the use of [CBC](http://www.datypic.com/sc/ubl22/s-UBL-CommonBasicComponents-2.2.xsd-a.html)often implies a mix of semantics and syntax.

### DCAT-AP discussion (reuse)

There is a misunderstanding on what dcat-ap:dataset is. A dcat-ap:dataset is a collection of records that share the same purpose. E.g. the addressregister, the vistors to the swimmingpools, the corona infections. This is not an individual record. So not a photo, a birthcertificate nor an observation in a statistic dataset. Despite this interpretation is a valid interpretation according to DCAT, this is not the perspective for DCAT-AP. It is not the intent of the European Data portal (EDP) to aggregate the current descriptions with all addresses in Europe. That would swamp the EDP and makes that catalog useless.

So the following is an example of incorrect usage of dcat-ap:distribution

<https://github.com/SEMICeu/CCCEV/blob/CV-2.0.0/use_cases/espd-cv/ESPD-response-example_GeneralAverageTurnoverAggregatedValues.xml>

Because one can only refer to a collection (a file with all data) and not to an individual record. The problem here is that the reuse is done because technically the Catalog-Dataset-Distribution structure can be applied. And therefore, people think they can reuse the application profile DCAT-AP without reading the scope and intentions of the application profile.

If one reuses an AP outside the application scope of the AP, it is a risk one must be aware of.

### Usage of the CCCEV 2.0.0 branch

During the TOOP sync meeting it became clear that the branch cccev 2.0.0 is being used as the sandbox to explore the XML distribution. This is not the right place for that. Better is to create a separate repository and keep the branches of a core vocabulary related to the business discussion of the core vocabulary. This also has impacts on the examples and references in the style guide. If one refers to an example, then this example should be best part of the style guide, not part of a core vocabulary for which the location might change.

## Alignment of existing XML distributions in the context of the SDG

The TOOP project (and maybe SDG WP7) intends to use the XML structures as defined by Everis in <https://github.com/SEMICeu/CCCEV/tree/CV-2.0.0>. That means this project heavily depends on the Style guide.

1. <http://www.heppnetz.de/files/dke2008.pdf>

   <https://papers.ssrn.com/sol3/papers.cfm?abstract_id=3315457> [↑](#footnote-ref-1)
2. Standard Setting Organisation and Standard Definition/Development Organisation. [↑](#footnote-ref-2)