On the Ostensibly Silent 'W' in OWL 2 RL[⋆]

Aidan Hogan and Stefan Decker

Digital Enterprise Research Institute,
National University of Ireland, Galway
`firstname.lastname@deri.org`

Abstract. In this paper, we discuss the draft OWL 2 RL profile from the perspective of applying the constituent rules over Web data. In particular, borrowing from previous work, we discuss (i) optimisations based on a separation of terminological data from assertional data and (ii) the application of authoritative analysis to constrain third party interference with popular ontology terms. We also provide discussion relating to the applicability of new OWL 2 constructs for two popular Semantic Web ontologies – namely FOAF and SIOC – and provide some evaluation of the proposed use-cases based on reasoning over a representative Web dataset of approx. 12 million statements.

1 Introduction

As more and more data becomes available on the Web, the Semantic Web movement aims to provide technologies which enable greater data-integration and query answering capabilities than the keyword/document centric models prevalent today. The core of these technologies is the Resource Description Framework (RDF) for publishing data in a machine-readable format, wherein there now exist millions of RDF data sources on the Web contributing billions of statements [9]. The Semantic Web technology stack also includes means to supplement instance (assertional) data being published in RDF with ontologies described in RDF Schema (RDFS) [3] and the Web Ontology Language (OWL) [23] (terminological data) providing machines a more sapient understanding of the information – in particular enabling deductive reasoning to be performed.

Reasoning over aggregated Web data is useful, for example, (i) to infer implicit knowledge and thus provide query-answering over a more complete dataset, (ii) to assert equality between equivalent resources resident in remote documents, (iii) to flag inconsistencies wherein conflicting data is provided by one or more parties; and (iv) to execute mappings, where they exist, between different data-models concerned with the same domain. However, reasoning over Web data is indeed an ambitious goal with many inherent difficulties, the most overt of which are (i) the requirement for near-linear scale in execution and (ii) the requirement to be tolerant with respect to noisy and conflicting data (for a detailed treatment of noise in RDF Web data, we refer the interested reader to [15]).

[⋆] The work presented in this paper has been funded in part by Science Foundation Ireland under Grant No. SFI/08/CE/I1380 (Lion-2) and by an IRCSET Scholarship.

With these requirements in mind, in previous work we introduced Scalable Authoritative OWL Reasoner (SAOR) [16]; we discussed the formulation and suitability of a set of rules inspired by pD* [24] – to cover a significant fragment of OWL Full reasoning – for forward-chaining materialisation over Web data. We gave particular focus to scalability and tolerance against noisy Web data showing that, by applying certain practical restrictions, materialisation over a diverse Web dataset – in the order of a billion statements – is feasible.

From the scalability perspective, we introduced a separation of terminological data from assertional data in our rule execution model, based on the premise that terminological data is the most frequently accessed segment of the knowledge base and represents only a small fraction of the overall data.

From the Web tolerance perspective, we presented many issues relating to the effects of noise – which is present in abundance on the Web – on reasoning. We particularly focused on the introduced problem of "ontology hijacking" wherein third-party sources redefine or subsume popular Web ontology terms. Our solution was to include consideration of the source or "context" of data, and provide "authoritative analysis" to curtail the privileges of third-parties.

Drawing on our experiences in reasoning over Web data, in this paper we discuss the new OWL 2 RL draft profile [18]. OWL 2 RL is a fragment of the new OWL 2 language for implementation within rule-based applications; hitherto, there existed only non-standard rule-implementable fragments of OWL reasoning, the mostly prominent thereof being Description Logic Programs (DLP) [11] and pD* [24]. OWL 2 RL extends upon both with a more complete list of rules, including support for a significant fragment of OWL 2 RDF-based semantics [21].

We subsequently present a number of Web use-cases for new OWL 2 terms in the context of two popular Web ontologies: Friend Of A Friend (FOAF) [4] and Semantically Interlinked Online Communities (SIOC) [1,2]; we evaluate our proposed use-cases based on reasoning over a 12m statement Web dataset.

Specifically, in this paper we: (i) discuss a separation of terminological data from assertional data in executing OWL 2 RL/RDF rules; (ii) discuss authoritative reasoning over OWL 2 RL/RDF rules; and (iii) present insights and evaluation on possible deployment of new OWL 2 constructs within two popular Web ontologies – viz.: SIOC and FOAF.

2 OWL 2 RL vs. SAOR

Before we continue, we recall pertinent high-level discussion relating to our previous work on SAOR, and draw parallels to OWL 2 RL (for a more extensive treatment of SAOR, we refer the interested reader to [16]; a full list of SAOR rules is available in [16, Table 2]). In doing so, we provide insights into possible obstacles and optimisations relating to applying OWL 2 RL for materialisation over an RDF dataset collected from the Web.[1] Please note that in Appendix A, we replicate the OWL 2 RL/RDF rules from [18] and denote certain characteristics which we will refer to in this section.

---

[1] Although we focus on forward-chaining applications of OWL 2 RL, much of our discussion has a more general appeal.

SAOR is designed to accept as input a Web knowledge-base in the form of a large body of statements collected by means of a Web crawl, and to output inferred statements by forward-chaining reasoning according to a tailored fragment of OWL; input and inferred statements can then be exploited by a consumer application, such as for query answering. We identified three main aspects around which our system and ruleset is designed and implemented: *computational feasibility* for scalability, *reduced output statements* such that consumer applications are not over-burdened, and finally *Web tolerance* for avoiding undesirable and potentially expensive "inflationary" inferences caused by noisy Web data.

In this section, we will introduce how our ruleset and implementation is designed to adhere to these requirements for Web reasoning, and contrast our approach with the OWL 2 RL ruleset; we begin by discussing general issues.

2.1 High-level Issues

Firstly, SAOR ignores inconsistencies in the data; in OWL 2 RL, inconsistencies are flagged by means of a `false` inference which indicates that the original input graph is inconsistent – such rules could additionally be supported in SAOR. In both cases, the explosive nature of reasoning upon inconsistent data is avoided; i.e., inconsistencies do not lead to the inference of all possible triples.

Secondly, in SAOR we avoid inventing new anonymous individuals. Such invention breaks the upper bound on possible inferable statements from an input graph – $|T|^3$ where T is the union of the set of RDF terms in the input and the set of 'built-in' terms that appear in the rule consequents – and allows for the inference of infinite statements. For example, in [24], pD*sv was introduced which extended pD* with an additional rule based on `owl:someValuesFrom`:[2]

`?v someValuesFrom ?w ; onProperty ?p . ?u a ?v .` ⇒ `?u ?p _:b . _:b a ?w .`

Here, `_:b` is a unique blank-node created for each set of variable bindings from the rule body. Now, given an input graph where a binding for `?w` is a subclass of the respective binding for `?v`, this rule will infer infinite statements; such rules are excluded from pD*, SAOR and OWL 2 RL due to such effects on termination.

In a related matter, in pD*, blank nodes are allowed in all positions in a form of partially-generalised triples; literals are not allowed in subject or predicate positions. Thus, and following RDFS entailment practices [13, Section 7.1], pD* includes rules which invent so called "surrogate blank nodes" to represent literals in subject and predicate positions where they would otherwise be disallowed. Although these blank nodes are formed by a direct mapping from a finite set of literals, they still create new terms and thus in SAOR, we opted to allow literals and blank-nodes in all positions of a triple. This is analogous to the Rule Interchange Format (RIF) [6] and the OWL 2 RL notion of a generalised triple.

Thus far, OWL 2 RL maintains an upper bound of $|T|^3$ inferred generalised statements. However, rules `dt-type2`, `dt-eq` and `dt-diff` (Table 1) are based on an infinite set of literals independent of the input graph. Thus, materialisation according to these rules (which are clearly intended for backward-chaining)

---

[2] In this paper, we use prefixed names as prevalent in the literature and, following Turtle syntax, use '`a`' as a shortcut for `rdf:type` and '`(...)`' for RDF lists.

would lead to inference of infinite triples. One could curtail such inferences by omitting the rules or only applying the rules over literals which appear in the input graph: either would maintain the $|T|^3$ upper bound. Also, rule `dt-eq` could be used to infer equivalence between literals and their canonical versions, introducing at most $|CL|$ terms where CL is the set of literals in the input with lexically distinct canons: the upper bound would then be $(|CL| + |T|)^3$.

Continuing, in SAOR we also aim to omit inference of what we term "extended axiomatic" statements. These include: (i) the set of RDF(S) axiomatic triples [13, Section 4.1]; (ii) the set of additional OWL axiomatic triples listed for pD* [24, Table 6]; and (iii) inferences which apply to every RDF term in the graph. For the latter, we firstly omit rules which assert membership of `rdfs:Resource` for all terms, viz.: RDFS/pD* rules `rdfs4a`/`rdfs4b` [13, Section 7.3]. Secondly, we omit rules which mandate symmetric `owl:sameAs` inferences for all terms, viz: OWL 2 RL rule `eq-ref` (Table 3)[3] and pD* rules `rdfp5a`/`rdfp5b` [24, Table 6]. Such rules immediately add $|T|$ statements to the graph and could be considered inflationary; they are, perhaps, better suited to backward-chaining support (in an approach such as [17]) than materialisation.

Indeed, reasoning involving `owl:sameAs` relations is problematic on the Web: in [14] we found 85,803 equivalent individuals to be inferable from a Web dataset through the incongruous values `08445a31a78661b5c746feff39a9db6e4e2cc5cf` and `da39a3ee5e6b4b0d3255bfef95601890afd80709` for the prominent inverse-functional property `foaf:mbox_sha1sum` – the former value is the sha1-sum of an empty string and the latter is the sha1-sum of the '`mailto:`' string, both of which are erroneously published by online FOAF exporters.[4] Thus, in SAOR, we cross-check the values of inverse-functional properties against a black-list of known noisy values. Also, we disallow `owl:sameAs` inferences from travelling to the predicate position of a triple or to the object position of an `rdf:type` triple: this is contrary to rule `eq-rep-p` in OWL 2 RL, and to the lack of a restriction on rule `eq-rep-o` where `rdf:type` predicates are allowed (Table 5).

Aside from noisy data, naïve materialisation over OWL 2 RL equality rules `eq-ref`, `eq-sym` (Table 3) and `eq-trans` (Table 6) – which axiomatise the reflexive, symmetric, and transitive nature of `owl:sameAs` resp. – leads to quadratic growth in inferences. Again, take for example the 85,803 equivalent individuals we had previously found; naïvely, the OWL 2 RL rules would mandate $85{,}803^2$=7.362b statements to represent the pair-wise equivalences. Also, assuming that each individual was mentioned in, on average, eight unique statements[5], the `eq-rep-*` rules would infer 7.362b * 8 = 59b statements, with massive repetition.

Although the above example again relies on noisy Web data, there do exist valid examples on the Web of large "equivalence chains" of individuals. Again in [14] we discovered a resource representing a "global user" on the `vox.com`

---

[3] One important note: rule `eq-diff1` requires reflexive `owl:sameAs` statements to flag inconsistent reflexive `owl:differentFrom` statements; in the absence of the former, one should support the following rule: `?x :differentFrom ?x .` ⇒ `false`

[4] See, for example `http://blog.livedoor.jp/nkgw/foaf.rdf`

[5] Here, perhaps assuming uniqueness which also considers context.

blogging platform which exports FOAF data; this global user was identified by a blank node and was mentioned in the FOAF profiles of all users.[6] Thus, in our crawl we found 32,390 unique resources, in different documents, with the valid value `http://team.vox.com/` for inverse-functional property `foaf:weblog`. Again, such would lead to the inference of over a billion `owl:sameAs` statements and billions more statements in duplicative data.

Taking such considerations into account, in order to avoid an explosion of repetitious inferences in [14,16] we instead choose a single 'pivot' identifier for identifying equivalent individuals. Thus, we compress repetitive entries into one single entry; we also store equivalence relations from the pivot element to all other identifiers such that the fully expanded view can be realised by the consumer application using backward-chaining techniques.

Finally, there are two cardinality-related rules supported in SAOR for which no equivalent rule exists in OWL 2 RL; namely **rdfc2** (Table 9) and an exact-cardinality version of **cls-maxc2** (Table 6). Their omission relates to the constraint that OWL 2 RL documents must also be valid OWL 2 DL documents [22, Section 2.1] which enforces certain computational guarantees, e.g., for entailment checking. Thus, the OWL 2 RL ruleset omits exact-cardinality versions of rules for **cls-maxc*** and **cls-maxqc*** (Table 6) and support for minimum-cardinality; also missing are rules relating to disjoint-union expressions, which could be supported analogously to union-of and disjoint-class expressions (resp. **cls-duni1** and **cls-duni2** in Table 9). More puzzlingly, the ruleset omits support for self-restriction expressions which are supported by OWL 2 DL/EL; the omitted rules (**cls-hs*** in Table 9) are reciprocal of those for has-value expressions (**cls-hv*** in Table 4); motivation for the omission is missing from the draft documents.[7]

In terms of Web reasoning, one other notable consequence of enforcing OWL 2 DL restrictions in OWL 2 RL documents is the forbiddance of inverse-functional datatype-properties [19, Section 9.2.8]: the definition of such properties is common on the Web; examples include `foaf:mbox_sha1sum` and various FOAF chat ID properties whereby the former is commonly used as a primary means of identifying `foaf:Person` members without using URIs.

In summary, although the OWL 2 RL profile does not introduce new individuals, and although sound and complete when applied to a valid OWL 2 RL document, the profile is not immediately suited to application over Web data. Indeed, a Web reasoner should perhaps consider abandoning *completeness* guarantees for a more syntactically permissive, semantically inclusive and practicable (albeit, possibly *incomplete*) approach: e.g., allowing inverse-functional datatype-properties, including omitted rules as exemplified in Table 9 and curtailing quadratic equivalence inferencing on the Web.

---

[6] See `http://team.vox.com/profile/foaf.rdf` for the RDF description of the resource with `foaf:nick "Team Vox"` and, e.g., `http://danbri.vox.com/profile/foaf.rdf` as an example of a user profile, all of which reference the `Team Vox` user.

[7] We can only conjecture that this is perhaps related to a possible `owl:hasSelf` emulation of an `owl:ReflexiveProperty` expression which is not supported.

2.2 Separating terminological data

The main optimisation of SAOR, and indeed the main divergence from traditional rules engines, is in considering a distinction between terminological data and assertional data. Herein, we refer to terminological data as the segment of the Web crawl which deals with class and property descriptions – using RDF(S) and OWL terms – that are supported by the given ruleset.

In [16], we showed that <1% of data in our large Web dataset represented terminological data; however, this small segment of data is the most frequently accessed for reasoning, with most rules including terminological expressions in their antecedents. For example, the FOAF ontology currently contains 559 triples, the majority of which we would consider to be terminological; however, there exists hundreds of millions[8] of statements on the Web which use the properties and classes defined by the former 559 triples. Based on such observations, we optimise access to the terminological data; we perform an initial scan of the dataset and extract terminological statements while building an in-memory hashtable representation of this information which we call our "TBox".

In creating an in-memory TBox, for which the terminological information required by each rule can be accessed in $\mathcal{O}(1)$ (in practical terms, considering our hashtable-based implementation), we significantly reduce the implementational complexity of all rules requiring terminological knowledge. Also, since we only index <1% of the data, the cost of building the hashtable is relatively low. In [16], we categorised our rules according to their terminological and assertional arity; i.e., the amount of patterns in the rule that could be answered by the TBox and the amount that could not. In particular, we identified eighteen rules which required zero or one assertional patterns and thus, could be serviced by statement-wise scan of the entire (possibly unsorted) dataset.

Take for example the following rule:

$$\underline{\texttt{?c owl:intersectionOf (?c}_1\texttt{ ... ?c}_n\texttt{) .}}\ \texttt{?x a ?c .} \Rightarrow \texttt{?x a ?c}_1\texttt{, ..., ?c}_n\texttt{ .}$$

Herein, the terminological patterns serviceable by the TBox are underlined. To execute this rule, the dataset can be scanned statement-by-statement, with triples satisfying the `?x a ?c .` pattern joined with the TBox; inferred statements can be recursively joined with the TBox in the same fashion. Thus, we can execute such rules using two scans of the unsorted data; the first builds the TBox and the second executes the rules (again, cf. [16] for more detail).

However, there exist a number of rules which contain more than one purely assertional pattern in the antecedent, and thus require execution of joins on the arbitrarily large ABox – and even worse – exhaustive application on all inferred ABox triples. Such rules are more expensive to compute and require indexing of a much larger portion of the data; in [16], we presented means to execute such rules using static sorted indices; however, such an approach encountered difficulties in achieving termination and is better suited to approximative reasoning. In any case, we showed that the majority of inferences for our Web dataset were covered by the set of rules with zero or one assertional pattern (<0.3% of inferences were

---

[8] E.g., see `http://vmlion25.deri.ie/`; the figure could however be in the billions.

found through rules with more than one assertional pattern[9]). Subsequently, using the rules with a low assertional arity, we demonstrating reasoning over 1.1b statements, crawled from 665k Web documents, in <16.5 hours.

Following from this, Appendix A lists OWL 2 RL rules in order of increasing complexity, starting with rules with no antecedent ($\mathcal{R}0$) and ending with rules with a variable number of assertional patterns ($\mathcal{R}6$-7). In practical terms, rules in $\mathcal{R}0$-3 present an opportunity for near-linear scale with respect to Web reasoning in a system such as SAOR (at least, given observable trends in Web data); rules in $\mathcal{R}4$-5 require assertional joins (with an upper-bound of five conjunctive patterns for rule **cls-maxqc4**), which are more expensive to compute at Web scale; rules in $\mathcal{R}6$-7 may present Web reasoners with the daunting task of computing arbitrarily-large conjunctive- assertional-patterns – Web reasoners would probably have to enforce maximally supported lengths for such expressions.

2.3 Authoritative reasoning

In preliminary work on SAOR, we encountered a puzzling deluge of inferences from naïve reasoning over a Web dataset. For example, we found that reasoning on a single membership assertion of `owl:Thing` – apparently the "top-level" concept – caused 4,251 inferences when naïve reasoning was applied to the Web dataset.[10] Again for example, the document `http://www.eiao.net/rdf/1.0` defines 9 *properties* to be in the domain of `rdf:type` [15].[11]

The problem is more widespread than core RDF(S)/OWL terms; for one membership assertion of `foaf:Person`, naïve reasoning created 4,631 inferences (an additional 380 inferences on top of `owl:Thing`) as opposed to the six inferences mandated by the FOAF ontology. As another example, there are multiple documents which declare the class `foaf:Image` to be an `owl:ObjectProperty` [15].[12]

In [16], we termed such third party redefinition of ontology terms "ontology hijacking" and proposed a solution to counter such behaviour based upon analysis of "authoritative sources" for terminological data:

Definition 1 (Authoritative Source). *A Web document from source (context) c speaks authoritatively about an RDF term n iff:*

1. *n is a blank node; or*
2. *n is a URI and c coincides with, or is redirected to by, the namespace of n.*

We then defined our notion of an "authoritative rule application" whereby, here paraphrasing, each Web document satisfying a terminological pattern in the antecedent must speak authoritatively for at least one term appearing in both the assertional and terminological parts of the antecedent; e.g., take the rule:

---

[9] This figure is, however, increasing with popular usage of transitive properties.

[10] E.g., the document `http://lsdis.cs.uga.edu/~{}oldham/ontology/wsag/wsag.owl` accounts for 55 such inferences where `owl:Thing` is a member of 55 union class descriptions.

[11] Such usage of `owl:Thing` is prohibited by the structural syntax of OWL 2 RL [19].

[12] E.g., see `http://wiki.sembase.at/index.php/Special:ExportRDF/Dieter_Fensel`

$$\underline{\texttt{?p rdfs:domain ?c .}}\ \texttt{?x ?p ?y .} \Rightarrow \texttt{?x a ?c .}$$

Here, the term matched by `?p` must be authoritatively spoken for by the document serving the `rdfs:domain` triple. Therefore, taking the previous example where nine domains for `rdf:type` are non-authoritatively defined, the document `http://www.eiao.net/rdf/1.0` does not speak authoritatively for `rdf:type`, which is bound by `?p`: thus, no inference takes place.

Of course, we still allow extension of external ontologies, whereby memberships of local terms are translated into memberships of remote terms, but not vice-versa; e.g., for the above rule, authoritative reasoning will still permit a triple such as `ex:sibling rdfs:domain foaf:Person .` when served in a location authoritative for the `ex:` namespace, facilitating translation from subject-members of the local property `ex:sibling` to the remote class `foaf:Person`.

Along these lines, Tables 4-9 indicate authoritative variables for the OWL 2 RL rules in bold-face; when enforced, the document(s) serving the terminological statements must speak authoritatively for *at least one* binding of an authoritative variable for the rule to fire.

3 Web Use-cases

The OWL 2 New Features and Rationale document [10] is intended to provide rationale and use-cases for novel OWL 2 features; in particular, the document defines 19 use-cases which motivate new features. However, the document focuses largely on domain-specific use-cases, with, e.g., nine use-cases tied to the Health Care and Life Sciences (HCLS) domain. In this section, we briefly look at how new OWL 2 features could be exploited on the Web by investigating possible pragmatic extensions of two prominent Web ontologies; namely Friend Of A Friend (FOAF) and Semantically Interlinked Online Communities (SIOC).

FOAF is a lightweight ontology providing classes and properties for describing personal information and resources; these terms are amongst the most commonly instantiated on the Web [9, Table 1&2], with many blogging platforms and social networks providing automatic exports of user profiles in FOAF. SIOC [2], similarly, is a lightweight ontology for describing and connecting resources relating to online social communities and the various platforms for information dissemination on the Web; SIOC reuses terms from other Web ontologies, including FOAF. SIOC terms have more recently seen a large growth in popularity as large-volume exporters have become available.[13]

Both ontologies are pragmatically lightweight to foster uptake amongst non-expert communities; we follow such precedent – e.g., we ignore the new disjoint-union qualified-cardinality and self-restriction constructs since both FOAF and SIOC have previously avoided complex class descriptions from the original OWL specification – and select the following novel OWL 2 constructs as possible targets for use in FOAF and/or SIOC: (i) `IrreflexiveProperty`/`AsymmetricProperty`; (ii) `propertyChainAxiom`; and (iii) `hasKey`.

---

[13] See `http://vmlion25.deri.ie/` for a recent survey of the terms in a >1bn statement Web dataset

In order to provide insights into the fecundity of our proposed extensions, we perform reasoning over a representative Web dataset. We retrieved this dataset from the Web in April 2009 by means of a Web crawl using MultiCrawler [12]; beginning with Tim Berners-Lee's FOAF file[14], we performed a seven-hop breadth first crawl for RDF/XML files; after each hop, we extracted all URIs from the crawled data as input for the next hop. Finally, we restricted the crawl according to pay-level-domains; we enforced a maximum of 5,000 crawled documents from each domain so as to ensure a diverse and representative dataset. The crawl consisted of access to 149,057 URIs, and acquired 54,836 (36.8%) valid RDF/XML documents containing 12,534,481 RDF statements; of these, 3,751,617 statements (29.9%) contain a URI in the FOAF namespace and 782,188 (6.2%) contain a URI in a SIOC namespace.

Property constraints We now look at asymmetrical/irreflexive property constraints: we take precedent from the current `owl:disjointWith` assertions in FOAF and SIOC which analogously provide simple means of consistency checking. Firstly, please note that all asymmetric properties are irreflexive. Also, we only select properties whose irreflexivity was not already implicitly constrained by disjoint domain/range assertions; we exclude datatype properties, and, e.g., we exclude `workplaceHomepage` since the domain (`Person`) and range (`Document`) are disjoint and symmetric or reflexive relations thereof would already be flagged. We chose 6 FOAF properties and 17 SIOC properties as being implicitly assymetric/irreflexive and one SIOC property as being irreflexive alone: `sibling`.

Applying these constraints to our Web dataset, we found 319 `sioc:link`[15], 2 `foaf:holdsAcccount` and 1 `foaf:mbox` reflexive statements. We found no examples of symmetric statements for the asymmetric properties. Although the results are less than convincing, the asymmetric/irreflexive constraints would constitute a straightforward extension of the FOAF/SIOC ontologies; please note that for authoritative reasoning, these constraints should be provided in the FOAF/SIOC ontologies for the FOAF/SIOC terms respectively.

Property chains The `owl:propertyChainAxiom` allows for defining arbitrarily long paths which, when present, succinctly infer membership of a single property.

The main use-case we envisage for this construct relates to translating SIOC attributes attached to an instance of `sioc:User` into FOAF attributes attached to `foaf:Agent`. The FOAF profile defines a class `foaf:OnlineAccount` intended to represent the online presence of a `foaf:Agent` through the `foaf:holdsAccount` relation; SIOC defines `sioc:User`, a subclass of `foaf:OnlineAccount`, and provides a more expressive vocabulary for defining and connecting the `sioc:User` with online services. Thus, we can translate from the SIOC attributes for `sioc:User/foaf:OnlineAccount` to equivalent FOAF properties with a domain of `foaf:Agent` or an encompassing class thereof.

---

[14] `http://www.w3.org/People/Berners-Lee/card`

[15] The textual description of the `link` property recommends irreflexive use [1]. Most such reflexive statements are produced by the SIOC WordPress exporter; e.g., see `http://dowhatimean.net/?sioc_type=site`

Another possible use-case is to formally realise the informal semantics of `sioc:topic` which state that "...a Container will have an associated topic or set of topics that can be propagated to the Items it contains" [1]: this propagation can be implemented using OWL 2 property chains.

Figure 1 depicts the envisaged translations.; to take an example, the assertion `foaf:nick :propertyChainAxiom (foaf:holdsAccount, sioc:name) .` made in the FOAF or SIOC ontology would allow for authoritative translation of `sioc:name` values attached to `sioc:User`s into `foaf:nick` values attached to `foaf:Agent`s.[16]

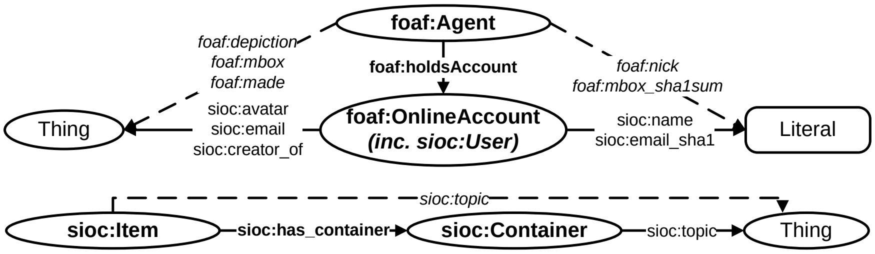

Fig. 1. FOAF/SIOC property chain translations

Applying the above chains to our dataset, we found 29,617 inferences; viz.: 29,373 `foaf:nick`, 216 `foaf:depiction`, 20 `foaf:mbox` and 8 `foaf:mbox_sha1sum` statements respectively. Here it seems, the only practically convincing use-case is the translation of `sioc:name` values into `foaf:nick` values, although perhaps with the growth of online SIOC data, the above figures may begin to increase.

Complex keys We examine one last use-case for the new OWL 2 constructs; namely the `owl:hasKey` construct which is used to define a set of properties whose values together uniquely identify a member of the specified class. We foresee one possible use-case which again lies on the intersection of FOAF and SIOC: members of `foaf:OnlineAccount`, and thereby of `sioc:User`, are uniquely defined by the properties `foaf:accountName` and `foaf:accountServiceHomepage` together.

Applying the above key to our dataset, we found 4,576,310 non-reflexive `:sameAs` inferences (only includes inferences from application of **prp-key** and not of, e.g., **eq-trans**) mentioning 78,534 individuals, with the longest equivalence chain containing 723 individuals. Due to rare usage of URIs for `OnlineAccount` members, complex keys are the only solution currently available to uniquely identify and aggregate such resources.

4 Related Work

Several rule expressible non-standard OWL fragments; namely OWL-DLP [11], OWL^- [7] (which is a slight extension of OWL-DLP), OWLPrime [25] and pD* [24]; have been defined in the literature and enable incomplete but sound RDFS and OWL Full inferences.

---

[16] This translation seems a neat fit: the informal description of `foaf:nick` states that it is for values "such as those use [sic.] in IRC chat, online accounts, and computer logins" [4].

Some existing systems already implement a separation of TBox and ABox for scalable reasoning, where in particular, assertional data are stored in some RDBMS; e.g., Hawkeye [20] demonstrates reasoning over a 166m triple Web dataset – however, they use a prescribed TBox. Also, like us, they internally choose pivot identifiers to represent equivalent sets of individuals.

Work presented in [5] introduced the notion of an *authoritative description* which is very similar to ours; however, we provide much more extensive treatment of the issue, supporting a much more varied range of RDF(S)/OWL constructs.

One promising alternative to authoritative reasoning for the Web is the notion of "context-dependant" or "quarantined reasoning" introduced in [8], whereby inference results are only considered valid (quarantined) within the given context of a document.

5 Conclusion

In this paper, we have presented discussion relating to applying OWL 2 RL over Web data. In particular, we discussed a separation of terminological data from purely assertional data wherein terminological data represents a small fraction of an overall Web dataset and is the most frequently accessed during reasoning. We presented a categorisation of OWL 2 RL rules based on terminological/assertional arity and discussed the implementational feasibility of said categories. We also discussed authoritative reasoning, which heeds the source of information when making inferences, thus countering unwanted third-party contributions. We identified those variable positions present in the OWL 2 RL/RDF rules which should be authoritatively restricted to counter-act ontology hijacking. Finally, motivated by a lack of Web reasoning discussion in the official specifications, we presented a number of Web use-cases for OWL 2 terms based on two popular Web ontologies: viz. SIOC and FOAF. Although some of the use-cases were not convincing when presented with a real Web dataset, we found some practical deployment for the `owl:propertyChainAxiom` and `owl:hasKey` constructs. In any case, our purview was limited to that of SIOC and FOAF, and we conclude that new OWL 2 terms may find more productive application in other/future Web ontologies.

References

1. U. Bojārs and J. G. Breslin. SIOC Core Ontology Specification, Jan. 2009. `http://rdfs.org/sioc/spec/`.
2. J. G. Breslin, A. Harth, U. Bojars, and S. Decker. Towards semantically-interlinked online communities. In *ESWC*, pages 500–514, 2005.
3. D. Brickley and R. Guha. RDF vocabulary description language 1.0: RDF Schema. W3C Recommendation, Feb. 2004. http://www.w3.org/TR/rdf-schema/.
4. D. Brickley and L. Miller. FOAF Vocabulary Specification 0.91, Nov. 2007. `http://xmlns.com/foaf/spec/`.
5. G. Cheng, W. Ge, H. Wu, and Y. Qu. Searching semantic web objects based on class hierarchies. In *Proceedings of Linked Data on the Web Workshop*, 2008.

6. J. de Bruijn. RIF RDF and OWL Compatibility, July 2009. W3C Working Draft, available at `http://www.w3.org/TR/rif-rdf-owl/`.
7. J. de Bruijn, A. Polleres, R. Lara, and D. Fensel. OWL$^-$. Technical Report WSML d20.1v0.2, 2005.
8. R. Delbru, A. Polleres, G. Tummarello, and S. Decker. Context dependent reasoning for semantic documents in Sindice. In *Proc. of the 4th Int. Workshop on Scalable Semantic Web Knowledge Base Systems (SSWS 2008)*, October 2008.
9. L. Ding and T. Finin. Characterizing the Semantic Web on the Web. In *Proceedings of the 5th International Semantic Web Conference*, November 2006.
10. C. Golbreich and E. K. Wallace. OWL 2 New Features and Rationale. W3C Working Draft, 2009. `http://www.w3.org/TR/owl2-new-features/`.
11. B. Grosof, I. Horrocks, R. Volz, and S. Decker. Description logic programs: Combining logic programs with description logic. In *13th International Conference on World Wide Web*, 2004.
12. A. Harth, J. Umbrich, and S. Decker. Multicrawler: A pipelined architecture for crawling and indexing semantic web data. In *5th International Semantic Web Conference*, pages 258–271, 2006.
13. P. Hayes. RDF semantics. W3C Recommendation, Feb. 2004. http://www.w3.org/TR/rdf-mt/.
14. A. Hogan, A. Harth, and S. Decker. Performing object consolidation on the semantic web data graph. In *I3: Identity, Identifiers, Identification Workshop*, 2007.
15. A. Hogan, A. Harth, A. Passant, S. Decker, and A. Polleres. Weaving the Pedantic Web. Technical report, DERI Galway, 2009. `http://www.deri.ie/fileadmin/documents/DERI-TR-2009-07-28.pdf`.
16. A. Hogan, A. Harth, and A. Polleres. Scalable Authoritative OWL Reasoning for the Web. *Int. Journal on Semantic Web and Information Systems*, 5(2), 2009.
17. G. Lukácsy and P. Szeredi. Efficient description logic reasoning in prolog: The dlog system. *CoRR*, abs/0904.0578, 2009.
18. B. Motik, B. C. Grau, I. Horrocks, Z. Wu, A. Fokoue, and C. Lutz. OWL 2 Web Ontology Language Profiles, June 2009. W3C Candidate Recommendation, available at `http://www.w3.org/TR/owl2-profiles/`.
19. B. Motik, P. F. Patel-Schneider, and B. Parsia. OWL 2 Web Ontology Language Structural Specification and Functional-Style Syntax, June 2009. W3C Candidate Recommendation, available at `http://www.w3.org/TR/owl2-syntax/`.
20. Z. Pan, A. Qasem, S. Kanitkar, F. Prabhakar, and J. Heflin. Hawkeye: A practical large scale demonstration of semantic web integration. In *OTM Workshops*, 2007.
21. M. Schneider. OWL 2 RDF-Based Semantics. W3C Candidate Recommendation, June 2009. `http://www.w3.org/TR/owl2-rdf-based-semantics/`.
22. M. Smith, I. Horrocks, and M. Krötzsch. OWL 2 Web Ontology Language Conformance, June 2009. W3C Candidate Recommendation, available at `http://www.w3.org/TR/owl2-conformance/`.
23. M. K. Smith, C. Welty, and D. L. McGuinness. OWL Web Ontology Language Guide. W3C Recommendation, Feb. 2004. http://www.w3.org/TR/owl-guide/.
24. H. J. ter Horst. Completeness, decidability and complexity of entailment for RDF Schema and a semantic extension involving the OWL vocabulary. *Journal of Web Semantics*, 3:79–115, 2005.
25. Z. Wu, G. Eadon, S. Das, E. I. Chong, V. Kolovski, M. Annamalai, and J. Srinivasan. Implementing an Inference Engine for RDFS/OWL Constructs and User-Defined Rules in Oracle. In *24th Int. Conf. on Data Engineering*. IEEE, 2008.

A Rule Tables

In this Section, we provide Tables 1-9 for reference (which include all OWL 2 RL rules) presented in Turtle-like syntax; the default namespace refers to `owl:`. Rules are categorised according to increasing terminological/assertional antecedent arity; authoritative variable positions are denoted using bold-face.

$\mathcal{R}0$: no antecedent			
OWL2RL	**SAOR**	**Consequent**	**Notes**
prp-ap	**-**	?ap a :AnnotationProperty .	For each built-in annotation property
cls-thing	**-**	:Thing a :Class .	-
cls-nothing	**-**	:Nothing a :Class .	-
dt-type1	**-**	?dt a rdfs:Datatype .	For each built-in datatype
dt-type2	**-**	?l a ?dt .	For all ?l in the value space of datatype ?dt
dt-eq	**-**	?l$_1$:sameAs ?l$_2$.	For all ?l$_1$ and ?l$_2$ with the same data value
dt-diff	**-**	?l$_1$:differentFrom ?l$_2$.	For all ?l$_1$ and ?l$_2$ with different data values

Table 1. Rules with no antecedent

$\mathcal{R}1$: only terminological patterns in antecedent			
OWL2RL	**SAOR**	**Antecedent** (terminological)	**Consequent**
cls-00	**rdfc0**	?c :oneOf (?x$_1$ ... ?x$_n$) .	?x$_1$ ... ?x$_n$ a ?c .
scm-cls	**-**	?c a :Class .	?c rdfs:subClassOf ?c , :Thing ; :equivalentClass ?c . :Nothing rdfs:subClassOf ?c .
scm-sco	**-**	?c$_1$ rdfs:subClassOf ?c$_2$. ?c$_2$ rdfs:subClassOf ?c$_3$.	?c$_1$ rdfs:subClassOf ?c$_3$.
scm-eqc1	**-**	?c$_1$:equivalentClass ?c$_2$.	?c$_1$ rdfs:subClassOf ?c$_2$. ?c$_2$ rdfs:subClassOf ?c$_1$.
scm-eqc2	**-**	?c$_1$ rdfs:subClassOf ?c$_2$. ?c$_2$ rdfs:subClassOf ?c$_1$.	?c$_1$:equivalentClass ?c$_2$.
scm-op	**-**	?p a :ObjectProperty .	?p rdfs:subPropertyOf ?p . ?p :equivalentProperty ?p .
scm-dp	**-**	?p a :DatatypeProperty .	?p rdfs:subPropertyOf ?p . ?p :equivalentProperty ?p .
scm-spo	**-**	?p$_1$ rdfs:subPropertyOf ?p$_2$. ?p$_2$ rdfs:subPropertyOf ?p$_3$.	?p$_1$ rdfs:subPropertyOf ?p$_3$.
scm-eqp1	**-**	?p$_1$:equivalentProperty ?p$_2$.	?p$_1$ rdfs:subPropertyOf ?p$_2$. ?p$_2$ rdfs:subPropertyOf ?p$_1$.
scm-eqp2	**-**	?p$_1$ rdfs:subPropertyOf ?p$_2$. ?p$_2$ rdfs:subPropertyOf ?p$_1$.	?p$_1$:equivalentProperty ?p$_2$.
scm-dom1	**-**	?p rdfs:domain ?c$_1$. ?c$_1$ rdfs:subClassOf ?c$_2$.	?p rdfs:domain ?c$_2$.
scm-dom2	**-**	?p$_2$ rdfs:domain ?c . ?p$_1$ rdfs:subPropertyOf ?p$_2$.	?p$_1$ rdfs:domain ?c .
scm-rng1	**-**	?p rdfs:range ?c$_1$. ?c$_1$ rdfs:subClassOf ?c$_2$.	?p rdfs:range ?c$_2$.
scm-rng2	**-**	?p$_2$ rdfs:range ?c . ?p$_1$ rdfs:subPropertyOf ?p$_2$.	?p$_1$ rdfs:range ?c .
scm-hv	**-**	?c$_1$:hasValue ?i ; :onProperty ?p$_1$. ?c$_2$:hasValue ?i ; :onProperty ?p$_2$. ?p$_1$ rdfs:subPropertyOf ?p$_2$.	?c$_1$ rdfs:subClassOf ?c$_2$.
scm-svf1	**-**	?c$_1$:someValuesFrom ?y$_1$; :onProperty ?p . ?c$_2$:someValuesFrom ?y$_2$; :onProperty ?p . ?y$_1$ rdfs:subClassOf ?y$_2$.	?c$_1$ rdfs:subClassOf ?c$_2$.
scm-svf2	**-**	?c$_1$:someValuesFrom ?y ; :onProperty ?p$_1$. ?c$_2$:someValuesFrom ?y ; :onProperty ?p$_2$. ?p$_1$ rdfs:subPropertyOf ?p$_2$.	?c$_1$ rdfs:subClassOf ?c$_2$.
scm-avf1	**-**	?c$_1$:allValuesFrom ?y$_1$; :onProperty ?p . ?c$_2$:allValuesFrom ?y$_2$; :onProperty ?p . ?y$_1$ rdfs:subClassOf ?y$_2$.	?c$_1$ rdfs:subClassOf ?c$_2$.
scm-avf2	**-**	?c$_1$:allValuesFrom ?y ; :onProperty ?p$_1$. ?c$_2$:allValuesFrom ?y ; :onProperty ?p$_2$. ?p$_1$ rdfs:subPropertyOf ?p$_2$.	?c$_1$ rdfs:subClassOf ?c$_2$.
scm-int	**-**	?c :intersectionOf (?c$_1$ ... ?c$_n$) .	?c rdfs:subClassOf ?c$_1$...?c$_n$.
scm-uni	**-**	?c :unionOf (?c$_1$ ... ?c$_n$) .	?c$_1$...?c$_n$ rdfs:subClassOf ?c .

Table 2. Only terminological antecedent patterns

$\mathcal{R}2$ **: one assertional antecedent pattern**				
OWL2RL	**SAOR**	**Antecedent**	**Consequent**	**Notes**
eq-ref	**-**	?s ?p ?o .	?s :sameAs ?s . ?p :sameAs ?p . ?o :sameAs ?o .	
eq-sym	**rdfp6′**	?x :sameAs ?y .	?y :sameAs ?x .	
cls-nothing2	**-**	?x a :Nothing .	**false**	
dt-not-type	**-**	?l a ?dt .	**false**	Where ?l is not in the value space of ?dt

Table 3. No terminological, but one assertional antecedent pattern

$\mathcal{R}3$ **: at least one terminological/only one assertional pattern in antecedent**				
OWL2RL	**SAOR**	**Antecedent**		**Consequent**
		terminological	*assertional*	
prp-dom	**rdfs2**	**?p** rdfs:domain ?c .	?x ?p ?y .	?x a ?c .
prp-rng	**rdfs3′**	**?p** rdfs:range ?c .	?x ?p ?y .	?y a ?c .
prp-irp	**-**	**?p** a :IrreflexiveProperty .	?x ?p ?x .	**false**
prp-symp	**rdfp3′**	**?p** a :SymmetricProperty .	?x ?p ?y .	?y ?p ?x .
prp-spo1	**rdfs7′**	**?p$_1$** rdfs:subPropertyOf ?p$_2$.	?x ?p$_1$?y .	?x ?p$_2$?y .
prp-eqp1	**rdfp13a′**	**?p$_1$** :equivalentProperty ?p$_2$.	?x ?p$_1$?y .	?x ?p$_2$?y .
prp-eqp2	**rdfp13b′**	?p$_1$:equivalentProperty **?p$_2$** .	?x ?p$_2$?y .	?x ?p$_1$?y .
prp-inv1	**rdfp8a′**	**?p$_1$** :inverseOf ?p$_2$.	?x ?p$_1$?y .	?y ?p$_2$?x .
cls-int2	**rdfc3a′**	**?c** :intersectionOf (?c$_1$ ... ?c$_n$) .	?x a ?c .	?x a ?c$_1$...?c$_n$.
cls-uni	**rdfc1′**	?c :unionOf (?c$_1$ ... **?c$_i$** ... ?c$_n$) .	?x a ?c$_i$	?x a ?c .
cls-svf2	**rdfp15′***	?x :someValuesFrom :Thing ; :onProperty **?p** .	?u ?p ?v .	?u a ?x .
cls-hv1	**rdfp14b′**	**?x** :hasValue ?y ; :onProperty ?p .	?u a ?x .	?u ?p ?y .
cls-hv2	**rdfp14a′**	?x :hasValue ?y ; :onProperty **?p** .	?u ?p ?y .	?u a ?x .
cax-sco	**rdfs9**	**?c$_1$** rdfs:subClassOf ?c$_2$.	?x a ?c$_1$.	?x a ?c$_2$.
cax-eqc1	**rdfp12a′**	**?c$_1$** :equivalentClass ?c$_2$.	?x a ?c$_1$.	?x a ?c$_2$.
cax-eqc2	**rdfp12b′**	?c$_1$:equivalentClass **?c$_2$** .	?x a ?c$_2$.	?x a ?c$_1$.

Table 4. At least one terminological and exactly one assertional pattern

$\mathcal{R}4$ **: no terminological pattern/multiple assertional patterns**			
OWL2RL	**SAOR**	**Antecedent**	**Consequent**
eq-trans	**rdfp7**	?x :sameAs ?y . ?y :sameAs ?z .	?x :sameAs ?z .
eq-rep-s	**rdfp11′***	?s :sameAs ?s′. ?s ?p ?o .	?s′?p ?o .
eq-rep-p	**-**	?p :sameAs ?p′. ?s ?p ?o .	?s ?p′?o .
eq-rep-o	**rdfp11′′***	?o :sameAs ?o′. ?s ?p ?o .	?s ?p ?o′.
eq-diff1	**-**	?x :sameAs ?y ; :differentFrom ?y .	**false**
prp-npa1	**-**	?x :sourceIndividual ?i$_1$; :assertionProperty ?p ; :targetIndividual ?i$_2$. ?i$_1$?p ?i$_2$.	**false**
prp-npa2	**-**	?x :sourceIndividual ?i$_1$; :assertionProperty ?p ; :targetValue ?lt . ?i$_1$?p ?lt .	**false**

Table 5. No terminological, but multiple assertional patterns

$\mathcal{R}5$ **: at least one terminological/multiple assertional patterns in antecedent**				
OWL2RL	**SAOR**	**Antecedent**		**Consequent**
		terminological	*assertional*	
prp-fp	**rdfp1′**	**?p** a :FunctionalProperty .	?x ?p ?y_1 , ?y_2 .	?y_1 :sameAs ?y_2 .
prp-ifp	**rdfp2**	**?p** a :InverseFunctionalProperty .	?x_1 ?p ?z . ?x_2 ?p ?z .	?x_1 :sameAs ?x_2 .
prp-asyp	**-**	**?p** a :AsymmetricProperty .	?x ?p ?y . ?y ?p ?x .	**false**
prp-trp	**rdfp4**	**?p** a :TransitiveProperty .	?x ?p ?y . ?y ?p ?z .	?x ?p ?z .
prp-pdw	**-**	**?$\mathbf{p}_1$** :disjointWith ?p_2 .	?x ?p_1 ?y ; ?p_2 ?y .	**false**
prp-adp	**-**	?x a :AllDisjointProperties ; owl:members (**?$\mathbf{p}_1$** ... $\mathbf{p}_n$) .	?u ?p_i ?y ; ?p_j ?y .	**false**
cls-com	**-**	**?$\mathbf{c}_1$** :complementOf **?$\mathbf{c}_2$** .	?x a ?c_1, ?c_2 .	**false**
cls-svf1	**rdfp15′**	?x :someValuesFrom **?y** ; :onProperty **?p** .	?u ?p ?v . ?v a ?y .	?u a ?x .
cls-avf	**rdfp16′**	**?x** :allValuesFrom ?y ; :onProperty **?p** .	?u a ?x ; ?p ?v .	?v a ?y .
cls-maxc1	**-**	**?x** :maxCardinality 0 ; :onProperty **?p** .	?u a ?x ; ?p ?y .	**false**
cls-maxc2	**rdfc4b**	**?x** :maxCardinality 1 ; :onProperty **?p** .	?u a ?x ; ?p ?y_1 , ?y_2 .	?y_1 :sameAs ?y_2 .
cls-maxqc1	**-**	**?x** :maxQualilifedCardinality 0 ; :onProperty **?p** ; :onClass ?**c** .	?u a ?x ; ?p ?y . ?y a ?c .	**false**
cls-maxqc2	**-**	**?x** :maxQualifiedCardinality 0 ; :onProperty **?p** ; :onClass :Thing .	?u a ?x ; ?p ?y .	**false**
cls-maxqc3	**-**	**?x** :maxQualifiedCardinality 1 ; :onProperty **?p** ; :onClass **?c** .	?u a ?x ; ?p ?y_1 , ?y_2 . ?y_1 a ?c . ?y_2 a ?c .	?y_1 :sameAs ?y_2 .
cls-maxqc4	**-**	**?x** :maxQualifiedCardinality 1 ; :onProperty **?p** ; :onClass :Thing .	?u a ?x ; ?p ?y_1 , ?y_2 .	?y_1 :sameAs ?y_2 .
cax-dw	**-**	**?$\mathbf{c}_1$** :disjointWith **?$\mathbf{c}_2$** .	?x a ?c1 , ?c2 .	**false**
cax-adc	**-**	?x a :AllDisjointClasses ; :members (**?$\mathbf{c}_1$** ... **?$\mathbf{c}_n$**) .	?z a ?c_i , ?c_j .	**false**

Table 6. At least one terminological and mulitple assertional patterns

$\mathcal{R}6$ **: no terminological pattern/variable assertional patterns**			
OWL2RL	**SAOR**	**Antecedent**	**Consequent**
eq-diff2	**-**	?x a :AllDifferent ; :members (z_1 ... z_n) . ?z_i :sameAs ?z_j .	**false**
eq-diff3	**-**	?x a :AllDifferent ; :distinctMembers (z_1 ... z_n) . ?z_i :sameAs ?z_j .	**false**

Table 7. No terminological, but a variable number of assertional patterns

$\mathcal{R}7$ **: at least one terminological/variable assertional patterns in antecedent**				
OWL2RL	**SAOR**	**Antecedent**		**Consequent**
		terminological	*assertional*	
prp-spo2	**-**	?p :propertyChainAxiom (**?$\mathbf{p}_1$** ... **?$\mathbf{p}_n$**) .	?u_1 ?p_1 ?u_2 . ... ?u_n ?p_n ?u_{n+1} .	?u_1 ?p ?u_{n+1} .
prp-key	**-**	**?c** :hasKey (**?$\mathbf{p}_1$** ... $\mathbf{p}_n$) .	?x a ?c . ?x ?p_1 ?z_1 . ... ?x ?p_n ?z_n . ?y a ?c . ?y ?p_1 ?z_1 . ... ?y ?p_n ?z_n .	?x :sameAs ?y .
cls-int1	**rdfc3c**	?c :intersectionOf (**?$\mathbf{c}_1$ ... ?$\mathbf{c}_n$**) .	?y a ?c_1 ... ?c_n .	?y a ?c .

Table 8. At least one terminological and variable assertional patterns

Rules not in OWL 2 RL					
ID	**SAOR**	**Antecedent**		**Consequent**	$\mathcal{R}$
		terminological	*assertional*		
cls-minc1	**rdfc2**	**?x** :minCardinality 1 ; :onProperty **?p** .	?u ?p ?y .	?u a ?x .	$\mathcal{R}3$
cls-hs1	-	**?x** :hasSelf true ; :onProperty ?p .	?u ?a ?x .	?u ?p ?u .	$\mathcal{R}3$
cls-hs2	-	?x :hasSelf true ; :onProperty **?p** .	?u ?p ?u .	?u a ?c .	$\mathcal{R}3$
cls-duni1	-	?x :disjointUnionOf (?c_1 ... **?$\mathbf{c}_i$** ... ?c_n) .	?y a ?c_i .	?y a ?x .	$\mathcal{R}3$
cls-duni2	-	?x :disjointUnionOf (... **?$\mathbf{c}_i$** ... **?$\mathbf{c}_j$** ...) .	?y a ?c_i , ?c_j .	**false**	$\mathcal{R}5$

Table 9. Rules not in OWL 2 RL