# Handbook and Style Guides

Different 'general comments' have led to the conclusion that distinguishing clearly between the goals, targeted audience and content of the '**eGovernment Core Vocabularies Handbook**' and the '**eGovernment Core Vocabularies Style Guides**' is urgently needed.

Subsections below reflect our opinion on this. In any case, the general comments are emerging as critical inputs for the scoping of the handbook content and the disambiguation between what should go in the handbook and what in the style guides.

## Handbook

### Audience

Policy makers, CIO, business owners and project leaders. Rationale for these targets is provided in farther sections below.

### Goals and content (TOP-DOWN approach)

1. Provide General Context:

* Alignment with the EU general principles and policy;
* Alignment with the European Commission principles and strategy;
* Alignment with the COM Interoperability strategy and principles, more concretely:
  + - References to the legal framework related to interoperability, standardisation and reusability (cfr. EIF Decision, Regulation 2015/2012, eur-lex texts on reusability, etc.);
    - Summary of ISA2 goals, actions and initiatives;
    - Summary of the SEMIC goals, works and road-map.

1. Provide Public Administration Interoperability-specific Context
2. The saliency of interoperability in the EU Digital Public Service Value Chain;
3. The role of the semantic interoperability;
4. The importance of sharing and reuse.
5. The need of a EU Catalogue of Reusable eGovernment Core Vocabularies:

* Explain and justify the need;
* Set the principles and recommendations for the identification and selection of reusable resources (cfr. the pros-and-cons comment, below);
* Illustrate with examples.

1. eGovernment Core Vocabularies:

* Definition
* What’s in ‘core’?
* Who should initiate them?
* When and how to develop a new eGovernment Core Vocabulary?
  + When: criteria to assess the convenience of initiating the development of a new eGCV;
  + How:
    - ‘the one-single-point of maintenance’
    - multi-syntax generation tooling (and what syntax to use and why)
    - recommended development life-cycle
    - other;
* Recap: ‘the story-line’ (illustrate the end-to-end process, including the identification of the need, roles, actions, groups and flows of tasks, etc.)
* The need of sharing new eGovernment Core Vocabularies (Sharing-by-design[[1]](#footnote-1))
* Principles and recommendations for the governance of eGovernment Core Vocabularies
* The Style Guides as an enabler of the semantic interoperability:
  + What are the Style Guides: introduction
  + How the Style Guides enable semantic interoperability
  + Why they should be systematically applied, by whom, under what circumstances (what for, etc.)

1. Application Profiles:

* Definition
* Who should initiate them?
* When and how to develop Application Profiles: story-line?
* Use Cases, examples
* Are APs shareable and reusable? When, how, by whom?
* Principles and recommendations for the governance of APs

1. Conformance and Compliance:
   * Disambiguation
   * Importance of conformance and compliance to ensure software compatibility and semantic interoperability;
   * Testing the conformance and the compliance: the importance of producing open reference implementations in front of complex conformance-testing solutions;
   * Should eGCV conformance and AP compliance be certified?
2. The Style Guides as an enabler of the semantic interoperability:

* What are the Style Guides;
* How the Style Guides enable semantic interoperability;
* Why they should be systematically applied, by whom, under what circumstances (what for, etc.)
* FAQ, e.g. how many applications profile may have an eGovernment Core Vocabulary? How many Public Administrations are reusing eGCV, what for, how, …? Which is the preferred syntax for the implementation of an eGCV by Public Administrations, and why? ...

## Style Guides

### Audience

Project leaders and technical implementers. Rationale for these targets is provided in farther sections below.

Add sections now missing, e.g. FAQ on technical aspects: What are the tools preferred by Public Administrations for the development of eGCV and APs? Are they open and/or free? Etc.

### Goals and content

The revision of the current content of the eGovernment Core Vocabularies Style Guides.

## Top-down reading order

For anyone interested in reading all the documentation, the recommended order of reading would be first the handbook and then style guides.

Both documents should refer one to another in different locations of the documents. When referring to a specific aspect of the other document, a rationale or justification should be provided. Also, a very specific indication of where the information referred to is located should also be supplied.

# COMMENT 1: Pros and Cons about the reuse of vocabularies

## General considerations:

Sharing and reuse do not apply only vocabularies, but any potentially reusable 'semantic interoperability solution' (i.e. ontological and non-ontological resources, including core vocabularies, controlled vocabularies, validations artefacts, documentation, etc.). This argument should be clearly stated and illustrated both in the handbook and the guides, since it is especially important for the ‘sharing-by-design principle’.

## Pros

Considerations on what could be included in the handbook follow. They are based on abundant literature on this topic, namely very recent works developed for COM[[2]](#footnote-2).

* **Semantic interoperability**: 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**: modularity is promoted in front of monolithic solutions. Modularity is the architectural approach making possible the Structural Interoperability. The more modular the more reusable;
* **Standardisation**: 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;
* **Data is the asset**: 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.
* **Focus and time-to-market**: 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 policy area and business domain problems to be solved. This also reduces the cost of maintenance and evolution and boosts the time-to-market;
* **Stability and sustainabililty**: 'Catalogues’ of reusable vocabularies are (to be) founded on sound governance principles. One of such ‘sound’ governance principles is that the vocabularies to reuse need to be developed and maintained by a stable SSO, SDO or Community ensuring the open access to their governance processes and developments. Also, these organisations have special interest in 1) releasing stable developments (i.e. provenly usable), and 2) ensuring backwards compatibility. This benefits directly the business and digital continuity, which are critical for the the long-term sustainability of the Public Services and the justification of the investment in their development and maintenance.

## Cons

* **Compatibility and *etiquette[[3]](#footnote-3)***: When SDO and Communities develope vocabularies they sometimes couple their conceptual models to specific design and implementation paradigms. Thus, many vocabularies that are of the interest for Public Administrations are only available as RDF models (some of them are not even represented graphically, e.g. FOAF, DCT, other). Some SDO Technical Committees and vocabulary publishers (e.g. OASIS UBL[[4]](#footnote-4), the Publications Office of the European Union footnote:[link:https://docs.oasis-open.org/ubl/UBL-2.3.html[UBL] produces W3C XSD and JSON Schemas. OP's https://op.europa.eu/en/web/eu-vocabularies[EU Vocabularies] publishes all the authority tables in different formats, XML, HTML, SKOS, Genericode.]) are moving towards the production of 'syntax-binding-agnostic' models that would decouple the models and ontologies from 1) the syntax used to express the core model, 2) the expression of additional richer semantics (to add on-top-of the core model), and 3) core reusable rules and restrictions applicable to the core model and the added-on semantics. SEMIC is promoting this methodology for the development and maintenance of vocabularies and ontologies. However, for those vocabularies and resources (e.g. validation methods and artefacts) that are only available under one design and implementation approach and technology, 'mappings' and 'adjustments' between the original specification and the desired design/implementation need to be done. One example of this has been the need to 1) map the DCAT RDF T-Box (and the vocabularies reused by DCAT) to XML Schemas, and 2) to adjust certain datatypes used in RDF to other preferred in XML (e.g. rdf:resource to either xsd:anyURI or xsd:anyType). These actions should to be reported back to the SDO maintaining the specification, as well as the need of decoupling the model from the technological aspects (and producing alternative syntax-bindings). In the meantime, the situation described imposes:
* *The need of constantly watch the evolution of the SDO's works;*
* *The need of maintaining the inner developments that depend on the SDO's works evolution;*
* *The need of re-adapting tools and solutions;*
* *The need of well communicating and disseminating the evolution of the inner works originated by this evolution imposed from the outside;*
* **Adaptibility**: One of the problems of reusing externally governed resources is that requirements that should fall within the compass of one resource, e.g. one vocabulary, are not covered sufficiently or at all. When this happens, the alternatives are not always viable and need to be dealt with internally. At least three situations can be identified related to this problem: pace, customisation and extensibility. The identification of these situations is a relevant task when building the catalogue of reusable ontological and linguistic resources:
* ***Pace***: new legal and business requirements need to be introduced timely in the vocabularies. If the SDO responsible for the maintenance of the vocabulary is not reactive[[5]](#footnote-5) and adapts the pace of its developments to the [re]user needs, the user will need to either look for 1) adapting the vocabulary to its needs, 2) to come up with its own *\_ad hoc\_* solution, or 3) look for combinations of alternative vocabularies that partially meet the user requirements. The optimal option would be the first one, that the vocabulary is adaptable. One indicator that the vocabulary is adaptable is that it can be customised and extended.
* ***Customisation*** *and* ***extensibility***: vocabularies are not truly reusable if they set very strict conditions or constraints[[6]](#footnote-6). As a matter of fact, constraints and particular business entities cannot be fully identified until the vocabulary is 'applied' to a context and, more specifically, to a business domain. Hence, a relevant factor (i.e., a *\_sematic interoperability enabler\_*, cfr. EIRA Key Interoperability Enablers) when identifying and selecting reusable candidate vocabularies is whether they are flexible enough so they can be, later on, be 'profiled'. The ISA2 Core Standard and Specification Vocabulary (https://joinup.ec.europa.eu/solution/core-standards-and-specifications-vocabulary-cssv[CSSV]) and the W3C https://www.w3.org/TR/dx-prof/[The Profiles Vocabulary] use the following definition of 'Application Profile': A [data/application] specification that constrains, extends, combines, or provides guidance or explanation about the usage of other [data/application] specifications".
* **Accessibility**: a relevant criterion for the assessment and selection of reusable resources is how unrestricted is the access to the resources, related documentation, reference implementations and to the participation in their evolution. These are also key factors that moderate the sharing and reuse of the semantic interoperability solutions, and therefore are Key Interoperability Enablers that contribute to the saliency of the interoperability in the European Digital Public Service Value Chain (EDPSVC). They should be taken into account when drafting the principles and recommendations for the cataloguing of reusable semantic solutions.

# COMMENT 2: Target audience

The target audience would vary depending on whether we are referring to the Handbook or to the Style Guides.

In our opinion, Cecile's suggestion of documenting the process of developing a Core Vocabulary is convenient and **should go in the handbook**, since it is of the interest of 'everyone'. The handbook should be the first reference document that ends up leading to the Style Guides.

Implementers, therefore, should also be recommended to have a look into the handbook to grasp the context and the 'story' of how a Core Vocabulary is expected to be developed, maintained, evolved, etc.

Some rationale to support the target audience of both documents, the handbook and the style guides, follow:

* Handbook:
  + Policy-makers should be interested in knowing the EU legal and organisational context and principles that direct the enablement semantic interoperability both at the EU and National levels. Relevant mentions to the EIF approach to interoperability and public service governance (as well as references to the EIRA Ontology and views) should be made in the Handbook. This would help the policy-makers understand the relations between the implementation of public policy and semantic interoperability. (Handbook. Guidelines refer to handbook);
  + CIOs and Project owners: should be aware of the existence of the style guide and handbook, namely because these documents contain references to standards and specifications, methodologies, techniques and resources that are candidates for reusing; which is also aligned with recommendations of the EIF on open standards and sharing and reuse, among other equally important;
  + Project leaders and analysts interested in getting context on the drivers of their projects when these refer or require the reuse of eGovernment Core Vocabularies and Application Profiles;
* Style guide:
  + Project leaders and analysts and developers that are required to understand, apply, and monitor the implementation and use of the technical normative rules and non-normative recommendations (e.g. architectural, naming, design rules and recommendations).
  + Developers with the need of finding pointers to additional information, open reference implementations for testing and reusing compatibility-for-interoperability solutions, bibliography, training-courses, links to existing projects and fora that use or have developed eGovernment Core Vocabularies and Applications profiles, etc.~

# COMMENT 3: target audience, what's an eGCV (definition and justification), just Core Vocabularies? and what about APs?

* Target audience: see comments in sections above.
* About the definition of what is an eGovernment Vocabulary and the justification of their existence:
  + Both should go in the handbook;
  + The Style Guides should summarise them and refer to the handbook for more details.

This would be consistent with the audience, goals and content of both documents.

* About the scope of the Handbook and Guides: only eGCV? Also commented above, but to sum up, our opinion is that each document should encompass:
  + ***Handbook***: principles, recommendations and how-to(s) for the starting and inception, development and governance of eGovernment Core Vocabularies and Application Profiles;
  + ***Style Guide***: architectural and technical guidelines and examples for the implementation, testing, validation and use of eGovernment Vocabularies and Application Profiles.

# COMMENT 4: Reuse needs further clarification

## What is normative and what non-normative

Yes, an effort should be done to clarify what is normative and what is just recommended, to whom it affects and how; and both for reused and for newly designed solutions. However, this is question is related to another dimension which is:

'*Who should make the decision of when to develop a new eGovernment Core Vocabulary?’*

Similarly for Application Profiles.

The answer to this question may be ***politically problematic***. A *diplomatic* expression of these responsibilities could be necessary. Depending of the ‘who’ the normativity could have to be relaxed. We propose that instead or labelling them ‘normative’ and ‘non-normative’ we come up with a more abstract term: what about distinguishing (and defining) ‘principle’, ‘rule’ and ‘recommendation’? ‘Rule’ could be defined as implying that breaking the guideline disables semantic interoperability and may break superior legal and organizational requirements and agreements. ‘Recommendation’ could be defined as implying that not respecting the guideline would impose additional effort and investment in terms of development, documentation, testing, validation, deployment, dissemination, training, etc. The degree of liability would be greater in the case of breaching the rule than if not following the recommendation.

Such a terminology (principle, rule and recommendation) would be easier to understand, and would make unnecessary the use of the RFC-2119 clauses SHOULD, SHALL, MAY, etc., which decrease the readability and user-friendliness of the documents.

The right place to introduce recommendations on this aspect should be the handbook, not the style guides.

* ***Handbook***: general principles and recommendations should not be marked as normative and non-normative in the Hand-book. However, an indicator of the relevance and the impact of not following the principle or recommendation should be suggested (e.g. how it may threaten or hinder the legal, organizational and/or semantic interoperability and consequently the interoperability saliency in the EDPSVC);
* ***Style guides***:
  + Particularly detailed technical rules and recommendations should be marked as normative and non-normative in the style guides, like ‘*import vocabularies, do not re-define entities defined in other vocabularies 🡪 (RULE)*’, ‘*do not change the datatype of the original entity 🡪 (RULE)*’, ‘*do not spare in-line and block comments 🡪 (RECOMMENDATION)’*, ‘*Polymorphism and inheritance is to be used vs. delegated inheritance (i.e. aggregation) 🡪 (RECOMMENDATION)*’.
  + Architectural topics also should be specified as normative or non-normative. As indicated in the General Comments, for example, is the general architecture normative? If yes:
    - Is the separation into three several files normative, i.e. ‘core’, ‘semantics’ and ‘rules and constraints’? or could ‘semantics’ go into the ‘core’ as some believe?
    - If the bullet above is accepted as a recommendation, how does the non-normativity is justified? Perhaps, this has been insufficiently justified in the style guide. For this we propose that:
      * The handbook introduces the general principles and justification for the architectural approach currently proposed in the style guides: separation into different files, yes, but why (reference the style guide for the ‘how’): e.g. core 🡪 domain-agnosticity of the concepts; semantics 🡪 different possible layers of ‘from-lesser-to-greater semantic’ properties and sub-properties that (some of them) may be reused to some extent in multiple domains (ePO has some interesting examples, like ‘document’ is used to ‘announce’, ‘notify’, ‘replace’, ‘supersede’, etc.); constraints 🡪 different possible layers of ‘from generic-to-more-particular’ constraints that (some of them) may be reused to some extent in multiple domains (ESPD has nice examples of this, like generic constraints for the validation of codes and identifiers, less generic for the validation of concept co-occurrence, other).
      * We do not share the opinion expressed in the General Comments that this architectural construct affects the semantics of the property and sub-property definitions. It is important to keep in mind that this is a design and implementation approach (recommendation), and therefore does not affect the conceptual model. The definitions of any type of entity involved in a vocabulary (classes, enumerations, associations, constraints, stereotypes, other) must, as a rule, go in the conceptual model. At design and implementation time (in multiple logic models and syntax-bindings) the separation in – several different but interconnected modules – contributes to clarify how a predicate builds the semantics of the possible associations of a vocabulary, thus contributing also to its implementation of ontology axioms and triple instances and, ultimately, to the automated creation of knowledge. This principle is also a good recommendation (not rule) for Application Profiles reusing the semantics applied to the core vocabulary and to ‘perfecting’ of business-domain semantics.

## Folder-organisation, Cherry-picking, and AP siblings

All the considerations poured into this General Comment are acceptable (and accepted), but need to be put in context, one by one. All of them are related to the topics covered in the previous sections. Nonetheless, let’s add some more reactions to these comments:

* Organisation of vocabularies requested/recommended for reuse in folders:
  + Any solution (vocabulary or other resource) being developed and maintained by an external authority must not be replicated in the SEMIC registries/repositories. The exception to this rule are those resources that are produced and maintained by SEMIC to map and adjust the specifications of third authorities that do not provided them. Example: XSD and JSON versions of RDF vocabularies.
  + Internally developed reusable vocabularies and associated resources should be made available from one single location. They must not be replicated per vocabulary, for example (as it is the case right now in the SEMIC GitHub).
* Cherry-picking of entities: already commented above. This is avoided if the rule is to import the vocabularies and reuse them as they are provided by the authority defining and maintaining them.
* Meaning, scope and uses of eGCVs and APs: also covered in the sections above. We accept the comment that the definition, disambiguation and illustration of what is an eGovernment Core Vocabulary and what is an AP must be done in the handbook. Project officers and leaders and implementers should be redirected to the handbook from inside the Style Guides. There is no point in duplicating the documentation. A brief summary of the difference between one and the other could be included, though, in the Style Guide.

# COMMENT 5: Validation process with regards to definitional and assertional knowledge

## Atomicity of the reusability

The style guide should indeed clarify what is the most atomic entity that is reusable from an external vocabulary or ontology. In our opinion this includes any type of entity that is susceptible of being defined unambiguously and has actually been defined in such a way in the reused vocabulary or ontology; namely classes, properties, constraints; but also related artefacts such as reference data from which the vocabulary could be dependent (by the way, this should never be the case if the reused vocabulary is generic and domain-agnostic).

## Assertional knowledge restrictions

The guidelines do not restrict knowledge about the ‘core’ but only when Application Profiles are being developed. May be this needs further elaboration, both in the handbook and in the style guide.

The idea of ‘conformance’ can be applied to the Core Vocabularies, but as far as Application Profiles are concerned, a very well disambiguated definition of ‘conformance’ and ‘compliance’ should be produced and included in the hand-book (and referred to in the Style Guide). This is a broad and classic discussion that has been documented by some SDOs (UBL amongst them).

The discussion should cover at least two aspects, at least:

1. How customisations, extensions and documentation are conformant to the Core Vocabularies they profile; and
2. How software solutions can demonstrate compliance with the Application Profiles.

## Validation severity levels

Also related to the inception and development of Application Profiles and particular usage in business domains, not to the eCGV. As stated in the general comment, ‘that depends on the usage context’. In our opinion, this would affect the style guide, not the handbook: the guide should recommend the assignment of the severity in function of the context, e.g. the requirements of the transactional business-processes in terms of information requirements and possible restrictions and constraints applied to those processes and requirements (for example, maximum severity if a completitudeness rule concerning a pay-load or of a portion of the pay-load is not met because it would stuck the process or a complete service delivery).

Nonetheless, the handbook could have a general principle like ‘the knowledge of the authorities and experts in the area and domains where an eCGV is being customized, extended and/or specified and documented should be taken into account (DCAT-AP could be taken as an example in the context of sharing datasets between open data portals. Joinup could also be referred to as a sibling Application Profile of ADMS, or is it a meta-AP?).

## What is definitional knowledge

Answer: every entity defined in the conceptual model, which includes: classes, associations, constraints (if any, for example 1..1 or 1..\* cardinalities, requires explanatory/motivational text justifying why this restriction is global to all domains), stereotypes (if any, may also require explanation and justification if they are not design-oriented, which is rarely the case).

Elaboration on this is admittedly needed in the style guide.

## Minimal reusability

Answer: the need for just one property or class already justifies the importation of a whole vocabulary. This may seem tremendous but should not be a technical problem. The enablement of the semantic interoperability is more important that any technological restriction to this rule.

Elaboration on this is admittedly needed in the style guide, e.g. a clear explanation of the minimum reusable item, and the justification for it.

## Compliance

The comment is: is a specification compliant with the Core Vocabulary if it does not satisfy the rdfs:domain/rdfs:range statements?

**Answer**: no, but it could be conformant. The distinction between what is conformance and what is compliance needs to be established in both the handbook and the style guides. See comments above in section ‘Assertional knowledge’

We recommend to add a section on ‘Conformance and compliance’ in the handbook. A reference to this disambiguation should also be included in the Style Guide aiming those project leaders, analysts and developers that may be in charge of producing compliant solutions.

Conformance and compliance are critical to ensure software compatibility and semantic interoperability. See sub-sections proposed for the content of the handbook.

# COMMENT 6: Versioning

## Versioning

The comments are fair. Indeed, the rules stated in *Common approach for the governance and management of persistent URIs by EU institutions* and *Study on persistent URIs, with identification of best practices and recommendations on the topic for the MSs and the EC*. Summarised here:

<https://www.europeandataportal.eu/sites/default/files/d2.1.2_training_module_2.3_persistent_uri_design_and_management_en_edp.pdf>

# COMMENT 7: URI remint

The document created by Makx Deckkers was used as a valuable input for the proposal issued in the style guide. This proposal was however reconsidered and dismissed, though not totally, because (we thought) that the methodological approach in the style guide would not be totally covered by Makx’s proposal (it’s not only about the re-minting, it’s also about the multi-syntaxed production of the conceptual model, and other requirements).

Another aspect included in this document is about who should be the PURI issuer. In our opinion that should be DIGIT, not the OP.

Both aspects would need to be further discussed. We think that these two issues should be be included in the style guide, not in the handbook, but this also can be discussed, since the second one could affect the interest of the policy maker and of the business owners (at least in the case of EU projects).

These two aspects should be generalised though beyond the needs of COM, e.g. the handbook could treat them from a more generic perspective, in the context of other government’s public administrations.

# COMMENT 8: SHACL distribution

The nucleus of the comment is summarized in this sentence:

‘*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?*’

Our answer is, ‘both aspects should be covered’. The architectural approach of layering the semantics and the rule and constraint validation should help in separating this twofold objective. An elaboration of this idea should be included in the style guide. In the handbook, a generic note of the fact that the architectural proposal presented in the style guide could be included to explain how this helps to rationalize and sustain the governance, the evolution and the investment.

Indeed the specification of what is permissible and what not in an Application Profile does not need the development of a SHACL ontology (remember that Shapes are not applied to the core vocabulary, but for the profiling of the core…there seems to be a confusion about this in the comments which indicates that further clarification should be added to both the handbook and the style guide). Narrative documentation complemented with graphical items could be sufficient to establish the AP policy (principles, rules and recommendations).

Nonetheless, the use of shapes to specify the AP can be used for testing the conformance and compliance. Should we discuss this aspect and include it in the handbook chapter on Conformance and Compliance? If yes, then indications (recommendations and references to bibliography and reference implementations) on how this could be achieved should also be included in the style guide.

# COMMENT 9: XML distribution

## Differences between the XML and the RDF architectures

* The XML syntax should not be produced, in our opinion, from the RDF-based syntax, but from the UML model. This should be clearly stated in the handbook and the Style Guide (see ToC entry ‘One single point of maintenance’ proposed for the handbook).
* In the architecture proposed for the XSDs the separation between core and semantics does not make sense and should not be recommended.
* We do not see/understand any problem related to domain and ranges in the XSD section. But we could discuss the problem, if there is any, at any time with the reviewers.
* Logic inference cannot be applied to the XSDs nor to the XML instances. Hence the ‘reasoning’ was taken out from the XML architectural proposal (please check).
* The Application Profiling targets only validation, not definition. Hence the proposal of Schematron. This could also be specified in the Style Guide.

In general, we think that in the handbook three main syntaxes should be mentioned in the chapter about modelling and serialization and explain the pros and cons of each one, or at least when and how/what for they are typically used, and point to where in the guide-lines the implementation in one or another syntax is explained and illustrated.

## Difference between XML eGCV and XML AP

The comment goes like this:

*‘it is unclear if the XSD for an application profile based on a core vocabulary is just a further restriction’*

**Our answer**: fair-enough, the difference should be made clearer. For this we have already proposed that eGCV and AP need to be defined and disambiguated, as proposed by the reviewers. We propose to do that in the handbook; the guideline can sum up the difference and refer to the handbook for complementary information.

Based on the definition of the CSSV and W3C profiles ontology ( the elaboration of what are the differences in terms of implementation should be easy to understand by any type of reader targeted. However, clear rules and recommendations, as well as examples and references to experience, should be added to the style guide about 1) customization and 2) extension.

## Bi-directional mappings

The comment goes like this: Currently the mapping is unidirectional from Core Voc/Application Profile to XML. Is this XML reversible interpretable?

Our answer: if the serialization is performed from the conceptual model the mapping is unnecessary since the ‘pivot’ artefact to produce one or another syntax (any syntax, RDF-based, XML**s** and JSON**s**) is (should be) the UML. Unless the reviewer presumes that the conceptual model is not available and the situation is one of ‘reverse-engineering’. If that is the case the mapping could be done from XML to RDF but the logic inference would not be possible since the axioms from the semantic and reasoning layers cannot be implemented in XML nor JSON.

## Naming conventions: reuse of the name in the URI

The comment can be summarized from this sentence:

*‘The match between the URI definition and the term label will decide if reuse of the external term is directly possible’*

**Our answer**: if the name of the original class or property is not kept, are we not confusing the implementor and breaking the norm of ‘reusing’ (i.e. respecting) also the naming provided by the original specification. In the end the term identifies the semantics of the entity, too. What is relevant for us is that the term is associated to the equivalent primitive type in both syntaxes (XML and RDF) when the type comes from and RDF model. But we can further discuss on why changing the term would be necessary, and how to deal with that situation.

# COMMENT 10: Synergies

The comment goes like this:

‘*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.*’

Our answer: we’re not sure to understand totally this comment. When we have said that the UML conceptual model should be the one-single-point of maintenance and that all syntaxes should be produced based on what the UML says. But we have also said that the UML is to be produced according to normative and non-normative rules and recommendations that are expressed in the specifications contained in the handbook and the style guide.

Therefore, the starting point for the production of an eGCV or an AP should be (in this order) the handbook, the style guide, the UML, and the rest of artefacts accompanying the UML (e.g. configuration files).

May be, the reviewer that issued this comment would be so kind so as to clarify the comment?

# COMMENT 11: Controlled vocabularies

The comments are:

*‘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)’*

**Our answer**:

* Agree, two notes or sub-sections should be included in the handbook, both under the eGCV and about APs explaining why an eGCV does not come accompanied (or specify) reference data and why they may be necessary for the APs. In the style guide this should also be commented and cross-referenced with the hand book,
* A more elaborated explanation should also be added when it comes to what to use for reference data in relation to each syntax. However, we do not totally agree that this should be a separated ‘monographic’ section named ‘Controlled Vocabularies’, at least not in the style guide (we agree that this make more sense for the handbook). The reason is that if the style guides were separated (or printed) in as many documents or documentary pieces per syntax, the rules and recommendations on the controlled vocabularies for that particular syntax could be lost.
* The thing about the ‘string’ mentioned by Enric is that in XML the ultimate datatype behind a code is a ‘xs:normalizedString’. This is the datatype of the ‘code identifier’. URIs can be normalized (<https://en.wikipedia.org/wiki/URI_normalization>) and therefore the rule that a code identifier must be a normalized string should not be a problem.
* Generic rules such as the fact that the serialization of an eGCV or an AP XSD, as well as their instance must be parseable by any XML editor or parser can be added. But it seems so obvious that we did not even consider it. But can be added, of course. It is similar also to saying: the XML produced has to be conformant to the W3C XML specification. Of course.

# COMMENT 12: Impact of the style guide in already used vocabularies

The comment proposes to include in the style guide:

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.*

We can see this in the handbook, more than in the style guide. All these points but the first and last one have already been responded in the sections above, in one way or another.

* Regarding the first point (using an existing voc-how to read the specs), we think the reviewer refers to elucidate whether the vocabulary is actually reusable or not. For this we have proposed to add a section in the handbook setting the criteria for the selection of reusable vocabularies for the catalogue of reusable vocabularies and related semantic interoperability solutions. As a general principle, only generic domain-agnostic vocabularies could be reused for the creation of a new eGCV. In the case of Application Profiles that are close to a business-domain, more domain-focused entities taken from other vocabularies and ontologies could be reused…but this phenomenon should considered and assessed case per case. Instead of reusing domain-coupled entities, it would make more sense to produce new entities that, to the maximum extent, are domain-agnostic and then map vocabularies (see next bullet).
* Defining equivalences to a core vocabulary: Agree, this should be covered both in the handbook and in the style guide. The handbook should describe when this is necessary and, for the how, refer to style guide. The style guide should provide methods, bibliography and tools, or at least to point to where these can be consulted. More importantly, the style guides could point, and briefly describe, to real projects that could be mapped (or that will have the need of be mapped) to the SEMIC eCGV (or other vocabularies developed by other stakeholders). These use cases could be used to illustrate the semantic problem of mapping elements (from the syntax perspective) and date (from the data perspective, e.g. mapping/converting different data types), whilst ensuring the semantics of the original information to the maximum extent. We fear, however that this good intention is ***too ambitious***to be included in the style guide. May be it could be considered as an evolution of the set of documents related to the handbook and style guide.

1. Reusing is not activated if the Sharing-by-design principle is not applied! [↑](#footnote-ref-1)
2. See http://www.heppnetz.de/files/dke2008.pdf and https://papers.ssrn.com/sol3/papers.cfm?abstract\_id=3315457, as well as ISA2 Actions such as Sharing and Reuse, CAMSS, SEMIC governance, etc. [↑](#footnote-ref-2)
3. The term 'ettiquette' is used in some scientific literature about interoperability. It has been applied related to the ICT communications fields to refer to 'protocols of protocols', to 'content-negotiation' and to 'syntax-bridging'. Cfr. XXXXX-REFERENCE]. [↑](#footnote-ref-3)
4. OASIS UBL-2.3: https://docs.oasis-open.org/ubl/UBL-2.3.html [↑](#footnote-ref-4)
5. There may be many reasons for the unresponsiveness of an SDO (or of a particular SDO TC), e.g. the SDO is not affected by the legislation, the SDO's roadmap is misaligned with the incoming requirement, the SDO policy does not allow for the inclusion of requirements that are not sufficiently global (i.e. international or of insufficient interest to a critical mass), etc. One example of highly responsive specification developers are OASIS business-oriented TC, e.g. UBL, LegalDocML, other.]. [↑](#footnote-ref-5)
6. The normalisation and standardisation of restrictive vocabularies become severely hampered. [↑](#footnote-ref-6)