OBDA Specification Language (OSL) Specification

Logic and Intelligent Data,
Department of Informatics,
University of Oslo

List of Tables

1 The OBDA Specification Language (OSL)

TODO: aims, proceeding, structure As described in [SGH+15], an OBDA specification consists of several types of maps, all containing data entries and links to other maps. This fits perfectly into the environment of ontologies and OWL, with data properties being the obvious choice to represent contained data entries and object properties being the obvious choice to represent links between maps. Also, a potential user probably to some degree is familiar with this environment, since this is what the bootstrapping process at the end amounts to.

Therefore, an ideal base for the OBDA Specification Language is OWL, being a solid framework for data and constraint representation with a high degree of software support, while imposing only a minimum of introductory preparation to the user.

Another advantage of this approach is that the specification is kept compact and focused on the entities that the language has to represent rather than primarily dealing with technical details. In particular, many of those details can be formulated as OWL restrictions in a header ontology demanded to be imported by documents conforming to the OSL specification. Thus, they are not only specified precisely but they are also stipulated in a machine-readable form for which tools are widely available, enabling the user to check many aspects of an OSL document for conformity with minimal effort.

1.1 Specification

[1] An OSL document is a valid OWL 2 document (as described in [W3C12]) containing individuals and data that represent the OBDA specification, as well as OWL properties that connect them. The individuals and OWL properties are recognized and mapped to their roles by their IRIs.

[2] An OSL document may contain more OWL entities (with IRIs not defined in this specification), which are ignored.

[3] An OSL document has to declare all individuals having different IRIs as different from each other (except those which are ignored, see paragraph 2).
It is recommended to use the `owl:AllDifferent` OWL statement for this purpose.

[4] Unless stated otherwise, IRIs mentioned in the following are IRIs relative to a base IRI chosen by the user being empty (which makes the IRIs absolute [W3C09]) or ending with a hash character (‘#’).

Map type	OWL IRI
Entity map	<*class URI*>`__ENTITY_MAP`
Attribute map	<*property URI*>`__ATTRIBUTE_MAP`
Identifier map	<*class URI*>`__IDENTIFIER_MAP`
Relation map	<*property URI*>`__RELATION_MAP`
Subtype map	<*class URI*>`__SUBTYPE_MAP`
Translation table of attribute map	<*property URI*>`__ATTRIBUTE_MAP__TRANSLATION_TABLE`
Translation table of subtype map	<*class URI*>`__SUBTYPE_MAP__TRANSLATION_TABLE`

Table 1.1: OWL individual IRIs in OSL

Map type	OWL class IRI
Entity map	`osl:EntityMap`
Attribute map	`osl:AttributeMap`
Identifier map	`osl:IdentifierMap`
Relation map	`osl:RelationMap`
Subtype map	`osl:SubtypeMap`
Translation table	`osl:TranslationTable`

Table 1.2: OWL class membership of map representations in OSL

It is recommended to use that base IRI as `xml:base` XML attribute.
IRIs prefixed with `osl:` are IRIs relative to the IRI
http://w3studi.informatik.uni-stuttgart.de/~martispp/ont# .

[5] An OSL document has to import the following ontology (referred to as "the OSL header" in the following):
http://w3studi.informatik.uni-stuttgart.de/~martispp/ont/db2osl.owl

[6] The OWL individuals described by the OSL document representing the certain types of OBDA maps must have the IRIs specified in table 1.1 (for base IRIs, see paragraph 4). Here, <*class URI*> refers
to the `OWL class URI` field of the respective entity map for entity maps,
to the `OWL class URI` field of the associated entity map for identifier maps,
to the `OWL class URI` field of the associated entity map for subtype maps and
to the `OWL class URI` field of the entity map associated with the respective subtype map for translation tables of subtype maps.
Similarly, <*property URI*> refers
to the `OWL property URI` field of the respective attribute map for attribute maps (or, if it is empty, the value that would have been generated for it if it weren't empty),
to the `OWL property URI` field of the respective relation map for relation maps and
to the `OWL property URI` field of the respective attribute map for translation tables of attribute maps (or, if it is empty, the value that would have been generated for it if it weren't empty).

[7] The OWL individuals described by the OSL document representing the certain types of OBDA maps must be of the OWL types specified in table 1.2 (for base IRIs, see paragraph 4).

[8] The OWL properties described by the OSL document representing the fields of the certain

OBDA maps must have the IRIs specified in table 1.3 (for base IRIs, see paragraph 4).

[9] The following OWL properties in the OSL document refer to lists of elements:

`osl:rm__sourceColumns`

`osl:rm__targetColumns`

`osl:tt__sourceValues`

`osl:tt__rdfRessources`

Therefore, they have the OWL class `osl:StringListNode` as their range, as is required by the OSL header. They must connect the respective individual to an `osl:StringListNode` individual in every case. This "root node" must *not* have an `osl:hasValue` property.
If the represented list is not empty, the list elements are represented by other `osl:StringListNode` individuals connected seriatim by the property `osl:nextNode`, with the first individual being connected to the root node. The node representing the last list element must not have an `osl:nextNode` property.
All nodes except the root node *may* have an `osl:hasValue` property connecting them to their values. The actual list consists of exactly these values, thus, nodes without values are ignored.
It is recommended to enumerate the node IRIs, using 0 for the root node.

Map type	Field label	Field name	OWL IRI
Entity map	E1	Table name	osl:em__tableName
Entity map	E2	Label	osl:em__label
Entity map	E3	Identifier map	osl:em__identifierMap
Entity map	E4	Attribute maps...	osl:em__attributeMaps
Entity map	E5	OWL class URI	osl:em__owlClassURI
Entity map	E6	Description	osl:em__description
Attribute map	A1	Column name	osl:am__columnName
Attribute map	A2	SQL datatype	osl:am__sqlDatatype
Attribute map	A3	Mandatory	osl:am__mandatory
Attribute map	A4	Label	osl:am__label
Attribute map	A5	OWL property URI	osl:am__owlPropertyURI
Attribute map	A6	Property type	osl:am__propertyType
Attribute map	A7	Translation	osl:am__translation
Attribute map	A8	URI pattern	osl:am__uriPattern
Attribute map	A9	RDF language	osl:am__rdfLanguage
Attribute map	A10	XSD datatype	osl:am__xsdDatatype
Attribute map	A11	Description	osl:am__description
Identifier map	I1	Entity map	osl:im__entityMap
Identifier map	I2	Attribute maps...	osl:im__attributeMaps
Identifier map	I3	URI pattern	osl:im__uriPattern
Relation map	R1	Source entity map	osl:rm__sourceEntityMap
Relation map	R2	Source column	osl:rm__sourceColumns
Relation map	R3	Target entity map	osl:rm__targetEntityMap
Relation map	R4	Target column	osl:rm__targetColumns
Relation map	R5	OWL property URI	osl:rm__owlPropertyURI
Subtype map	S1	Entity Map	osl:sm__entityMap
Subtype map	S2	Column Name	osl:sm__columnName
Subtype map	S3	OWL superclass URI	osl:sm__owlSuperclassURI
Subtype map	S4	Prefix	osl:sm__prefix
Subtype map	S5	Suffix	osl:sm__suffix
Subtype map	S6	Translation	osl:sm__translation
Translation table	T1	Source value...	osl:tt__sourceValues
Translation table	T2	RDF ressource...	osl:tt__rdfRessources

Table 1.3: OWL property IRIs in OSL

Bibliography

[SGH+15] Martin G. Skjæveland, Martin Giese, Dag Hovland, Espen H. Lian, and Arild Waaler. "Engineering ontology-based access to real-world data sources". In: *Web Semantics: Science, Services and Agents on the World Wide Web* 33 (2015), pp. 112–140 (cit. on p. 1).

[W3C09] W3C XML Core Working Group. *XML Base (Second Edition)*. https://www.w3.org/TR/xmlbase/. [Accessed: 2016-04-02]. 2009 (cit. on p. 1).

[W3C12] W3C OWL Working Group. *OWL 2 Web Ontology Language - Document Overview (Second Edition)*. https://www.w3.org/TR/owl2-overview/. [Accessed: 2016-04-02]. 2012 (cit. on p. 1).