

An OWL Full Interpretation

Jeremy J. Carroll
HP Labs, Bristol, UK

Abstract

This report is an appendix to report HPL-2008-59. It gives a worked example of the construction used in the proof from that report. For finiteness, a reduced datatype map consisting of only `xsd:boolean` is used. Each of the graphs in the construction is listed explicitly, with some redundancy eliminated. The final Herbrand graph contains about 15,000 triples.

1 Introduction

This is an appendix to [1, 2]. The construction in those reports builds a Herbrand graph in which the property extension of every property is totally explicit. This appendix presents a complete worked example.

The construction is essentially an infinite one. We make it finite in the following ways:

- We have a very small datatype map D , being `{ xsd:boolean }`
- We pretend $L_{\text{plain}} = \{ "" \}$
- We use the ter Horst technique of ignoring every `rdf:_i` except the first; these only appear because of their presence in the RDF and RDFS axioms.

The first of these two are non-conformant. Datatype maps are required to include `xsd:string`, `xsd:integer` and `XMLLiteral`. L_{plain} is infinite. The differences are of no great consequence here.

Each section presents one of the graphs in the construction of section 9 of the two papers. Each graph is given by listing the new triples in that graph that were not in the previous triples.

1.1 Notation

In addition to the abbreviations specified in [2], we have the following: `disjointWith`, `intersectionOf`, `equivalentClass`, `minCardinality`, `maxCardinality`, `cardinality`, `differentFrom`, `AllDifferent`, `distinctMembers`, `FunctionalProperty`, `subClassOf`, `allValuesFrom`, `someValuesFrom`, `ContainerMembershipProperty`, `versionInfo`. and `^^xsd:boolean`.

Some of the triples in the first few sections (5 to 9), are shown with a wavy line under, such as ("0"^^b.type eg:c). This means that the triple is not included in the graphs H_i , i.e. sections 10 to 22.

To avoid too much redundancy we also use the following sets of nodes, identified by their first element, with an overline.

<i>Relating to classes</i>	
$\overline{b_{10}}$	$b_{10}, b_9,$
$\overline{\text{Nothing}}$	Nothing, b_2 , DataRange, DataProp, DeprClass, DeprProp, Ontology, Restrict, Alt, Bag, Seq, Statement, XMLLiteral, Container,
$\overline{b_3}$	$b_3, b_6, b_7,$
$\overline{\text{owl:Class}}$	owl:Class, rdfs:Class,
$\overline{\text{ObjProp}}$	ObjProp, Prop,
$\overline{\text{Thing}}$	Thing, Resource,
<i>Relating to properties</i>	
$\overline{b_{11}}$	$b_{11}, b_9,$
$\overline{\text{priorVers}}$	priorVers, allVals, backCompat, card, hasValue, imports, incompat, maxCard, minCard, onProperty, someVals, versInfo, $_1$, object, predicate, subject, value, comment, isDefinedBy, label, member, seeAlso,

These are used in *metatriples* like $(\overline{\text{owl:Class}}, \text{equivC}, \overline{\text{owl:Class}})$ in the listing. This one stands for four triples in the graph, by taking each member of the set for both subject and object. Some of these may have occurred earlier in the construction.

2 The initial graph

This is given in section 5. It has several features intended to illustrate the action of the construction on literals, which while somewhat artificial in this example, are part of the D-semantics [3], and covered by our method.

- We use non-canonical lexical forms. The notion of ‘canonical’ used in $\mathcal{V}\mathcal{L}_D$ is not, in general, the same as that in XML Schema [4], which provides several different ‘canonical’ forms for the same value depending on which derived datatype is used. So in general, the literal replacement step between G_4 and H_0 is necessary. In our example, it is artificial.
- We use URI and blank nodes which implicitly represent literals; and then use some of these in predicate position.

3 The Function ψ

The function ψ , see section 9.2 of [2], is determined by some D-interpretation of G_4 . We take ψ to be:

$$\psi(x) = \begin{cases} \text{"true"}^{\wedge\wedge\mathbf{b}} & x = \text{"1"}^{\wedge\wedge\mathbf{b}} \\ \text{"false"}^{\wedge\wedge\mathbf{b}} & x = \text{"0"}^{\wedge\wedge\mathbf{b}} \\ \text{"true"}^{\wedge\wedge\mathbf{b}} & x = \text{eg:v} \\ \text{"false"}^{\wedge\wedge\mathbf{b}} & x = b_{15} \\ x & \text{otherwise} \end{cases} \quad (1)$$

Others would be found from different D-interpretations.

4 The Interpretation

The interpretation in section 10 of [2], uses the graph presented in this document, along with the empty sting, the boolean datatype, and its values as the domain of discourse. The function χ is

then given as:

$$\chi(x) = \begin{cases} \text{the boolean datatype} & x = \text{boolean} \\ "" & x = "" \\ \text{TRUE} & x = \text{"true"}^{\wedge\wedge b} \\ \text{FALSE} & x = \text{"false"}^{\wedge\wedge b} \\ x & \text{otherwise} \end{cases} \quad (2)$$

With the last line meaning the occurrences of the appropriate text string in triples (and implicitly in metatriples) in sections 5 to 22, except where deleted by the wavy line. The actual interpretation is built as specified in [2], via the function θ , which we modify, following ter Horst [5] to deal with the `rdf:⌊`:

$$\theta(x) = \begin{cases} \chi(\lrcorner) & x \in \{\text{rdf:}\lrcorner : \lrcorner = 2, 3 \dots\} \\ \chi(x) & x \in \text{nd}(H_{12}) \\ \chi(\psi(x)) & x \in \text{nd}(G_4) \end{cases} \quad (3)$$

5 G_0

<code>("", type, eg:c)</code>	<code>(eg:a, eg:p, eg:v)</code>	<code>(eg:p, range, boolean)</code>
<code>""^{⌊⌋}, type, eg:c)</code>	<code>(eg:b, eg:p, b₁₅)</code>	
<code>(eg:a, eg:p, "1"^{⌊⌋})</code>	<code>(eg:v, eg:v, eg:v)</code>	

6 G_1

<code>(boolean, type, Datatype)</code>	<code>("false"^{⌊⌋}, type, boolean)</code>
<code>("", type, Literal)</code>	<code>("true"^{⌊⌋}, type, boolean)</code>

7 G_2

<code>(nil, type, List)</code>	<code>(XMLLiteral, type, Datatype)</code>	<code>(domain, domain, Prop)</code>
<code>(⌊, type, Prop)</code>	<code>(⌊, domain, Resource)</code>	<code>(isDefinedBy, domain, Resource)</code>
<code>(first, type, Prop)</code>	<code>(first, domain, List)</code>	<code>(label, domain, Resource)</code>
<code>(object, type, Prop)</code>	<code>(object, domain, Statement)</code>	<code>(member, domain, Resource)</code>
<code>(predicate, type, Prop)</code>	<code>(predicate, domain, Statement)</code>	<code>(range, domain, Prop)</code>
<code>(rest, type, Prop)</code>	<code>(rest, domain, List)</code>	<code>(seeAlso, domain, Resource)</code>
<code>(subject, type, Prop)</code>	<code>(subject, domain, Statement)</code>	<code>(subClass, domain, rdfs:Class)</code>
<code>(type, type, Prop)</code>	<code>(type, domain, Resource)</code>	<code>(subPropOf, domain, Prop)</code>
<code>(value, type, Prop)</code>	<code>(value, domain, Resource)</code>	<code>(⌊, range, Resource)</code>
<code>(⌊, type, CMemProp)</code>	<code>(comment, domain, Resource)</code>	<code>(first, range, Resource)</code>
<code>(object, range, Resource)</code>	<code>(isDefinedBy, range, Resource)</code>	<code>(Bag, subClass, Container)</code>
<code>(predicate, range, Resource)</code>	<code>(label, range, Literal)</code>	<code>(Seq, subClass, Container)</code>
<code>(rest, range, List)</code>	<code>(member, range, Resource)</code>	<code>(XMLLiteral, subClass, Literal)</code>
<code>(subject, range, Resource)</code>	<code>(range, range, rdfs:Class)</code>	<code>(CMemProp, subClass, Prop)</code>
<code>(type, range, rdfs:Class)</code>	<code>(seeAlso, range, Resource)</code>	<code>(Datatype, subClass, rdfs:Class)</code>
<code>(value, range, Resource)</code>	<code>(subClass, range, rdfs:Class)</code>	<code>(isDefinedBy, subPropOf, seeAlso)</code>
<code>(comment, range, Literal)</code>	<code>(subPropOf, range, Prop)</code>	
<code>(domain, range, rdfs:Class)</code>	<code>(Alt, subClass, Container)</code>	

8 G_3

(eg:p, type, Prop)	("true"^^b, type, Resource)	(Prop, type, Resource)
(<u>eg:v, type, Prop</u>)	(<u>b₁₅, type, Resource</u>)	(Seq, type, Resource)
(domain, type, Prop)	(eg:a, type, Resource)	(Statement, type, Resource)
(range, type, Prop)	(eg:b, type, Resource)	(XMLLiteral, type, Resource)
(subClass, type, Prop)	(eg:c, type, Resource)	(_1, type, Resource)
(subPropOf, type, Prop)	(eg:p, type, Resource)	(first, type, Resource)
("", type, Resource)	(<u>eg:v, type, Resource</u>)	(nil, type, Resource)
(<u>"0"^^b, type, Resource</u>)	(Alt, type, Resource)	(object, type, Resource)
(<u>"1"^^b, type, Resource</u>)	(Bag, type, Resource)	(predicate, type, Resource)
("false"^^b, type, Resource)	(List, type, Resource)	(rest, type, Resource)
(subject, type, Resource)	(Literal, type, Resource)	(range, type, Resource)
(type, type, Resource)	(Resource, type, Resource)	(seeAlso, type, Resource)
(value, type, Resource)	(comment, type, Resource)	(subClass, type, Resource)
(rdfs:Class, type, Resource)	(domain, type, Resource)	(subPropOf, type, Resource)
(Container, type, Resource)	(isDefinedBy, type, Resource)	(boolean, type, Resource)
(CMemProp, type, Resource)	(label, type, Resource)	
(Datatype, type, Resource)	(member, type, Resource)	

9 G_4

(comment, type, Prop)	(Seq, type, rdfs:Class)	(<u>"1"^^b, type, Literal</u>)
(isDefinedBy, type, Prop)	(Statement, type, rdfs:Class)	("false"^^b, type, Literal)
(label, type, Prop)	(XMLLiteral, type, rdfs:Class)	("true"^^b, type, Literal)
(member, type, Prop)	(rdfs:Class, type, rdfs:Class)	(<u>b₁₅, type, Literal</u>)
(seeAlso, type, Prop)	(Container, type, rdfs:Class)	(<u>eg:v, type, Literal</u>)
(eg:c, type, rdfs:Class)	(CMemProp, type, rdfs:Class)	(<u>"1"^^b, type, boolean</u>)
(Alt, type, rdfs:Class)	(Datatype, type, rdfs:Class)	(<u>b₁₅, type, boolean</u>)
(Bag, type, rdfs:Class)	(Literal, type, rdfs:Class)	(<u>eg:v, type, boolean</u>)
(List, type, rdfs:Class)	(Resource, type, rdfs:Class)	(boolean, subClass, Literal)
(Prop, type, rdfs:Class)	(boolean, type, rdfs:Class)	(_1, subPropOf, member)

10 H_0

("false"^^b, type, eg:c)	("true"^^b, "true"^^b, "true"^^b)	(eg:b, eg:p, "false"^^b)
("true"^^b, type, Prop)	(eg:a, eg:p, "true"^^b)	

11 H_1

(rdfs:Class, type, $\overline{b_3}$)	(priorVers, type, FunProp)	(allVals, type, Prop)
(b_2 , type, b_8)	(priorVers, type, InvFunProp)	(card, type, Prop)
(b_2 , type, b_9)	(backCompat, type, OntProp)	(complmntOf, type, Prop)
(b_3 , type, b_9)	(imports, type, OntProp)	(different, type, Prop)
(b_1 , type, AllDiff)	(incompat, type, OntProp)	(disjoint, type, Prop)
(versInfo, type, AnnProp)	(priorVers, type, OntProp)	(dstnctMems, type, Prop)
(comment, type, AnnProp)	(equivProp, type, SymProp)	(hasValue, type, Prop)
(isDefinedBy, type, AnnProp)	(inverseOf, type, SymProp)	(intersect, type, Prop)
(label, type, AnnProp)	(subClass, type, TransProp)	(inverseOf, type, Prop)
(seeAlso, type, AnnProp)	(subPropOf, type, TransProp)	(maxCard, type, Prop)

(minCard, type, Prop)	(Nothing, type, rdfs:Class)	(b ₃ , b ₁₂ , b ₂)
(onProperty, type, Prop)	(Ontology, type, rdfs:Class)	(Nothing, complmntOf, <u>Thing</u>)
(oneOf, type, Prop)	(Literal, type, Datatype)	(<u>Thing</u> , complmntOf, Nothing)
(someVals, type, Prop)	(b ₂ , b ₈ , b ₄)	(Thing, different, Nothing)
(unionOf, type, Prop)	(b ₃ , b ₈ , b ₂)	(Nothing, disjoint, Nothing)
(b ₂ , type, rdfs:Class)	(b ₈ , b ₈ , b ₂)	(Nothing, disjoint, Resource)
(owl:Class, type, rdfs:Class)	(b ₈ , b ₈ , b ₃)	(Resource, disjoint, Nothing)
(DataRange, type, rdfs:Class)	(b ₂ , b ₉ , b ₃)	(b ₁ , dstnctMems, b ₂)
(DeprClass, type, rdfs:Class)	(b ₄ , b ₁₀ , b ₅)	(b ₁ , dstnctMems, b ₃)
(DeprProp, type, rdfs:Class)	(b ₂ , b ₁₁ , b ₃)	(<u>Thing</u> , equivC, <u>Thing</u>)
(b ₈ , equivProp, b ₈)	(imports, inverseOf, priorVers)	(Thing, sameAs, Thing)
(incompat, equivProp, backCompat)	(incompat, inverseOf, backCompat)	(b ₂ , unionOf, nil)
(priorVers, equivProp, backCompat)	(inverseOf, inverseOf, inverseOf)	(owl:Class, unionOf, b ₂)
(priorVers, equivProp, priorVers)	(priorVers, inverseOf, backCompat)	(rdfs:Class, unionOf, b ₃)
(b ₃ , intersect, b ₅)	(priorVers, inverseOf, priorVers)	(b ₂ , first, rdfs:Class)
(owl:Class, intersect, b ₂)	(b ₂ , oneOf, nil)	(b ₃ , first, rdfs:Class)
(rdfs:Class, intersect, b ₃)	(b ₃ , oneOf, b ₂)	(b ₄ , first, Nothing)
(b ₉ , inverseOf, b ₁₂)	(b ₆ , oneOf, b ₂)	(b ₅ , first, b ₃)
(b ₁₂ , inverseOf, b ₉)	(b ₆ , oneOf, b ₃)	(b ₂ , rest, nil)
(equivProp, inverseOf, equivProp)	(b ₇ , oneOf, b ₃)	(b ₃ , rest, nil)
(b ₄ , rest, nil)	(b ₁₀ , range, b ₈)	(Prop, subClass, Thing)
(b ₅ , rest, nil)	(AnnProp, subClass, Prop)	(rdfs:Class, subClass, owl:Class)
(b ₉ , domain, b ₁₀)	(DataProp, subClass, Prop)	(Resource, subClass, Thing)
(b ₁₀ , domain, b ₈)	(OntProp, subClass, Prop)	(b ₈ , subPropOf, b ₈)
(b ₁₂ , domain, b ₁₀)	(Restrict, subClass, rdfs:Class)	(inverseOf, subPropOf, inverseOf)
(b ₉ , range, b ₁₀)	(Prop, subClass, ObjProp)	(priorVers, subPropOf, b ₉)

12 H_2

(b ₄ , type, b ₈)	(owl:Class, type, owl:Class)	(CMemProp, type, owl:Class)
(b ₅ , type, b ₈)	(FunProp, type, owl:Class)	(Datatype, type, owl:Class)
(b ₂ , type, b ₁₀)	(InvFunProp, type, owl:Class)	(Literal, type, owl:Class)
(b ₃ , type, b ₁₀)	(<u>Nothing</u> , type, owl:Class)	(boolean, type, owl:Class)
(<u>b₃</u> , type, owl:Class)	(<u>ObjProp</u> , type, owl:Class)	("true"^^b, type, ObjProp)
(b ₈ , type, owl:Class)	(OntProp, type, owl:Class)	(b ₈ , type, <u>ObjProp</u>)
(<u>b₁₀</u> , type, owl:Class)	(SymProp, type, owl:Class)	(<u>b₁₀</u> , type, <u>ObjProp</u>)
(eg:c, type, owl:Class)	(<u>Thing</u> , type, owl:Class)	(b ₁₂ , type, <u>ObjProp</u>)
(AllDiff, type, owl:Class)	(TransProp, type, owl:Class)	(eg:p, type, ObjProp)
(AnnProp, type, owl:Class)	(List, type, owl:Class)	(complmntOf, type, ObjProp)
(different, type, ObjProp)	(type, type, ObjProp)	(b ₄ , type, <u>Thing</u>)
(disjoint, type, ObjProp)	(domain, type, ObjProp)	(b ₅ , type, <u>Thing</u>)
(dstnctMems, type, ObjProp)	(range, type, ObjProp)	(b ₈ , type, <u>Thing</u>)
(intersect, type, ObjProp)	(subClass, type, ObjProp)	(<u>b₁₀</u> , type, <u>Thing</u>)
(inverseOf, type, ObjProp)	(subPropOf, type, ObjProp)	(b ₁₂ , type, <u>Thing</u>)
(oneOf, type, ObjProp)	("", type, Thing)	(eg:a, type, Thing)
(priorVers, type, <u>ObjProp</u>)	("false"^^b, type, Thing)	(eg:b, type, Thing)
(unionOf, type, ObjProp)	("true"^^b, type, Thing)	(eg:c, type, Thing)
(first, type, ObjProp)	(b ₁ , type, <u>Thing</u>)	(eg:p, type, Thing)
(rest, type, ObjProp)	(<u>b₃</u> , type, <u>Thing</u>)	(AllDiff, type, <u>Thing</u>)

(AnnProp, type, $\overline{\text{Thing}}$)	(complmntOf, type, $\overline{\text{Thing}}$)	(List, type, Thing)
(owl:Class, type, $\overline{\text{Thing}}$)	(different, type, $\overline{\text{Thing}}$)	(first, type, Thing)
(FunProp, type, $\overline{\text{Thing}}$)	(disjoint, type, $\overline{\text{Thing}}$)	(nil, type, Thing)
(InvFunProp, type, $\overline{\text{Thing}}$)	(dstnctMems, type, $\overline{\text{Thing}}$)	(rest, type, Thing)
(Nothing, type, $\overline{\text{Thing}}$)	(equivProp, type, $\overline{\text{Thing}}$)	(type, type, Thing)
(ObjProp, type, $\overline{\text{Thing}}$)	(intersect, type, $\overline{\text{Thing}}$)	(CMemProp, type, Thing)
(OntProp, type, $\overline{\text{Thing}}$)	(inverseOf, type, $\overline{\text{Thing}}$)	(Datatype, type, Thing)
(SymProp, type, $\overline{\text{Thing}}$)	(oneOf, type, $\overline{\text{Thing}}$)	(Literal, type, Thing)
(Thing, type, $\overline{\text{Thing}}$)	(priorVers, type, $\overline{\text{Thing}}$)	(domain, type, Thing)
(TransProp, type, $\overline{\text{Thing}}$)	(unionOf, type, $\overline{\text{Thing}}$)	(range, type, Thing)
(subClass, type, Thing)	(b_2 , type, List)	(b_5 , type, List)
(subPropOf, type, Thing)	(b_3 , type, List)	(Literal, subClass, Literal)
(boolean, type, Thing)	(b_4 , type, List)	

13 H_3

(b_{11} , type, Prop)	(sameAs, type, Prop)	(sameAs, type, Resource)
(equivC, type, Prop)	(b_{11} , type, Resource)	
(equivProp, type, Prop)	(equivC, type, Resource)	

14 H_4

("true"^^b, domain, Resource)	(equivC, domain, Resource)	(priorVers, domain, AnnProp)
(b_8 , domain, Resource)	(equivProp, domain, Resource)	(priorVers, domain, owl:Class)
(b_{10} , domain, Resource)	(intersect, domain, Resource)	(priorVers, domain, FunProp)
($\overline{b_{11}}$, domain, Resource)	(inverseOf, domain, Resource)	(priorVers, domain, InvFunProp)
(b_{12} , domain, Resource)	(oneOf, domain, Resource)	(priorVers, domain, Nothing)
(eg:p, domain, Resource)	(priorVers, domain, $\overline{b_3}$)	(priorVers, domain, ObjProp)
(complmntOf, domain, Resource)	(priorVers, domain, b_8)	(priorVers, domain, OntProp)
(different, domain, Resource)	(priorVers, domain, $\overline{b_{10}}$)	(priorVers, domain, SymProp)
(disjoint, domain, Resource)	(priorVers, domain, eg:c)	(priorVers, domain, Thing)
(dstnctMems, domain, Resource)	(priorVers, domain, AllDiff)	(priorVers, domain, TransProp)
(priorVers, domain, List)	(domain, domain, Resource)	(complmntOf, range, Resource)
(priorVers, domain, CMemProp)	(range, domain, Resource)	(different, range, Resource)
(priorVers, domain, Datatype)	(subClass, domain, Resource)	(disjoint, range, Resource)
(priorVers, domain, Literal)	(subPropOf, domain, Resource)	(dstnctMems, range, Resource)
($\overline{\text{priorVers}}$, domain, Resource)	("true"^^b, range, Resource)	(equivC, range, Resource)
(priorVers, domain, boolean)	(b_8 , range, Resource)	(equivProp, range, Resource)
(sameAs, domain, Resource)	(b_{10} , range, Resource)	(intersect, range, Resource)
(unionOf, domain, Resource)	($\overline{b_{11}}$, range, Resource)	(inverseOf, range, Resource)
(first, domain, Resource)	(b_{12} , range, Resource)	(oneOf, range, Resource)
(rest, domain, Resource)	(eg:p, range, Resource)	(priorVers, range, $\overline{b_3}$)
(priorVers, range, b_8)	(priorVers, range, OntProp)	(sameAs, range, Resource)
(priorVers, range, $\overline{b_{10}}$)	(priorVers, range, SymProp)	(unionOf, range, Resource)
(priorVers, range, eg:c)	(priorVers, range, Thing)	(rest, range, Resource)
(priorVers, range, AllDiff)	(priorVers, range, TransProp)	(type, range, Resource)
(priorVers, range, AnnProp)	(priorVers, range, List)	(domain, range, Resource)
(priorVers, range, owl:Class)	(priorVers, range, CMemProp)	(range, range, Resource)
(priorVers, range, FunProp)	(priorVers, range, Datatype)	(subClass, range, Resource)
(priorVers, range, InvFunProp)	(priorVers, range, Literal)	(subPropOf, range, Resource)
(priorVers, range, Nothing)	($\overline{\text{priorVers}}$, range, Resource)	(b_2 , subClass, b_2)
(priorVers, range, ObjProp)	(priorVers, range, boolean)	(b_3 , subClass, b_3)

$(b_6, \text{subClass}, b_6)$	$(\text{DataProp}, \text{subClass}, \text{DataProp})$	$(\text{SymProp}, \text{subClass}, \text{SymProp})$
$(b_7, \text{subClass}, b_7)$	$(\text{DeprClass}, \text{subClass}, \text{DeprClass})$	$(\text{Thing}, \text{subClass}, \text{Thing})$
$(b_8, \text{subClass}, b_8)$	$(\text{DeprProp}, \text{subClass}, \text{DeprProp})$	$(\text{TransProp}, \text{subClass}, \text{TransProp})$
$(b_9, \text{subClass}, b_9)$	$(\text{FunProp}, \text{subClass}, \text{FunProp})$	$(\text{Alt}, \text{subClass}, \text{Alt})$
$(b_{10}, \text{subClass}, b_{10})$	$(\text{InvFunProp}, \text{subClass}, \text{InvFunProp})$	$(\text{Bag}, \text{subClass}, \text{Bag})$
$(\text{eg:c}, \text{subClass}, \text{eg:c})$	$(\text{Nothing}, \text{subClass}, \text{Nothing})$	$(\text{List}, \text{subClass}, \text{List})$
$(\text{AllDiff}, \text{subClass}, \text{AllDiff})$	$(\text{ObjProp}, \text{subClass}, \text{ObjProp})$	$(\text{Prop}, \text{subClass}, \text{Prop})$
$(\text{AnnProp}, \text{subClass}, \text{AnnProp})$	$(\text{Ontology}, \text{subClass}, \text{Ontology})$	$(\text{Seq}, \text{subClass}, \text{Seq})$
$(\text{owl:Class}, \text{subClass}, \text{owl:Class})$	$(\text{OntProp}, \text{subClass}, \text{OntProp})$	$(\text{Statement}, \text{subClass}, \text{Statement})$
$(\text{DataRange}, \text{subClass}, \text{DataRange})$	$(\text{Restrict}, \text{subClass}, \text{Restrict})$	$(\text{XMLLiteral}, \text{subClass}, \text{XMLLiteral})$
$(\text{rdfs:Class}, \text{subClass}, \text{rdfs:Class})$	$(b_{12}, \text{subPropOf}, b_{12})$	$(\text{equivProp}, \text{subPropOf}, \text{equivProp})$
$(\text{Container}, \text{subClass}, \text{Container})$	$(\text{eg:p}, \text{subPropOf}, \text{eg:p})$	$(\text{hasValue}, \text{subPropOf}, \text{hasValue})$
$(\text{CMemProp}, \text{subClass}, \text{CMemProp})$	$(\text{allVals}, \text{subPropOf}, \text{allVals})$	$(\text{imports}, \text{subPropOf}, \text{imports})$
$(\text{Datatype}, \text{subClass}, \text{Datatype})$	$(\text{backCompat}, \text{subPropOf}, \text{backCompat})$	$(\text{incompat}, \text{subPropOf}, \text{incompat})$
$(\text{Resource}, \text{subClass}, \text{Resource})$	$(\text{card}, \text{subPropOf}, \text{card})$	$(\text{intersect}, \text{subPropOf}, \text{intersect})$
$(\text{boolean}, \text{subClass}, \text{boolean})$	$(\text{complmntOf}, \text{subPropOf}, \text{complmntOf})$	$(\text{maxCard}, \text{subPropOf}, \text{maxCard})$
$(\text{"true"}^{\wedge\wedge}b, \text{subPropOf}, \text{"true"}^{\wedge\wedge}b)$	$(\text{different}, \text{subPropOf}, \text{different})$	$(\text{minCard}, \text{subPropOf}, \text{minCard})$
$(b_9, \text{subPropOf}, b_9)$	$(\text{disjoint}, \text{subPropOf}, \text{disjoint})$	$(\text{onProperty}, \text{subPropOf}, \text{onProperty})$
$(b_{10}, \text{subPropOf}, b_{10})$	$(\text{dstnctMems}, \text{subPropOf}, \text{dstnctMems})$	$(\text{oneOf}, \text{subPropOf}, \text{oneOf})$
$(b_{11}, \text{subPropOf}, b_{11})$	$(\text{equivC}, \text{subPropOf}, \text{equivC})$	$(\text{priorVers}, \text{subPropOf}, \text{priorVers})$
$(\text{sameAs}, \text{subPropOf}, \text{sameAs})$	$(\text{predicate}, \text{subPropOf}, \text{predicate})$	$(\text{isDefinedBy}, \text{subPropOf}, \text{isDefinedBy})$
$(\text{someVals}, \text{subPropOf}, \text{someVals})$	$(\text{rest}, \text{subPropOf}, \text{rest})$	$(\text{label}, \text{subPropOf}, \text{label})$
$(\text{unionOf}, \text{subPropOf}, \text{unionOf})$	$(\text{subject}, \text{subPropOf}, \text{subject})$	$(\text{member}, \text{subPropOf}, \text{member})$
$(\text{versInfo}, \text{subPropOf}, \text{versInfo})$	$(\text{type}, \text{subPropOf}, \text{type})$	$(\text{range}, \text{subPropOf}, \text{range})$
$(\text{_l}, \text{subPropOf}, \text{_l})$	$(\text{value}, \text{subPropOf}, \text{value})$	$(\text{seeAlso}, \text{subPropOf}, \text{seeAlso})$
$(\text{first}, \text{subPropOf}, \text{first})$	$(\text{comment}, \text{subPropOf}, \text{comment})$	$(\text{subClass}, \text{subPropOf}, \text{subClass})$
$(\text{object}, \text{subPropOf}, \text{object})$	$(\text{domain}, \text{subPropOf}, \text{domain})$	$(\text{subPropOf}, \text{subPropOf}, \text{subPropOf})$

15 H_5

$(b_{11}, \text{type}, \text{ObjProp})$	$(\text{sameAs}, \text{type}, \text{ObjProp})$	$(\text{sameAs}, \text{type}, \text{Thing})$
$(\text{equivC}, \text{type}, \text{ObjProp})$	$(b_{11}, \text{type}, \text{Thing})$	
$(\text{equivProp}, \text{type}, \text{ObjProp})$	$(\text{equivC}, \text{type}, \text{Thing})$	

16 H_6

The nodes of the graph H_5 are: $\{ "", "false"^{\wedge\wedge}b, "true"^{\wedge\wedge}b, b_1, b_2, b_3, b_4, b_5, b_6, b_7, b_8, b_9, b_{10}, b_{11}, b_{12}, \text{eg:a}, \text{eg:b}, \text{eg:c}, \text{eg:p}, \text{AllDiff}, \text{AnnProp}, \text{owl:Class}, \text{DataRange}, \text{DataProp}, \text{DeprClass}, \text{DeprProp}, \text{FunProp}, \text{InvFunProp}, \text{Nothing}, \text{ObjProp}, \text{Ontology}, \text{OntProp}, \text{Restrict}, \text{SymProp}, \text{Thing}, \text{TransProp}, \text{allVals}, \text{backCompat}, \text{card}, \text{complmntOf}, \text{different}, \text{disjoint}, \text{dstnctMems}, \text{equivC}, \text{equivProp}, \text{hasValue}, \text{imports}, \text{incompat}, \text{intersect}, \text{inverseOf}, \text{maxCard}, \text{minCard}, \text{onProperty}, \text{oneOf}, \text{priorVers}, \text{sameAs}, \text{someVals}, \text{unionOf}, \text{versInfo}, \text{Alt}, \text{Bag}, \text{List}, \text{Prop}, \text{Seq}, \text{Statement}, \text{XMLLiteral}, \text{_l}, \text{first}, \text{nil}, \text{object}, \text{predicate}, \text{rest}, \text{subject}, \text{type}, \text{value}, \text{rdfs:Class}, \text{Container}, \text{CMemProp}, \text{Datatype}, \text{Literal}, \text{Resource}, \text{comment}, \text{domain}, \text{isDefinedBy}, \text{label}, \text{member}, \text{range}, \text{seeAlso}, \text{subClass}, \text{subPropOf}, \text{boolean}, \}$.

For any single node n in this set, we add (n, sameAs, n) .

For any pair of distinct nodes n, n' in this set, we add $(n, \text{different}, n')$.

17 H_7

("true"^^b, type, FunProp)	(b_{10} , type, InvFunProp)	(disjoint, type, SymProp)
(b_{10} , type, FunProp)	($\overline{b_{11}}$, type, InvFunProp)	(equivC, type, SymProp)
($\overline{b_{11}}$, type, FunProp)	(b_{12} , type, InvFunProp)	($\overline{\text{priorVers}}$, type, SymProp)
(b_{12} , type, FunProp)	(eg:p, type, InvFunProp)	(sameAs, type, SymProp)
(eg:p, type, FunProp)	(dstnctMems, type, InvFunProp)	("true"^^b, type, TransProp)
($\overline{\text{priorVers}}$, type, FunProp)	($\overline{\text{priorVers}}$, type, InvFunProp)	(b_{10} , type, TransProp)
(sameAs, type, FunProp)	(sameAs, type, InvFunProp)	($\overline{b_{11}}$, type, TransProp)
(first, type, FunProp)	("true"^^b, type, SymProp)	(b_{12} , type, TransProp)
(rest, type, FunProp)	(complmntOf, type, SymProp)	(eg:p, type, TransProp)
("true"^^b, type, InvFunProp)	(different, type, SymProp)	(dstnctMems, type, TransProp)
(equivC, type, TransProp)	(sameAs, type, TransProp)	(eg:c, subClass, Literal)
(equivProp, type, TransProp)	(rest, type, TransProp)	
($\overline{\text{priorVers}}$, type, TransProp)	(eg:c, type, Datatype)	

18 H_8

($\overline{\text{Nothing}}$, complmntOf, $\overline{\text{Thing}}$)	($\overline{b_3}$, disjoint, $\overline{\text{ObjProp}}$)	(b_8 , disjoint, eg:c)
(Thing , complmntOf, $\overline{\text{Nothing}}$)	($\overline{b_3}$, disjoint, OntProp)	(b_8 , disjoint, AllDiff)
($\overline{b_3}$, disjoint, b_8)	($\overline{b_3}$, disjoint, SymProp)	(b_8 , disjoint, AnnProp)
($\overline{b_3}$, disjoint, $\overline{b_{10}}$)	($\overline{b_3}$, disjoint, TransProp)	(b_8 , disjoint, FunProp)
($\overline{b_3}$, disjoint, eg:c)	($\overline{b_3}$, disjoint, List)	(b_8 , disjoint, InvFunProp)
(b_3 , disjoint, AllDiff)	($\overline{b_3}$, disjoint, CMemProp)	(b_8 , disjoint, $\overline{\text{Nothing}}$)
(b_3 , disjoint, AnnProp)	($\overline{b_3}$, disjoint, Datatype)	(b_8 , disjoint, $\overline{\text{ObjProp}}$)
(b_3 , disjoint, FunProp)	($\overline{b_3}$, disjoint, Literal)	(b_8 , disjoint, OntProp)
(b_3 , disjoint, InvFunProp)	($\overline{b_3}$, disjoint, boolean)	(b_8 , disjoint, SymProp)
(b_3 , disjoint, $\overline{\text{Nothing}}$)	(b_8 , disjoint, $\overline{b_3}$)	(b_8 , disjoint, TransProp)
(b_8 , disjoint, CMemProp)	($\overline{b_{10}}$, disjoint, $\overline{\text{Nothing}}$)	(eg:c, disjoint, b_8)
(b_8 , disjoint, Datatype)	($\overline{b_{10}}$, disjoint, $\overline{\text{ObjProp}}$)	(eg:c, disjoint, $\overline{b_{10}}$)
(b_8 , disjoint, Literal)	($\overline{b_{10}}$, disjoint, OntProp)	(eg:c, disjoint, AllDiff)
(b_8 , disjoint, boolean)	($\overline{b_{10}}$, disjoint, SymProp)	(eg:c, disjoint, AnnProp)
($\overline{b_{10}}$, disjoint, b_3)	($\overline{b_{10}}$, disjoint, TransProp)	(eg:c, disjoint, owl:Class)
($\overline{b_{10}}$, disjoint, eg:c)	($\overline{b_{10}}$, disjoint, CMemProp)	(eg:c, disjoint, FunProp)
($\overline{b_{10}}$, disjoint, AllDiff)	($\overline{b_{10}}$, disjoint, Datatype)	(eg:c, disjoint, InvFunProp)
($\overline{b_{10}}$, disjoint, AnnProp)	($\overline{b_{10}}$, disjoint, Literal)	(eg:c, disjoint, $\overline{\text{Nothing}}$)
($\overline{b_{10}}$, disjoint, FunProp)	($\overline{b_{10}}$, disjoint, boolean)	(eg:c, disjoint, $\overline{\text{ObjProp}}$)
($\overline{b_{10}}$, disjoint, InvFunProp)	(eg:c, disjoint, $\overline{b_3}$)	(eg:c, disjoint, OntProp)
(eg:c, disjoint, SymProp)	(AllDiff, disjoint, owl:Class)	(AllDiff, disjoint, Datatype)
(eg:c, disjoint, TransProp)	(AllDiff, disjoint, FunProp)	(AllDiff, disjoint, Literal)
(eg:c, disjoint, List)	(AllDiff, disjoint, InvFunProp)	(AllDiff, disjoint, boolean)
(eg:c, disjoint, CMemProp)	(AllDiff, disjoint, $\overline{\text{Nothing}}$)	(AnnProp, disjoint, $\overline{b_3}$)
(eg:c, disjoint, Datatype)	(AllDiff, disjoint, $\overline{\text{ObjProp}}$)	(AnnProp, disjoint, b_8)
(AllDiff, disjoint, $\overline{b_3}$)	(AllDiff, disjoint, OntProp)	(AnnProp, disjoint, $\overline{b_{10}}$)
(AllDiff, disjoint, b_8)	(AllDiff, disjoint, SymProp)	(AnnProp, disjoint, eg:c)
(AllDiff, disjoint, $\overline{b_{10}}$)	(AllDiff, disjoint, TransProp)	(AnnProp, disjoint, AllDiff)
(AllDiff, disjoint, eg:c)	(AllDiff, disjoint, List)	(AnnProp, disjoint, owl:Class)
(AllDiff, disjoint, AnnProp)	(AllDiff, disjoint, CMemProp)	(AnnProp, disjoint, $\overline{\text{Nothing}}$)

(AnnProp, disjoint, OntProp)	(owl:Class, disjoint, OntProp)	(FunProp, disjoint, Nothing)
(AnnProp, disjoint, List)	(owl:Class, disjoint, SymProp)	(FunProp, disjoint, List)
(AnnProp, disjoint, CMemProp)	(owl:Class, disjoint, CMemProp)	(FunProp, disjoint, Datatype)
(AnnProp, disjoint, Datatype)	(owl:Class, disjoint, Literal)	(InvFunProp, disjoint, $\overline{b_3}$)
(AnnProp, disjoint, Literal)	(owl:Class, disjoint, boolean)	(InvFunProp, disjoint, b_8)
(AnnProp, disjoint, boolean)	(FunProp, disjoint, $\overline{b_3}$)	(InvFunProp, disjoint, $\overline{b_{10}}$)
(owl:Class, disjoint, eg:c)	(FunProp, disjoint, b_8)	(InvFunProp, disjoint, eg:c)
(owl:Class, disjoint, AllDiff)	(FunProp, disjoint, $\overline{b_{10}}$)	(InvFunProp, disjoint, AllDiff)
(owl:Class, disjoint, AnnProp)	(FunProp, disjoint, eg:c)	(InvFunProp, disjoint, Nothing)
(owl:Class, disjoint, Nothing)	(FunProp, disjoint, AllDiff)	(InvFunProp, disjoint, List)
(InvFunProp, disjoint, Datatype)	(Nothing, disjoint, Nothing)	(Nothing, disjoint, boolean)
(Nothing, disjoint, $\overline{b_3}$)	(Nothing, disjoint, ObjProp)	(ObjProp, disjoint, $\overline{b_3}$)
(Nothing, disjoint, b_8)	(Nothing, disjoint, OntProp)	(ObjProp, disjoint, b_8)
(Nothing, disjoint, $\overline{b_{10}}$)	(Nothing, disjoint, SymProp)	(ObjProp, disjoint, $\overline{b_{10}}$)
(Nothing, disjoint, eg:c)	(Nothing, disjoint, Thing)	(ObjProp, disjoint, eg:c)
(Nothing, disjoint, AllDiff)	(Nothing, disjoint, TransProp)	(ObjProp, disjoint, AllDiff)
(Nothing, disjoint, AnnProp)	(Nothing, disjoint, List)	(ObjProp, disjoint, Nothing)
(Nothing, disjoint, owl:Class)	(Nothing, disjoint, CMemProp)	(ObjProp, disjoint, List)
(Nothing, disjoint, FunProp)	(Nothing, disjoint, Datatype)	(ObjProp, disjoint, Datatype)
(Nothing, disjoint, InvFunProp)	(Nothing, disjoint, Literal)	(OntProp, disjoint, $\overline{b_3}$)
(OntProp, disjoint, b_8)	(OntProp, disjoint, Literal)	(SymProp, disjoint, Datatype)
(OntProp, disjoint, $\overline{b_{10}}$)	(OntProp, disjoint, boolean)	(Thing, disjoint, Nothing)
(OntProp, disjoint, eg:c)	(SymProp, disjoint, $\overline{b_3}$)	(TransProp, disjoint, $\overline{b_3}$)
(OntProp, disjoint, AllDiff)	(SymProp, disjoint, b_8)	(TransProp, disjoint, b_8)
(OntProp, disjoint, AnnProp)	(SymProp, disjoint, $\overline{b_{10}}$)	(TransProp, disjoint, $\overline{b_{10}}$)
(OntProp, disjoint, owl:Class)	(SymProp, disjoint, eg:c)	(TransProp, disjoint, eg:c)
(OntProp, disjoint, Nothing)	(SymProp, disjoint, AllDiff)	(TransProp, disjoint, AllDiff)
(OntProp, disjoint, List)	(SymProp, disjoint, owl:Class)	(TransProp, disjoint, Nothing)
(OntProp, disjoint, CMemProp)	(SymProp, disjoint, Nothing)	(TransProp, disjoint, List)
(OntProp, disjoint, Datatype)	(SymProp, disjoint, List)	(TransProp, disjoint, Datatype)
(List, disjoint, $\overline{b_3}$)	(List, disjoint, TransProp)	(CMemProp, disjoint, AnnProp)
(List, disjoint, eg:c)	(List, disjoint, CMemProp)	(CMemProp, disjoint, owl:Class)
(List, disjoint, AllDiff)	(List, disjoint, Datatype)	(CMemProp, disjoint, Nothing)
(List, disjoint, AnnProp)	(List, disjoint, Literal)	(CMemProp, disjoint, OntProp)
(List, disjoint, FunProp)	(List, disjoint, boolean)	(CMemProp, disjoint, List)
(List, disjoint, InvFunProp)	(CMemProp, disjoint, $\overline{b_3}$)	(CMemProp, disjoint, Datatype)
(List, disjoint, Nothing)	(CMemProp, disjoint, b_8)	(CMemProp, disjoint, Literal)
(List, disjoint, ObjProp)	(CMemProp, disjoint, $\overline{b_{10}}$)	(CMemProp, disjoint, boolean)
(List, disjoint, OntProp)	(CMemProp, disjoint, eg:c)	(Datatype, disjoint, $\overline{b_3}$)
(List, disjoint, SymProp)	(CMemProp, disjoint, AllDiff)	(Datatype, disjoint, b_8)
(Datatype, disjoint, $\overline{b_{10}}$)	(Datatype, disjoint, TransProp)	(Literal, disjoint, owl:Class)
(Datatype, disjoint, eg:c)	(Datatype, disjoint, List)	(Literal, disjoint, Nothing)
(Datatype, disjoint, AllDiff)	(Datatype, disjoint, CMemProp)	(Literal, disjoint, OntProp)
(Datatype, disjoint, AnnProp)	(Datatype, disjoint, Literal)	(Literal, disjoint, List)
(Datatype, disjoint, FunProp)	(Datatype, disjoint, boolean)	(Literal, disjoint, CMemProp)
(Datatype, disjoint, InvFunProp)	(Literal, disjoint, $\overline{b_3}$)	(Literal, disjoint, Datatype)
(Datatype, disjoint, Nothing)	(Literal, disjoint, b_8)	(boolean, disjoint, $\overline{b_3}$)
(Datatype, disjoint, ObjProp)	(Literal, disjoint, $\overline{b_{10}}$)	(boolean, disjoint, b_8)
(Datatype, disjoint, OntProp)	(Literal, disjoint, AllDiff)	(boolean, disjoint, $\overline{b_{10}}$)
(Datatype, disjoint, SymProp)	(Literal, disjoint, AnnProp)	(boolean, disjoint, AllDiff)

(boolean, disjoint, AnnProp)	(eg:c, equivC, eg:c)	(TransProp, equivC, TransProp)
(boolean, disjoint, owl:Class)	(AllDiff, equivC, AllDiff)	(List, equivC, List)
(boolean, disjoint, Nothing)	(AnnProp, equivC, AnnProp)	(CMemProp, equivC, CMemProp)
(boolean, disjoint, OntProp)	(owl:Class, equivC, owl:Class)	(Datatype, equivC, Datatype)
(boolean, disjoint, List)	(FunProp, equivC, FunProp)	(Literal, equivC, Literal)
(boolean, disjoint, CMemProp)	(InvFunProp, equivC, InvFunProp)	(boolean, equivC, boolean)
(boolean, disjoint, Datatype)	(Nothing, equivC, Nothing)	(b3, subClass, b3)
(b3, equivC, b3)	(ObjProp, equivC, ObjProp)	(b3, subClass, owl:Class)
(b8, equivC, b8)	(OntProp, equivC, OntProp)	(b3, subClass, Thing)
(b10, equivC, b10)	(SymProp, equivC, SymProp)	(b8, subClass, Thing)
(b8, subClass, List)	(AnnProp, subClass, ObjProp)	(InvFunProp, subClass, TransProp)
(b9, subClass, b10)	(AnnProp, subClass, SymProp)	(Nothing, subClass, b3)
(b10, subClass, b9)	(AnnProp, subClass, Thing)	(Nothing, subClass, b8)
(b10, subClass, owl:Class)	(AnnProp, subClass, TransProp)	(Nothing, subClass, b10)
(b10, subClass, Thing)	(owl:Class, subClass, Thing)	(Nothing, subClass, eg:c)
(b10, subClass, List)	(owl:Class, subClass, rdfs:Class)	(Nothing, subClass, AllDiff)
(eg:c, subClass, Thing)	(FunProp, subClass, ObjProp)	(Nothing, subClass, AnnProp)
(AllDiff, subClass, Thing)	(FunProp, subClass, Thing)	(Nothing, subClass, owl:Class)
(AnnProp, subClass, FunProp)	(InvFunProp, subClass, ObjProp)	(Nothing, subClass, FunProp)
(AnnProp, subClass, InvFunProp)	(InvFunProp, subClass, Thing)	(Nothing, subClass, InvFunProp)
(Nothing, subClass, Nothing)	(Nothing, subClass, boolean)	(SymProp, subClass, Thing)
(Nothing, subClass, ObjProp)	(ObjProp, subClass, Thing)	(Thing, subClass, Resource)
(Nothing, subClass, OntProp)	(ObjProp, subClass, Prop)	(TransProp, subClass, ObjProp)
(Nothing, subClass, SymProp)	(OntProp, subClass, FunProp)	(TransProp, subClass, Thing)
(Nothing, subClass, Thing)	(OntProp, subClass, InvFunProp)	(List, subClass, Thing)
(Nothing, subClass, TransProp)	(OntProp, subClass, ObjProp)	(CMemProp, subClass, FunProp)
(Nothing, subClass, List)	(OntProp, subClass, SymProp)	(CMemProp, subClass, InvFunProp)
(Nothing, subClass, CMemProp)	(OntProp, subClass, Thing)	(CMemProp, subClass, ObjProp)
(Nothing, subClass, Datatype)	(OntProp, subClass, TransProp)	(CMemProp, subClass, SymProp)
(Nothing, subClass, Literal)	(SymProp, subClass, ObjProp)	(CMemProp, subClass, Thing)
(CMemProp, subClass, TransProp)	(Datatype, subClass, Thing)	(boolean, subClass, Thing)
(Datatype, subClass, owl:Class)	(Literal, subClass, Thing)	

19 H_9

(b1, dstnctMems, b4)	(Nothing, intersect, b4)	(b3, unionOf, b5)
(b1, dstnctMems, b5)	(Thing, intersect, nil)	(owl:Class, unionOf, b3)
(b1, dstnctMems, nil)	(b3, oneOf, b3)	(Nothing, unionOf, b4)
(b3, intersect, b5)	(b7, oneOf, b2)	(Nothing, unionOf, nil)
(owl:Class, intersect, b3)	(Nothing, oneOf, nil)	

20 H_{10}

("true"^^b, inverseOf, "true"^^b)	(complmntOf, inverseOf, complmntOf)	(equivC, inverseOf, equivC)
(b11, inverseOf, b12)	(different, inverseOf, different)	(priorVers, inverseOf, priorVers)
(b12, inverseOf, b11)	(disjoint, inverseOf, disjoint)	(sameAs, inverseOf, sameAs)

21 H_{11}

("true"^^b, domain, FunProp)	(b_{10} , domain, Thing)	(b_{12} , domain, List)
("true"^^b, domain, InvFunProp)	(b_{10} , domain, List)	(eg:p, domain, Thing)
("true"^^b, domain, ObjProp)	($\overline{b_{11}}$, domain, b_8)	(complmntOf, domain, $\overline{\text{owl:Class}}$)
("true"^^b, domain, SymProp)	($\overline{b_{11}}$, domain, $\overline{b_{10}}$)	(complmntOf, domain, Thing)
("true"^^b, domain, Thing)	($\overline{b_{11}}$, domain, $\overline{\text{owl:Class}}$)	(different, domain, Thing)
("true"^^b, domain, TransProp)	($\overline{b_{11}}$, domain, Thing)	(disjoint, domain, $\overline{\text{owl:Class}}$)
("true"^^b, domain, Literal)	($\overline{b_{11}}$, domain, List)	(disjoint, domain, Thing)
("true"^^b, domain, boolean)	(b_{12} , domain, b_9)	(dstnctMems, domain, AllDiff)
(b_8 , domain, $\overline{\text{owl:Class}}$)	(b_{12} , domain, $\overline{\text{owl:Class}}$)	(dstnctMems, domain, Thing)
(b_8 , domain, Thing)	(b_{12} , domain, Thing)	(equivC, domain, $\overline{\text{owl:Class}}$)
(equivC, domain, Thing)	($\overline{\text{priorVers}}$, domain, b_8)	($\overline{\text{priorVers}}$, domain, OntProp)
(equivProp, domain, ObjProp)	($\overline{\text{priorVers}}$, domain, $\overline{b_{10}}$)	($\overline{\text{priorVers}}$, domain, SymProp)
(equivProp, domain, Thing)	($\overline{\text{priorVers}}$, domain, eg:c)	($\overline{\text{priorVers}}$, domain, Thing)
(intersect, domain, $\overline{\text{owl:Class}}$)	($\overline{\text{priorVers}}$, domain, AllDiff)	($\overline{\text{priorVers}}$, domain, TransProp)
(intersect, domain, Thing)	($\overline{\text{priorVers}}$, domain, AnnProp)	($\overline{\text{priorVers}}$, domain, List)
(inverseOf, domain, ObjProp)	($\overline{\text{priorVers}}$, domain, $\overline{\text{owl:Class}}$)	($\overline{\text{priorVers}}$, domain, CMemProp)
(inverseOf, domain, Thing)	($\overline{\text{priorVers}}$, domain, FunProp)	($\overline{\text{priorVers}}$, domain, Datatype)
(oneOf, domain, $\overline{\text{owl:Class}}$)	($\overline{\text{priorVers}}$, domain, InvFunProp)	($\overline{\text{priorVers}}$, domain, Literal)
(oneOf, domain, Thing)	($\overline{\text{priorVers}}$, domain, Nothing)	($\overline{\text{priorVers}}$, domain, boolean)
($\overline{\text{priorVers}}$, domain, b_3)	($\overline{\text{priorVers}}$, domain, ObjProp)	(sameAs, domain, Thing)
(unionOf, domain, $\overline{\text{owl:Class}}$)	(subClass, domain, Thing)	("true"^^b, range, boolean)
(unionOf, domain, Thing)	(subPropOf, domain, ObjProp)	(b_8 , range, Thing)
(first, domain, Thing)	(subPropOf, domain, Thing)	(b_8 , range, List)
(rest, domain, Thing)	("true"^^b, range, FunProp)	(b_{10} , range, Thing)
(type, domain, Thing)	("true"^^b, range, InvFunProp)	(b_{10} , range, List)
(domain, domain, ObjProp)	("true"^^b, range, ObjProp)	($\overline{b_{11}}$, range, $\overline{b_{10}}$)
(domain, domain, Thing)	("true"^^b, range, SymProp)	($\overline{b_{11}}$, range, $\overline{\text{owl:Class}}$)
(range, domain, ObjProp)	("true"^^b, range, Thing)	($\overline{b_{11}}$, range, Thing)
(range, domain, Thing)	("true"^^b, range, TransProp)	($\overline{b_{11}}$, range, List)
(subClass, domain, $\overline{\text{owl:Class}}$)	("true"^^b, range, Literal)	(b_{12} , range, b_8)
(b_{12} , range, $\overline{b_{10}}$)	(disjoint, range, Thing)	(inverseOf, range, Thing)
(b_{12} , range, $\overline{\text{owl:Class}}$)	(dstnctMems, range, Thing)	(oneOf, range, Thing)
(b_{12} , range, Thing)	(dstnctMems, range, List)	(oneOf, range, List)
(b_{12} , range, List)	(equivC, range, $\overline{\text{owl:Class}}$)	($\overline{\text{priorVers}}$, range, b_3)
(eg:p, range, Thing)	(equivC, range, Thing)	($\overline{\text{priorVers}}$, range, b_8)
(eg:p, range, Literal)	(equivProp, range, ObjProp)	($\overline{\text{priorVers}}$, range, $\overline{b_{10}}$)
(complmntOf, range, $\overline{\text{owl:Class}}$)	(equivProp, range, Thing)	($\overline{\text{priorVers}}$, range, eg:c)
(complmntOf, range, Thing)	(intersect, range, Thing)	($\overline{\text{priorVers}}$, range, AllDiff)
(different, range, Thing)	(intersect, range, List)	($\overline{\text{priorVers}}$, range, AnnProp)
(disjoint, range, $\overline{\text{owl:Class}}$)	(inverseOf, range, ObjProp)	($\overline{\text{priorVers}}$, range, $\overline{\text{owl:Class}}$)
($\overline{\text{priorVers}}$, range, FunProp)	($\overline{\text{priorVers}}$, range, Datatype)	(type, range, Thing)
($\overline{\text{priorVers}}$, range, InvFunProp)	($\overline{\text{priorVers}}$, range, Literal)	(domain, range, $\overline{\text{owl:Class}}$)
($\overline{\text{priorVers}}$, range, Nothing)	($\overline{\text{priorVers}}$, range, boolean)	(domain, range, Thing)
($\overline{\text{priorVers}}$, range, ObjProp)	(sameAs, range, Thing)	(range, range, $\overline{\text{owl:Class}}$)
($\overline{\text{priorVers}}$, range, OntProp)	(unionOf, range, Thing)	(range, range, Thing)
($\overline{\text{priorVers}}$, range, SymProp)	(unionOf, range, List)	(subClass, range, $\overline{\text{owl:Class}}$)
($\overline{\text{priorVers}}$, range, Thing)	(first, range, $\overline{\text{owl:Class}}$)	(subClass, range, Thing)
($\overline{\text{priorVers}}$, range, TransProp)	(first, range, Thing)	(subPropOf, range, ObjProp)
($\overline{\text{priorVers}}$, range, List)	(rest, range, Thing)	(subPropOf, range, Thing)
($\overline{\text{priorVers}}$, range, CMemProp)	(type, range, $\overline{\text{owl:Class}}$)	

22 H_{12}

$(\text{true}^{\wedge\wedge b}, \text{equivProp}, \text{true}^{\wedge\wedge b})$	$(\text{equivProp}, \text{equivProp}, \text{equivProp})$	$(\text{domain}, \text{equivProp}, \text{domain})$
$(b_{10}, \text{equivProp}, b_{10})$	$(\text{intersect}, \text{equivProp}, \text{intersect})$	$(\text{range}, \text{equivProp}, \text{range})$
$(\overline{b_{11}}, \text{equivProp}, \overline{b_{11}})$	$(\text{inverseOf}, \text{equivProp}, \text{inverseOf})$	$(\text{subClass}, \text{equivProp}, \text{subClass})$
$(b_{12}, \text{equivProp}, b_{12})$	$(\text{oneOf}, \text{equivProp}, \text{oneOf})$	$(\text{subPropOf}, \text{equivProp}, \text{subPropOf})$
$(\text{eg:p}, \text{equivProp}, \text{eg:p})$	$(\overline{\text{priorVers}}, \text{equivProp}, \overline{\text{priorVers}})$	$(\text{true}^{\wedge\wedge b}, \text{subPropOf}, \text{equivProp})$
$(\text{complmntOf}, \text{equivProp}, \text{complmntOf})$	$(\text{sameAs}, \text{equivProp}, \text{sameAs})$	$(\text{true}^{\wedge\wedge b}, \text{subPropOf}, \text{inverseOf})$
$(\text{different}, \text{equivProp}, \text{different})$	$(\text{unionOf}, \text{equivProp}, \text{unionOf})$	$(\text{true}^{\wedge\wedge b}, \text{subPropOf}, \text{sameAs})$
$(\text{disjoint}, \text{equivProp}, \text{disjoint})$	$(\text{first}, \text{equivProp}, \text{first})$	$(\text{true}^{\wedge\wedge b}, \text{subPropOf}, \text{subPropOf})$
$(\text{dstnctMems}, \text{equivProp}, \text{dstnctMems})$	$(\text{rest}, \text{equivProp}, \text{rest})$	$(b_8, \text{subPropOf}, \text{different})$
$(\text{equivC}, \text{equivProp}, \text{equivC})$	$(\text{type}, \text{equivProp}, \text{type})$	$(b_9, \text{subPropOf}, b_{11})$
$(b_{10}, \text{subPropOf}, \text{different})$	$(\text{complmntOf}, \text{subPropOf}, \text{different})$	$(\overline{\text{priorVers}}, \text{subPropOf}, b_{12})$
$(b_{11}, \text{subPropOf}, b_9)$	$(\text{complmntOf}, \text{subPropOf}, \text{disjoint})$	$(\overline{\text{priorVers}}, \text{subPropOf}, \text{eg:p})$
$(\overline{b_{11}}, \text{subPropOf}, \text{different})$	$(\text{dstnctMems}, \text{subPropOf}, \text{different})$	$(\overline{\text{priorVers}}, \text{subPropOf}, \text{complmntOf})$
$(\overline{b_{11}}, \text{subPropOf}, \text{disjoint})$	$(\text{equivC}, \text{subPropOf}, \text{subClass})$	$(\overline{\text{priorVers}}, \text{subPropOf}, \text{different})$
$(\overline{b_{11}}, \text{subPropOf}, \text{subClass})$	$(\text{equivProp}, \text{subPropOf}, \text{subPropOf})$	$(\overline{\text{priorVers}}, \text{subPropOf}, \text{disjoint})$
$(b_{12}, \text{subPropOf}, b_8)$	$(\text{intersect}, \text{subPropOf}, \text{different})$	$(\overline{\text{priorVers}}, \text{subPropOf}, \text{dstnctMems})$
$(b_{12}, \text{subPropOf}, \text{different})$	$(\overline{\text{priorVers}}, \text{subPropOf}, \text{true}^{\wedge\wedge b})$	$(\overline{\text{priorVers}}, \text{subPropOf}, \text{equivC})$
$(b_{12}, \text{subPropOf}, \text{disjoint})$	$(\overline{\text{priorVers}}, \text{subPropOf}, b_8)$	$(\overline{\text{priorVers}}, \text{subPropOf}, \text{equivProp})$
$(b_{12}, \text{subPropOf}, \text{oneOf})$	$(\overline{\text{priorVers}}, \text{subPropOf}, b_{10})$	$(\overline{\text{priorVers}}, \text{subPropOf}, \text{intersect})$
$(\text{eg:p}, \text{subPropOf}, \text{different})$	$(\overline{\text{priorVers}}, \text{subPropOf}, \overline{b_{11}})$	$(\overline{\text{priorVers}}, \text{subPropOf}, \text{inverseOf})$
$(\overline{\text{priorVers}}, \text{subPropOf}, \text{oneOf})$	$(\overline{\text{priorVers}}, \text{subPropOf}, \text{rest})$	$(\overline{\text{priorVers}}, \text{subPropOf}, \text{subPropOf})$
$(\overline{\text{priorVers}}, \text{subPropOf}, \overline{\text{priorVers}})$	$(\overline{\text{priorVers}}, \text{subPropOf}, \text{type})$	$(\text{unionOf}, \text{subPropOf}, \text{different})$
$(\overline{\text{priorVers}}, \text{subPropOf}, \text{sameAs})$	$(\overline{\text{priorVers}}, \text{subPropOf}, \text{domain})$	$(\text{first}, \text{subPropOf}, \text{different})$
$(\overline{\text{priorVers}}, \text{subPropOf}, \text{unionOf})$	$(\overline{\text{priorVers}}, \text{subPropOf}, \text{range})$	$(\text{rest}, \text{subPropOf}, \text{different})$
$(\overline{\text{priorVers}}, \text{subPropOf}, \text{first})$	$(\overline{\text{priorVers}}, \text{subPropOf}, \text{subClass})$	

References

- [1] Carroll, J.J., Turner, D.: The Consistency of OWL Full. Technical Report, HP Labs (2008) HPL-2008-58.
- [2] Carroll, J.J., Turner, D.: The Consistency of OWL Full (with proofs). Technical Report, HP Labs (2008) HPL-2008-59.
- [3] Hayes, P.: RDF Semantics. W3C recommendation, W3C (February 2004) <http://www.w3.org/TR/2004/REC-rdf-mt-20040210/>.
- [4] Malhotra, A., Biron, P.V.: XML Schema Part 2: Datatypes Second Edition. W3C Recommendation, W3C (October 2004) <http://www.w3.org/TR/2004/REC-xmlschema-2-20041028/>.
- [5] ter Horst, H.J.: Completeness, decidability and complexity of entailment for RDF Schema and a semantic extension involving the OWL vocabulary. *J. Web Semantics* **3**(2-3) (2005) 79–115