# SemWeb Best Practices for Dummies with PerfectO

| Document Title | Title V2: SemWeb Best Practices for Dummies with PerfectO  
Title V1: SemWeb Best Practices in IoT for Dummies with PerfectO |
|----------------|---------------------------------------------------------------------------------------------------|
| Project | PerfectO  
| Creator | Amelie Gyrard (Kno.e.sis - Ohio Center of Excellence in Knowledge-enabled Computing Wright State University, Ohio, USA)  
Previously (Ecole des Mines de Saint-Etienne, France) |
| Contributors / Feedback to improve the documentation | Ghislain Atemezing (Mondeca, & Linked Open Vocabularies community, France)  
Martin Serrano (Insight Center for Data Analytics, NUIG, Galway, Ireland)  
Utkarshani Jaimini (Kno.e.sis, Wright State University, USA) |
| Last Updated | April 2019  
- Test ontology matching with the LogMap web service. If HTTPS URL, it seems it does not work.  
September 2018:  
- Step by step tutorial for improving Knoesis ontologies, slides done in parallel  
  - Semantic Web Best Practices  
- Updates rule 10 to 16  
  - Prefix.cc, yasgui ontology namespace auto-completion |
<table>
<thead>
<tr>
<th>Month</th>
<th>Activities</th>
</tr>
</thead>
<tbody>
<tr>
<td>August 2018</td>
<td>• Add more rules – checklist summary</td>
</tr>
<tr>
<td>April 2018</td>
<td>• Table of content reorganization</td>
</tr>
<tr>
<td></td>
<td>• Semantic Web Tutorials – Useful links</td>
</tr>
<tr>
<td></td>
<td>• Tutorial for PURL</td>
</tr>
<tr>
<td></td>
<td>• Various improvements – ontology best practices checklist summary</td>
</tr>
<tr>
<td>March 2018</td>
<td>• Screenshot example to avoid bad practices</td>
</tr>
<tr>
<td></td>
<td>• Improve check list summary</td>
</tr>
<tr>
<td></td>
<td>• Various updates</td>
</tr>
<tr>
<td>February 2018</td>
<td>• Adding Ontology metadata code within your ontology</td>
</tr>
<tr>
<td></td>
<td>• Add useful reference to explain the difference between taxonomy, vocabulary, etc.</td>
</tr>
<tr>
<td></td>
<td>• Improving ontology quality at Knoesis, share all best practices, tips and basics to be familiar with best practices.</td>
</tr>
<tr>
<td>September 2017</td>
<td>• Issue with ontologies when suggest them to LOV</td>
</tr>
<tr>
<td>August 2017</td>
<td>• Frequently seen bad practices section</td>
</tr>
<tr>
<td>May 2017:</td>
<td>• All errors encountered when evaluating various ontologies with the different tools</td>
</tr>
</tbody>
</table>

---

**Created**  April 2017  
**Status**  Ongoing Work  

**Goal**
- Sharing a set of semantic best practices
- Help you improve your ontologies
- Documentation to help fix errors encountered to design perfect ontologies with existing Semantic Web (SemWeb) validation or documentation tools!
  - TripleChecker
  - OOPS
  - WebVOWL
  - Parrot
  - Vapour
  - LODE
- Encouraging semantic interoperability.

**Guidance Book**
Dummies term is used for fun to follow the books!  
Do not take it personally! 😊
## Acknowledgments

- Knoesis colleagues with fruitful discussions to improve this guide (Farahnaz Golroo, Shruti Kar, etc.).
- The Linked Open Vocabularies (LOV) Team for sharing their knowledge regarding validation tools and best practices.
  - Pierre-Yves Vandenbussche
  - Maria Poveda (she also created the OOPS validation tool)
  - Bernart Vatant
- Christopher Gutteridge (for his help with TripleChecker)
- Steffen Lohmann (for his help with WebVOWL)
- Carlos Tejo and Sergio Fernández (for their help with Parrot)
- Laurens Rietveld (YASGUI SPARQL editor)
Table of Content

I. Please cite our research work if you are using the PerfectO web site, this guide, etc. .......................... 7

II. Semantic Web and Ontology Basics (UPDATED April 2018) .......................................................... 7
   1. Semantic Web Basics (CREATED April 2018) ................................................................................. 7
   2. Designing your first ontology? Ontology Tutorials (UPDATED April 2018) ..................................... 8
   3. Definitions (NEW February 2018) ................................................................................................. 9

III. Summary of some awesome tools to use ......................................................................................... 9

IV. Ontology Best Practices - Checklist (NEW February 2018) ......................................................... 11
   1. Ontology Best Practices - Checklist Summary (UPDATED August 2018) .................................... 11
   2. Rule 1: Finding a good ontology name (UPDATED March 2018) ............................................... 12
   3. Rule 2: Finding a good ontology namespace (UPDATED April 2018) ........................................... 13
      a) Permanent URL with PURL ........................................................................................................... 13
      b) Permanent URL with W3id ........................................................................................................ 14
   4. Rule 3: Sharing your ontology online (NEW February 2018) ...................................................... 14
   7. Rule 6: All classes should start with an Uppercase. The properties should start with a lowercase (TO DO NEW February 2018) ................................................................. 17
   8. Rule 7: Submitting your ontology to ontology catalogs (NEW February 2018) ...................... 17
   9. Rule 8: Reusing and linking your ontology to other ontologies (TO DO - NEW February 2018) 18
   10. Rule 9: Reuseable URI: copy paste the namespace URL of your ontology in a web browser to get the code ........................................................................................................... 19
   11. Rule 10 (TO DO) ......................................................................................................................... 21
   12. Rule 11 (TO DO) ......................................................................................................................... 21
   13. Rule 12 (TO DO) ......................................................................................................................... 21
   14. Rule 13 (TO DO) ......................................................................................................................... 21
   15. Rule 14 (TO DO) ......................................................................................................................... 21
   16. Rule 15 (TO DO) ......................................................................................................................... 21
   17. Rule 16: Registering your ontology on prefix catalogs .................................................................. 21

V. How to check the design of your ontology (TO DO - NEW February 2018) ............................. 24

VI. Set of Tools to improve ontologies ............................................................................................... 24
   1. Ontology Documentation with Parrot ........................................................................................... 24
      a) Parrot Error: Unable to read input document: invalid mimeType "application/octet-stream" (returned by URI) for parrot .................................................................................. 24
      b) Parrot Error: The ontology can be loaded but nothing is displayed ...................................... 24
      c) Parrot Error: I/O Error: Server returned HTTP response code: 403 for URL ......................... 25
d) Parrot Error: Unable to read input document: application/rdf+xml parse error: Content is
   not allowed in prolog. ................................................................................................................. 26
2. Syntax Validator with TripleChecker .......................................................................................... 27
   a) TripleChecker Error: Possible match to "date" ............................................................................ 27
   b) TripleChecker Error: VERY close match to "license" ................................................................. 27
   c) TripleChecker Error - loading - No parser available ............................................................... 28
3. Ontology Visualization with WebVOWL ...................................................................................... 28
   a) WebVOWL Error: There is nothing to visualize ........................................................................ 28
4. Improving Ontology Design with OOPS ..................................................................................... 28
   a) OOPS Error: Pitfall 36 - URI contains file extension............................................................... 29
   b) OOPS Error: Pitfall 37 - Ontology not available on the Web .................................................. 29
1. Ontology Matching with LogMap web service (NEW: April 2019) .............................................. 31
2. Ontology Documentation with LODE .......................................................................................... 31
   a) LODE Error - Reason: An empty sequence is not allowed as the value of variable $rdf........... 31
   b) LODE Error - Reason: A sequence of more than one item is not allowed as the @select
      attribute of xsl:sort...................................................................................................................... 31
   c) LODE Error - Reason: org.xml.sax.SAXParseException; lineNumber: 1; columnNumber: 1;
      Content is not allowed in prolog. ............................................................................................. 32
   d) LODE Error - The source can't be downloaded in any permitted format. # Connection reset
      # Connection reset # Connection reset # Connection reset # Connection reset # Connection
      reset........................................................................................................................................... 32
   e) LODE Error - The source can't be downloaded in any permitted format. # Received fatal
      alert: protocol_version............................................................................................................ 32
VII. Suggesting Ontologies on LOV .................................................................................................. 33
VIII. Frequently Seen Bad Practices .................................................................................................. 33
IX. Useful Links .................................................................................................................................. 33
X. References ..................................................................................................................................... 33
Table of figures

FIGURE 1. GETTING STARTED WITH APACHE JENA .............................................................. 8
FIGURE 2. WIMMICS TEAM, SEMANTIC WEB COURSE EXAMPLE .................................... 8
FIGURE 3. BAD EXAMPLES: YOU NEED TO FIND AN EXPLICIT NAME TO DESCRIBE YOUR ONTOLOGY .................................................... 12
FIGURE 4. RENAME YOU ONTOLOGY UNDER PROTEGE (AND THINK ABOUT THE ONTOLOGY URL/NAMESPACE) ..................................................... 13
FIGURE 5. CREATE A NEW URL UNDER PURL (E.G., IOT/VOCAB/NAME-ONTO#) ...................... 13
FIGURE 6. LINK THE PURL URLS TO YOUR ONTOLOGY OR ANY WEB SITE .......................... 14
FIGURE 7. ADDING ONTOLOGY METADATA CODE WITHIN YOUR ONTOLOGY .......................... 15
FIGURE 8. BAD PRACTICE EXAMPLE: THERE IS A NEED TO DESCRIBE CONCEPTS OR PROPERTIES TO KNOW THEIR MEANING AND DEFINITION ................................................................. 17
FIGURE 9. GOOD PRACTICE: ADD RDFS:LABEL AND RDFS:COMMENT TO BETTER DESCRIBE CONCEPTS OR PROPERTIES .......................................................... 17
FIGURE 10. ONTOLOGY CATALOGUE MINDMAP WITH A SET OF TOOLS TO USE ..................... 18
FIGURE 11. BAD EXAMPLE: COPY PASTE THE URL NAMESPACE OF YOUR ONTOLOGY IN A WEB BROWSER (E.G., CHROME), YOU SHOULD NOT GET THE "404 – NOT FOUND" ERROR, BUT GET THE ONTOLOGY CODE OR THE DOCUMENTATION ............................................................................. 19
FIGURE 12. GOOD EXAMPLE: COPY AND PASTE AN ONTOLOGY NAMESPACE URL IN A WEB BROWSER TO GET THE CODE ................................................................................................................. 20
FIGURE 13. UNABLE TO READ INPUT DOCUMENT: INVALID MIME TYPE "APPLICATION/OCTET-STREAM" (RETURNED BY URI) FOR PARROT .......................................................... 24
FIGURE 14. NO ERROR ENCOUNTERED WHEN LOADED THE ONTOLOGY BUT NOTHING IS DISPLAYED ............................................................ 25
FIGURE 15. I/O ERROR: SERVER RETURNED HTTP RESPONSE CODE: 403 FOR URL .................. 25
FIGURE 16. UNABLE TO READ INPUT DOCUMENT: APPLICATION/RDF+XML PARSE ERROR: CONTENT IS NOT ALLOWED IN PROLOG ................................................................. 26
FIGURE 17. MISUSE OF TERMS FROM DUBLIN CORE NAMESPACE AND DATE FORMAT WITH TRIPLECHECKER .......................... 27
FIGURE 18. ERROR: VERY CLOSE MATCH TO "LICENSE" .......................................................... 27
FIGURE 19. ERROR LOADING- NO PARSER AVAILABLE ......................................................... 28
FIGURE 20. ERROR "THERE IS NOTHING TO VISUALIZE" ...................................................... 28
FIGURE 21. OOPS REPORT WITH CRITICAL PITFALLS ......................................................... 29
FIGURE 22. OOPS PITFALL 36 URI CONTAINS FILE EXTENSION ............................................... 29
FIGURE 23. OOPS ERROR: P37: ONTOLOGY NOT AVAILABLE ON THE WEB ......................... 30
FIGURE 24. REASON: AN EMPTY SEQUENCE IS NOT ALLOWED AS THE VALUE OF VARIABLE $RDF ................................................................. 31
FIGURE 25. LODE ERROR - REASON: A SEQUENCE OF MORE THAN ONE ITEM IS NOT ALLOWED AS THE @SELECT ATTRIBUTE OF XSL:SORT ............................................................................. 31
FIGURE 26. REASON: org.xml.sax.SAXParseException; lineNumber: 1; columnNumber: 1; Content is not allowed in prolog................................................................................................. 32
FIGURE 27. LODE Error - The source can’t be downloaded in any permitted format. # Connection reset # Connection reset # Connection reset # Connection reset # Connection reset # Connection reset ................................................................. 32
FIGURE 28. THE SOURCE CAN’T BE DOWNLOADED IN ANY PERMITTED FORMAT. # RECEIVED FATAL ALERT: PROTOCOL_VERSION ............................................................................. 32
FIGURE 29. LOV SUGGEST ERROR – SUGGESTING AN ONTOLOGY TO LOV .............................. 33
I. Please cite our research work if you are using the PerfectO web site, this guide, etc.

Please do not forget to cite our research work if you are using this documentation and the PerfectO web site. Thank you very much in advance.

Publications:

- Semantic Web Methodologies, Best Practices and Ontology Engineering Applied to Internet of Things.
  - IEEE World Forum on Internet of Things (WF-IoT), Milan, Italy, December 14-16, 2015
  - Amelie Gyrard, Martin Serrano, Ghislain Atemezing.
- A survey and analysis of ontology-based software tools for semantic interoperability in IoT and WoT landscapes.
  - IEEE 4th World Forum on Internet of Things (WF-IoT), 2018
  - Amelie Gyrard, Soumya Kanti Datta, Christian Bonnet
- Semantic Web Guidelines for domain knowledge interoperability to build the Semantic Web of Things.

II. Semantic Web and Ontology Basics (UPDATED April 2018)

This section recommends useful links. They are already numerous tutorials and courses on the web. We just provide few recommendations.

1. Semantic Web Basics (CREATED April 2018)

There are already nice tutorials to learn Semantic Web basics.

The Jena Semantic Web Framework. Jena is a framework for Java developers. They provide excellent tutorial to learn the basics¹ (see Figure 1):

- Jena tutorial - An Introduction to RDF and the Jena RDF API
- Jena tutorial - SPARQL language, a SQL-like language to query semantic web data, called triples.
- Jena tutorial - Manipulating SPARQL using ARQ
- OWL and ontologies to model and structure data
- Inference engine and rules

¹ https://jena.apache.org/getting_started/index.html
Getting started with Apache Jena

We also recommend the Semantic Web class from the Wimmics team, INRIA Sophia Antipolis, France (see Figure 2):

- Corese tutorial and tool

2013-2014

- Corese
- Inference Rules
- TP Inference Rules
- Template Inference Rules

2012-2013

- Corese API
- Corese
- Corese Rules
- RIF
- TP Rules
- SPARQL 1.0
- TD SPARQL 1.0
- SPARQL 1.1
- SPARQL Update
- TD SPARQL 1.1

More and more universities are having MOOC and Semantic Web course.

2. Designing your first ontology? Ontology Tutorials (UPDATED April 2018)

Enclosed a set of useful links:

- Protégé: A software having a Graphical User Interface (GUI) to design and develop ontologies.
- Protégé Tutorial [Horridge et al. 2011] – Design the Pizza ontology. Check if there is a more recent documentation.
• The Neon methodology to discover more about ontology best practices.

Book Recommendation: Handbook of Ontologies [Staab et al. 2010]

3. Definitions (NEW February 2018)
What are the differences between Taxonomy, Ontology, Thesaurus, Vocabulary, Ontology/namespace, Schema, etc.?

You can find those answers here:
• Taxonomies & Controlled Vocabularies SIG explains the difference between Vocabularies, Taxonomies, Thesauri and Ontologies.
• http://www.dataversity.net/taxonomy-vs-ontology-machine-learning-breakthroughs/

III. Summary of some awesome tools to use

<table>
<thead>
<tr>
<th>Name Tool</th>
<th>Goal</th>
<th>URL</th>
<th>Reference</th>
</tr>
</thead>
<tbody>
<tr>
<td>TripleChecker</td>
<td>Syntax validation</td>
<td><a href="http://graphite.ecs.soton.ac.uk/checker/">http://graphite.ecs.soton.ac.uk/checker/</a></td>
<td>No</td>
</tr>
<tr>
<td>OOPS</td>
<td>Design a better ontology</td>
<td><a href="http://oops.linkeddata.es/">http://oops.linkeddata.es/</a></td>
<td>[3]</td>
</tr>
<tr>
<td>LODE</td>
<td>Automatic Ontology Documentation</td>
<td><a href="http://www.essepuntato.it/lode">http://www.essepuntato.it/lode</a></td>
<td>[9]</td>
</tr>
<tr>
<td>OWL Manchester</td>
<td>Syntax validation</td>
<td><a href="http://mowl-power.cs.man.ac.uk:8080/validator/">http://mowl-power.cs.man.ac.uk:8080/validator/</a> or <a href="http://visualdataweb.de/validator/">http://visualdataweb.de/validator/</a></td>
<td>Not found yet</td>
</tr>
<tr>
<td>LOV Suggest</td>
<td>Suggest the ontology to LOV</td>
<td><a href="http://lov.okfn.org/dataset/lov/suggest">http://lov.okfn.org/dataset/lov/suggest</a></td>
<td>[1]</td>
</tr>
<tr>
<td>PURL</td>
<td>Permanent URLs</td>
<td><a href="https://archive.org/services/purl/">https://archive.org/services/purl/</a></td>
<td>Not found yet</td>
</tr>
<tr>
<td>W3id</td>
<td>Permanent URLs</td>
<td><a href="https://w3id.org/">https://w3id.org/</a></td>
<td></td>
</tr>
</tbody>
</table>
IV. **Ontology Best Practices - Checklist (NEW February 2018)**

Some recommendations to guide you to design a better ontology. Check all of those items.

1. **Ontology Best Practices - Checklist Summary (UPDATED August 2018)**

<table>
<thead>
<tr>
<th>Rules</th>
<th>Description</th>
<th>More explanations</th>
<th>Difficulty (*, **, ***</th>
<th>Status (Done, In progress)?</th>
<th>Estimated time</th>
</tr>
</thead>
<tbody>
<tr>
<td>Rule 1</td>
<td>Finding a good ontology name</td>
<td>Find an explicit name for your ontology, we frequently see “unnamed.owl”!</td>
<td>*</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Rule 2</td>
<td>Finding a good ontology namespace</td>
<td>Ideally on PURL or W3id. Otherwise, think about the server hosting the ontology (e.g., <a href="http://knoesis.org/ontology/nameOntology#">http://knoesis.org/ontology/nameOntology#</a>)</td>
<td>**</td>
<td>15 mins</td>
<td></td>
</tr>
<tr>
<td>Rule 3</td>
<td>Sharing your ontology online</td>
<td>Accessible with an URL (<a href="http://knoesis.org/ontology/nameOntology#">http://knoesis.org/ontology/nameOntology#</a>)</td>
<td>**</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>No server? Push the ontology code on Github? No server? we can host the ontology code on the LOV4IoT server (but is the ontology stable version?)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Rule 4</td>
<td>Adding ontology metadata</td>
<td>This is important to later reference the ontology on ontology catalogs, or even to provide automatic ontology visualization, ontology documentation, etc.</td>
<td>*</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Rule 5</td>
<td>Adding rdfs:label, rdfs:comment, dc:description for each concept and property</td>
<td>This is important to later provide automatic ontology visualization, documentation, etc. Some tools prefer dc:description, check which ones (e.g., LODE?).</td>
<td>*</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Rule 6</td>
<td>All classes should start with an uppercase and properties with a lowercase.</td>
<td>To follow usual software and ontology development guidelines.</td>
<td>*</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Rule 7</td>
<td>Submitting your ontology to ontology catalogs</td>
<td>Ontology catalogs: LOV, LOV4IoT, BioPortal. It depends on your applicative domain.</td>
<td>**</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Rule 8</td>
<td>Reusing and linking ontologies</td>
<td>Reuse an existing concepts and properties from an existing ontology/namespace (e.g., ssn:Device) Otherwise add owl:EquivalentClass, owl:sameAs, owl:equivalentProperty, etc.</td>
<td>***</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Rule</td>
<td>Description</td>
<td>Importance</td>
<td>Notes</td>
<td></td>
<td></td>
</tr>
<tr>
<td>-------</td>
<td>------------------------------------------------------------------------------</td>
<td>---------------------------------------------------------------------------</td>
<td>-------</td>
<td></td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>Deferenceable URI: copy paste the namespace URL of your ontology in a web browser to get the code.</td>
<td>Important to automatize the tasks to automatically retrieve the ontology code for automatic analysis of ontologies.</td>
<td>**</td>
<td></td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>Checking syntax validator</td>
<td>TripleChecker tool is an easy web service to use. It can check incorrect use of ontologies. Other tools: OWL Validator, RDF Validator, etc.</td>
<td>*</td>
<td></td>
<td></td>
</tr>
<tr>
<td>11</td>
<td>Adding ontology documentation</td>
<td>Ontology documentation can be done automatically with easy to use tools by using their web services if you have labels and comments. E.g., LODE, Widoco, Parrot.</td>
<td>*</td>
<td></td>
<td></td>
</tr>
<tr>
<td>12</td>
<td>Adding ontology visualization</td>
<td>Usage of the WebVOWL tool to provide the ontology visualization automatically.</td>
<td>*</td>
<td></td>
<td></td>
</tr>
<tr>
<td>13</td>
<td>Improving Ontology Design</td>
<td>Usage of the Oops tool to improve the ontology design.</td>
<td>***</td>
<td></td>
<td></td>
</tr>
<tr>
<td>14</td>
<td>Improving deferring URI and content negotiation</td>
<td>Usage of the Vapour tool</td>
<td>***</td>
<td></td>
<td></td>
</tr>
<tr>
<td>15</td>
<td>Loading the ontology on the most popular ontology editor</td>
<td>Should be able to be loaded under Protégé Ontology editor since it is a popular editor Sometimes ontologies cannot be loaded or are empty.</td>
<td>**</td>
<td></td>
<td></td>
</tr>
<tr>
<td>16</td>
<td>Registering your ontology on prefix catalogs</td>
<td>Registering your ontology (namespace and prefix) on prefix.cc. For instance, it is relevant for YASGUI auto-completion. Permanent URL are important (e.g., possible with PURL)</td>
<td>*</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

2. **Rule 1: Finding a good ontology name (UPDATED March 2018)**

Do not forget to find a good name for your ontology. Frequently we find “unnamed.owl”, “schema.ttl” or “ontology.owl”! Please find an explicit name related to your project, etc.

Further, find the best prefix (.xml, .rdf, .rdfs, .owl)

![Figure 3. Bad examples: You need to find an explicit name to describe your ontology](image-url)
Figure 4. Rename your ontology under Protégé (and think about the ontology URL/namespace)

1. **Rule 2: Finding a good ontology namespace (UPDATED April 2018)**

   When you copy paste the ontology URL or even the ontology namespace in a web browser you should get the ontology documentation or the ontology code and not Error 404, page not found. 😊

   a) **Permanent URL with PURL**

   Get a permanent URL with the PURL tool: [https://archive.org/services/purl/](https://archive.org/services/purl/)

   You can easily create a permanent URL with Purl. It takes less than 15 minutes:

   - Create an account and register under this tool.
   - Create a new domain.
     - We have created [http://purl.org/iot/ontology/](http://purl.org/iot/ontology/) for any IoT ontologies
     - Example: [http://purl.org/iot/vocab/m3-lite#](http://purl.org/iot/vocab/m3-lite#)

---

**Figure 5. Create a new URL under PURL (e.g., iot/vocab/name-onto#)**
Figure 6. Link the PURL URLs to your ontology or any web site

Example:

b) **Permanent URL with W3id**

You can do something similar with the w3id tool: [https://w3id.org/](https://w3id.org/)

However, we have noticed that your URLs will start with HTTPS which might generate issues with other tools hosted on non-secured servers!

For this reason, we encourage PURL also because it is faster to create a PURL URL.

2. **Rule 3: Sharing your ontology online (NEW February 2018)**

Ideally, share your ontology code online, for instance you can either:

- Share the file on a server
- Share the file on Github, BitBucket, etc.

Adding ontology metadata is important. We need to know when the ontology has been created, if the ontology is still maintained, the creators to contact them, the related papers explaining more about the project and the ontology, etc.

**Why is it important?**

Adding ontology metadata is important to be referenced by ontology catalog, to provide automatic ontology visualization, automatic documentation, etc.

Further, machines could automatically process your ontology. For instances, semantic search engines could automatically retrieve your ontology.

We encourage to follow the recommendation from this paper [11]. To reduce the learning curve, enclosed the code below.

```xml
<owl:Ontology rdf:about="http://purl.org/iot/ontology/nameOnt#"
  <owl:versionInfo>1.0/owl:versionInfo</owl:versionInfo>
  <dc:title>Work in progress</dc:title>
  <dcterms:modified rdf:datatype="http://www.w3.org/2001/XMLSchema#date">2018-02-16</dcterms:modified>
  <dcterms:issued rdf:datatype="http://www.w3.org/2001/XMLSchema#date">2018-02-16</dcterms:issued>
  <dcterms:license rdf:resource="http://www.pablexample/Image.png">Copyright - What is the license for your ontology</dcterms:license>
  <dcterms:bibliographicCitation rdf:resource="http://www.pablexample/Image.png">Citation of the paper describing your ontology</dcterms:bibliographicCitation>
</owl:Ontology>
```

**Figure 7. Adding Ontology metadata code within your ontology**

Copy paste the following code to add ontology metadata. This code is for any ontology implemented with OWL/XML:
<owl:Ontology rdf:about="http://purl.org/iot/ontology/fiesta-iot#">
  <owl:versionInfo>1.0</owl:versionInfo>
  <vs:term_status>Work in progress</vs:term_status>
  <dcterms:modified rdf:datatype="http://www.w3.org/2001/XMLSchema#date">2018-02-16</dcterms:modified>
  <dcterms:issued rdf:datatype="http://www.w3.org/2001/XMLSchema#date">2018-02-16</dcterms:issued>
  <dc:title xml:lang="en">Name of your ontology Ontology</dc:title>
  <rdfs:label rdf:datatype="http://www.w3.org/2001/XMLSchema#string">Find a nice name for your ontology</rdfs:label>
  <rdfs:comment xml:lang="en">Description of your ontology is important for automatic, visualization documentation, etc.</rdfs:comment>
  <dc:creator xml:lang="en">XXX, Knoesis, USA</dc:creator>
  <dc:contributor xml:lang="en">XXX, Knoesis, USA</dc:contributor>
  <dc:description xml:lang="en">Describe the ontology</dc:description>
  <vann:preferredNamespaceUri>http://purl.org/iot/ontology/nameOnto#</vann:preferredNamespaceUri>
  <vann:preferredNamespacePrefix>Find a short prefix for your ontology</vann:preferredNamespacePrefix>
  <dc:rights>Copyright - What is the license for your ontology</dc:rights>
  <dcterms:bibliographicCitation xml:lang="en">Citation of the paper describing your ontology</dcterms:bibliographicCitation>
</owl:Ontology>

We recommend to follow the recommendation from this paper:

Figure 8. Bad practice example: There is a need to describe concepts or properties to know their meaning and definition

Recommendation example:

```xml
<Declaration>
    <Class IRI="Availability"/>
</Declaration>
<Declaration>
    <Class IRI="AvailableBikes"/>
</Declaration>
<Declaration>
    <Class IRI="AvailableDesks"/>
</Declaration>
```

Figure 9. Good Practice: Add rdfs:label and rdfs:comment to better describe concepts or properties

<table>
<thead>
<tr>
<th>Suggestion for improvement:</th>
</tr>
</thead>
<tbody>
<tr>
<td>Protégé tool needs to encourage ontology developers to create labels and comments. It is important to have such descriptions to have automatic ontology visualization, documentation, automatic alignment, etc.</td>
</tr>
</tbody>
</table>

5. **Rule 6:** All classes should start with an Uppercase. The properties should start with a lowercase *(TO DO NEW February 2018)*

6. **Rule 7:** Submitting your ontology to ontology catalogs *(NEW February 2018)*

To disseminate your ontology, we recommend to suggest your ontology to ontology catalogues. There are numerous ontology catalogues:

- **Linked Open Vocabularies (LOV)** designed by the Semantic Web community and requires that your ontology follows a set of best practices.
  - Suggest you ontology on LOV here.
- **Linked Open Vocabularies for Internet of Things (LOV4IoT)** references more than 400 ontology-base projects using sensors and classified according to IoT applicative domain (e.g.,
healthcare, smart city, etc.). The ontologies are also classified according to their best practices status (ontology shared online, ontology referenced by the LOV community, etc.)

- Suggest your ontology on LOV4IoT here
- BioPortal, with a focus on health and biomedical ontologies
- OpenSensingCity with a focus on smart city.
- Ready4SmartCity, seems not maintained anymore.

Much more is explained here. With an interactive mindmap (see also figure below for a quick overview).

---

**Figure 10. Ontology catalogue Mindmap with a set of tools to use.**

---

7. **Rule 8: Reusing and linking your ontology to other ontologies (TO DO - NEW February 2018)**

Reuse an existing concepts and properties from an existing ontology/namespace (e.g., ssn:Device)

owl:EquivalentClass, owl:sameAs, owl:equivalentProperty,

TO DO: Explain the differences.
8. **Rule 9: Deferenceable URI: copy paste the namespace URL of your ontology in a web browser to get the code**

It means that when you copy paste the namespace URL of your ontology, you should get the code or the documentation.

Frequently we get the “404 – Not Found” Error:

![404 - Not Found](image)

**Figure 11.** Bad example: Copy paste the URL namespace of your ontology in a web browser (e.g., Chrome), you should not get the “404 – Not Found” Error, but get the ontology code or the documentation.

Good practice example:

http://purl.org/iot/ontology/fiesta-iot# redirects to http://ontology.fiesta-iot.eu/ontologyDocs/fiesta-iot.owl# which provides the ontology code.
Figure 12. Good example: copy and paste an ontology namespace URL in a web browser to get the code
9. Rule 10 (TO DO)
10. Rule 11 (TO DO)
11. Rule 12 (TO DO)
12. Rule 13 (TO DO)
13. Rule 14 (TO DO)
14. Rule 15 (TO DO)
15. Rule 16: Registering your ontology on prefix catalogs

http://purl.org/dao#
This is probably because yasgui cached the prefixes locally in your browser. To test, you can open a private browser window (in chrome, that's 'ctrl+shift+n'), and try yasgui from there.
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX dao: <http://purl.org/dao#>


LIMIT 10
V. How to check the design of your ontology (TO DO - NEW February 2018)

TO DO: How to check the conceptual level?
Otherwise check with the Oops tool.

VI. Set of Tools to improve ontologies

1. Ontology Documentation with Parrot
We have encountered numerous errors when loading IoT ontologies.
   a) Parrot Error: Unable to read input document: invalid mimeType
      "application/octet-stream" (returned by URI) for parrot

Figure 13. Unable to read input document: invalid mimeType "application/octet-stream" (returned by URI) for parrot

<table>
<thead>
<tr>
<th>How to fix this?</th>
</tr>
</thead>
<tbody>
<tr>
<td>The mimetype returned for the Ontology URL is &quot;application/octet-stream&quot; (checked using curl -I), so Parrot is not able to identify (by content negotiation) the parser it use. In the web interface, you can select a concrete type (instead of &quot;allow content negotiation&quot;).</td>
</tr>
<tr>
<td>After selecting &quot;It is an OWL ontology&quot;, the documentation is generated.</td>
</tr>
</tbody>
</table>

b) Parrot Error: The ontology can be loaded but nothing is displayed
Figure 14. No error encountered when loaded the ontology but nothing is displayed

How to fix this?

The content-type returned by the server is "text/html", so it uses a parser for HTML. I guess that the file with the ontology URL is ending with .owl (even this file is served as "text/html").

When the parser is forced using the web interface ("It is an OWL ontology"), the documentation is generated.

c) Parrot Error: I/O Error: Server returned HTTP response code: 403 for URL

Figure 15. I/O Error: Server returned HTTP response code: 403 for URL

How to fix this?

Using curl, this address returns a 200 OK code (with a strange content-type "application/rdf+xml"); see the \ before the '+' symbol).

After some testing, the server is not serving content if the "Accept" header does not include the "\+'.

I was able to generate the documentation in 2 steps. First, I have downloaded the file. Second, I have run Parrot using the "by file upload" tab and selecting directly the "It is an OWL ontology" parser.
d) Parrot Error: Unable to read input document: application/rdf+xml parse error: Content is not allowed in prolog.

Figure 16. Unable to read input document: application/rdf+xml parse error: Content is not allowed in prolog.

How to fix this?

The mimetype returned for the address ontology URL is "application/octet-stream" (checked using curl -l), so Parrot is not able to identify (by content negotiation) the parser to use. Also, the extension of the file is .owl and it looks like a Turtle file (.ttl).

But in that case, Parrot is not returning nothing if you change the parser.
2. Syntax Validator with TripleChecker

The RDF Triple-Checker tool helps find typos and common errors in RDF datasets or OWL ontologies.

a) **TripleChecker Error: Possible match to "date"**

<table>
<thead>
<tr>
<th>Property</th>
<th>Namespace</th>
<th>Type</th>
<th>Looks Legit?</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td><a href="http://purl.org/dc/elements/1.1/">http://purl.org/dc/elements/1.1/</a></td>
<td>issued</td>
<td>ERROR - Possible match to &quot;date&quot;. probable type? [st-]</td>
</tr>
<tr>
<td>1</td>
<td><a href="http://purl.org/dc/elements/1.1/">http://purl.org/dc/elements/1.1/</a></td>
<td>modified</td>
<td>ERROR - Possible match to &quot;date&quot;. probable type? [st-]</td>
</tr>
</tbody>
</table>

*Figure 17. Misuse of terms from Dublin Core namespace and date format with TripleChecker*

![How to fix this?](https://example.com/how-to-fix.png)

You might have mixed between two namespaces:
xmlns:dc= [http://purl.org/dc/elements/1.1/](http://purl.org/dc/elements/1.1/) instead of
"modified" an "issued" terms are from dcterms and not dc namespace.

You might need to modify the code as well:

```xml
&lt;dc:modified&gt;2017-04-20&lt;/dc:modified&gt;
&lt;dc:issued&gt;2016-04-28&lt;/dc:issued&gt;

by:
&lt;dcterms:issued
rdf:datatype="[http://www.w3.org/2001/XMLSchema#date](http://www.w3.org/2001/XMLSchema#date)"&gt;2016-04-28&lt;/dcterms:issued&gt;
&lt;dcterms:modified
rdf:datatype="[http://www.w3.org/2001/XMLSchema#date](http://www.w3.org/2001/XMLSchema#date)"&gt;2017-04-20&lt;/dcterms:modified&gt;
```

Additionnal documentation:
- [http://dublincore.org/documents/2012/06/14/dcmi-terms/?v=elements#terms-issued](http://dublincore.org/documents/2012/06/14/dcmi-terms/?v=elements#terms-issued)
- [http://dublincore.org/documents/2012/06/14/dcmi-terms/?v=elements#terms-modified](http://dublincore.org/documents/2012/06/14/dcmi-terms/?v=elements#terms-modified)

b) **TripleChecker Error: VERY close match to "license"**

<table>
<thead>
<tr>
<th>Count</th>
<th>Type</th>
<th>Namespace</th>
<th>Term</th>
<th>Looks Legit?</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>property</td>
<td><a href="http://creativecommons.org/ns#">http://creativecommons.org/ns#</a></td>
<td>license</td>
<td>ERROR - VERY close match to &quot;license&quot;. probable type? [st-]</td>
</tr>
</tbody>
</table>

*Figure 18. ERROR: VERY close match to “license”*

![How to fix this?](https://example.com/how-to-fix.png)

TO DO
c) TripleChecker Error - loading - No parser available

Figure 19. Error loading - No parser available

<table>
<thead>
<tr>
<th>How to fix this?</th>
<th>TO DO</th>
</tr>
</thead>
</table>

3. **Ontology Visualization with WebVOWL**

We have encountered some errors when loading IoT ontologies.

a) **WebVOWL Error: There is nothing to visualize**

Figure 20. ERROR "There is nothing to visualize"

<table>
<thead>
<tr>
<th>How to fix this?</th>
<th>The ontology should passed the test with the OWL Manchester tool.</th>
</tr>
</thead>
</table>

4. **Improving Ontology Design with OOPS**

Several kind of pitfalls can be encountered:
- Critical
- Important
- Minor

It is obvious that not all the pitfalls are equally important; their impact in the ontology will depend on multiple factors. For this reason, each pitfall has an importance level attached indicating how important it is. We have identified three levels:

- **Critical**: It is crucial to correct the pitfall. Otherwise, it could affect the ontology consistency, reasoning, applicability, etc.
- **Important**: Though not critical for ontology function, it is important to correct this type of pitfall.
- **Minor**: It is not really a problem, but by correcting it, we will make the ontology nicer.

### Figure 21. OOPS Report with Critical pitfalls

| Results for P02: Creating synonyms as classes. | 1 case | Minor |
| Results for P04: Creating unconnected ontology elements. | 16 cases | Minor |
| Results for P07: Merging different concepts in the same class. | 11 cases | Minor |
| Results for P06: Missing annotations. | 10 cases | Minor |
| Results for P11: Missing domain or range in properties. | 27 cases | Important |
| Results for P12: Equivalent properties not explicitly declared. | 4 cases | Important |
| Results for P13: Inverse relationships not explicitly declared. | 54 cases | Minor |
| Results for P20: Misusing ontology annotations. | 1 case | Minor |
| Results for P21: Using a miscellaneous class. | 1 case | Minor |
| Results for P22: Using different naming conventions in the ontology. | ontology | Minor |
| Results for P30: Equivalent classes not explicitly declared. | 8 cases | Important |
| Results for P40: Namespace hijacking. | 2 cases | Critical |
| **SUGGESTION**: symmetric or transitive object properties. | 12 cases |

### Figure 22. OOPS Pitfall 36 URI contains file extension

**How to fix this?**

Please check the documentation on OOPS web site to fix the errors.

#### a) OOPS Error: Pitfall 36 - URI contains file extension

This pitfall occurs if file extensions such as ".owl", ".rdfs", ".ttf", ".n3" and ".jena" are included in an ontology URI. This pitfall is related with the recommendations provided in [9].

*This pitfall applies to the ontology in general instead of specific elements.

### Figure 22. OOPS Pitfall 36 URI contains file extension

**How to fix this?**

The URL tested has something like: [http://www.onto.org/ontology/Base_Ontology/BaseOntology.owl](http://www.onto.org/ontology/Base_Ontology/BaseOntology.owl)

Better to remove the .owl extension.

Be careful, when you copy paste the URL of the ontology on the web browser, we still need to get the ontology file and not the 404 page not found.

#### b) OOPS Error: Pitfall 37 - Ontology not available on the Web

---

How to fix this? Please check the documentation on OOPS web site to fix the errors.

### Figure 22. OOPS Pitfall 36 URI contains file extension

**How to fix this?**

The URL tested has something like: [http://www.onto.org/ontology/Base_Ontology/BaseOntology.owl](http://www.onto.org/ontology/Base_Ontology/BaseOntology.owl)

Better to remove the .owl extension.

Be careful, when you copy paste the URL of the ontology on the web browser, we still need to get the ontology file and not the 404 page not found.

#### b) OOPS Error: Pitfall 37 - Ontology not available on the Web

---
Figure 23. OOPS Error: P37: Ontology not available on the Web

How to fix this?  TO DO
1. **Ontology Matching with LogMap web service (NEW: April 2019)**

LogMap error encountered:

We have detected an error when parsing the input ontologies. Possible causes: the given URI is not accessible, the URI contains a non permitted character, or the given ontology format is not currently accepted.

In this case, 2 URLs tested:
- [http://purl.org/kao#](http://purl.org/kao#) which redirects to:

Probably because, of the HTTPS, because if we load the file itself, the matching will work.

2. **Ontology Documentation with LODE**

We have encountered numerous errors when loading IoT ontologies.

a) **LODE Error - Reason: An empty sequence is not allowed as the value of variable $rdf**

LODE error

Reason: An empty sequence is not allowed as the value of variable $rdf

Figure 24. Reason: An empty sequence is not allowed as the value of variable $rdf

How to fix this? TO DO

b) **LODE Error - Reason: A sequence of more than one item is not allowed as the @select attribute of xsl:sort**

LODE error

Reason: A sequence of more than one item is not allowed as the @select attribute of xsl:sort

Figure 25. LODE Error - Reason: A sequence of more than one item is not allowed as the @select attribute of xsl:sort

How to fix this? TO DO
c) **LODE Error** - Reason: org.xml.sax.SAXParseException; lineNumber: 1; columnNumber: 1; Content is not allowed in prolog.

**LODE error**

**Reason:** org.xml.sax.SAXParseException; lineNumber: 1; columnNumber: 1; Content is not allowed in prolog.

Figure 26. Reason: org.xml.sax.SAXParseException; lineNumber: 1; columnNumber: 1; Content is not allowed in prolog.

<table>
<thead>
<tr>
<th>How to fix this?</th>
<th>TO DO</th>
</tr>
</thead>
</table>

---

d) **LODE Error** - The source can't be downloaded in any permitted format. # Connection reset # Connection reset # Connection reset # Connection reset # Connection reset # Connection reset # Connection reset # Connection reset

**LODE error**

**Reason:** The source can't be downloaded in any permitted format. # Connection reset # Connection reset # Connection reset # Connection reset # Connection reset # Connection reset # Connection reset

Figure 27. LODE Error - The source can't be downloaded in any permitted format. # Connection reset # Connection reset # Connection reset # Connection reset # Connection reset # Connection reset # Connection reset # Connection reset

<table>
<thead>
<tr>
<th>How to fix this?</th>
<th>TO DO</th>
</tr>
</thead>
</table>

---

e) **LODE Error** - The source can't be downloaded in any permitted format. # Received fatal alert: protocol_version

**LODE error**

**Reason:** The source can't be downloaded in any permitted format. # Received fatal alert: protocol_version

Figure 28. LODE error. The source can't be downloaded in any permitted format. # Received fatal alert: protocol_version

<table>
<thead>
<tr>
<th>How to fix this?</th>
<th>TO DO</th>
</tr>
</thead>
</table>
VII. Suggesting Ontologies on LOV

Figure 29. LOV Suggest Error – Suggesting an ontology to LOV

An error occurred:


Bad IRI: <http://example.com> Code: 57/REQUIRED_COMPONENT_MISSING in HOST: A component that is required by the schema is missing.

How to fix this? TO DO

VIII. Frequently Seen Bad Practices

- The default namespace has not been changed (frequently with Protege). A protégé extension would be nice to encourage to have a good namespace.
- Ontologies only available within a PDF file! Arghhhhh!!!!
- See more on this document: http://sensormeasurement.appspot.com/publication/OneM2MBestPractices.pdf

IX. Useful Links

By the creator of TripleChecker: "Linked Data Basics for Techies":

X. References


