# Rule-based OWL Modeling
# with ROWLTab Protégé Plugin

Md. Kamruzzaman Sarker[1], Adila Krisnadhi[1,2],
David Carral[3], and Pascal Hitzler[1]

[1] Data Semantics (DaSe) Laboratory, Wright State University, OH, USA
[2] Universitas Indonesia, Depok, Indonesia
[3] Center for Advancing Electronics Dresden (cfaed), TU Dresden, Germany

**Abstract.** It has been argued that it is much easier to convey logical statements using rules rather than OWL (or description logic (DL)) axioms. Based on recent theoretical developments on transformations between rules and DLs, we have developed ROWLTab, a Protégé plugin that allows users to enter OWL axioms by way of rules; the plugin then automatically converts these rules into OWL 2 DL axioms if possible, and prompts the user in case such a conversion is not possible without weakening the semantics of the rule. In this paper, we present ROWLTab, together with a user evaluation of its effectiveness compared to entering axioms using the standard Protégé interface. Our evaluation shows that modeling with ROWLTab is much quicker than the standard interface, while at the same time, also less prone to errors for hard modeling tasks.

## 1   Introduction

About a decade ago, not long after description logics [1] had been chosen as the basis for the then-forthcoming W3C Recommmendation for the Web Ontology Language OWL [6], a rather agressively voiced discussion as to whether a rule-based paradigm might have been a better choice emerged in the Semantic Web community [7,19,23]. On the one hand, this eventually led to a new W3C Recommendation on the Rule Interchange Format RIF [9], based on the rules paradigm, while an alternative approach which layered rules on top of the existing OWL standard, known as SWRL [8], remained a mere W3C member submission. However, SWRL has proven significantly more popular than RIF. To see this, it may suffice to compare the Google Scholar citation numbers for SWRL – over 2500 since 2004 – and RIF – just over 50 since 2009.

At the same time, researchers kept investigating more elaborate ways to bridge between the two paradigms [10,12,13,16,20], and in particular how to convert rules into OWL [2,11,14,15]. These results regarding conversion now make it possible to express axioms first as rules, and only then to convert them into OWL. We have consistently used this approach to model complex OWL axioms throughout the last few years, as rules are, arguably, easier to understand and produce than OWL (or description logic) axioms in whichever syntax.

Consider the sentence: "If a person has a parent who is female, then this parent is a mother", which we consider to be of medium difficulty in terms of modeling it in OWL. As a first-order logic rule, this can be expressed as

$$\text{Person}(x) \wedge \text{hasParent}(x, y) \wedge \text{Female}(y) \rightarrow \text{Mother}(y).$$

This can be expressed in OWL using description logic syntax as follows:

$$\text{Female} \sqcap \exists \text{hasParent}^-.\text{Person} \sqsubseteq \text{Mother}$$

Based on anecdotal evidence, many people find it easier to come up with the rule than directly with the OWL axiom. Following this lead, we have produced a Protégé [18] plugin, called ROWLTab, which accepts rules as input, and adds them as OWL axioms to a given ontology, provided the rule is expressible by an equivalent set of such axioms. In case the rule is not readily convertible, the user is prompted and asked whether the rule shall be saved as SWRL rule.

In order to assess the usefulness of the ROWLTab, we have furthermore conducted a user experiment in which we compare the ROWLTab interface for adding axioms to the standard Protégé interface. Our hypotheses for the user evaluation were that given complex relationships expressed as natural language sentences as above, users will be quicker to add them to an ontology using the ROWLTab than with the standard Protégé interface, and that they will also make less mistakes in doing so. The first hypothesis has been fully confirmed by our experiment, the second has been partially confirmed.

The rest of the paper will be structured as follows. In Section 2 we explain in more detail the rule-to-OWL conversion algorithm used. In Section 3 we present the ROWLTab Protégé plugin. In Section 4 we present our user evaluation and results, and in Section 5 we conclude. More information about the plugin can be found at `http://daselab.org/content/modeling-owl-rules`. A preliminary report on the plugin, without evaluation and with much fewer details, was presented as a software demonstration at the ISWC2016 conference [21].

## 2   SWRL Rules to OWL Axioms Transformation

In this section we introduce theoretical notions employed across the paper. Note that, due to space constraints, some of the definitions below are simplified and may not exactly correspond with existing definitions from different sources.

Let $\mathbf{C}$, $\mathbf{R}$, $\mathbf{I}$ and $\mathbf{V}$ be pairwise disjoint, countably infinite sets of *classes*, *properties*, *individuals* and *variables*, respectively, where $\top, \bot \in \mathbf{C}$, the *universal property* $U \in \mathbf{R}$ (i.e., `owl:topObjectProperty`) and, for every $R \in \mathbf{R}$, $R^- \in \mathbf{R}$ and $R^{--} = R$. A *class expression* is an element of the grammar $\mathbf{E} ::= (\mathbf{E} \sqcap \mathbf{E}) \mid \exists R.\mathbf{E} \mid \exists R.\mathsf{Self} \mid C \mid \{a\}$ where $C \in \mathbf{C}$, $R \in \mathbf{R}$ and $a \in \mathbf{I}$. Furthermore, let $\mathbf{T} = \mathbf{I} \cup \mathbf{V}$ be the set of *terms*. An *atom* is a formula of the form $C(t)$ or $R(t, u)$ where $C \in \mathbf{E}$, $R \in \mathbf{R}$ and $t, u \in \mathbf{T}$. For the remainder of the paper, we identify pairs of atoms of the form $R(t, u)$ and $R^-(u, t)$. Furthermore, we identify a conjunction of formulas with the set containing all the formulas in the conjunction and vice-versa.

An *axiom* is a formula of the form $C \sqsubseteq D$ or $R_1 \circ \ldots \circ R_n \sqsubseteq R$ with $C, D \in \mathbf{E}$ and $R_{(i)} \in \mathbf{R}$. A *rule* is a first-order logic formula of the form $\forall \boldsymbol{x}(\beta(\boldsymbol{x}) \to \eta(\boldsymbol{z}))$ with $\beta$ and $\eta$ are conjunctions of atoms, $\boldsymbol{x}$ and $\boldsymbol{z}$ are non-empty sets of terms where $\boldsymbol{z} \subseteq \boldsymbol{x}$. As customary, we often omit the universal quantifier from rules. Axioms and rules are also referred to as *logical formulas*. Axioms as defined above essentially correspond to OWL 2 EL axioms [4] plus inverse property expression, while rules correspond to SWRL rules minus (in)equality and built-in atoms.

Consider some terms $t$ and $u$ and a conjunction of atoms $\beta$. We say that $t$ *and $u$ are directly connected in* $\beta$ if both terms occur in the same atom in $\beta$. We say $t$ *and $u$ are connected in* $\beta$ if there is some sequence of terms $t_1, \ldots, t_n$ with $t_1 = t$, $t_n = u$, and $t_{i-1}$ and $t_i$ are directly connected in $\beta$ for every $i = 2, \ldots, n$.

The notions of *interpretation* and of an interpretation *entailing an axiom* follow the standard definitions for description logics [5]. For rules $\rho$ of the form $\beta \to \eta$ we say that an interpretation $\mathcal{I}$ *entails* $\rho$ if, for every substitution $\sigma$ we have that $\mathcal{I}, \sigma \models \beta$ implies $\mathcal{I}, \sigma \models \eta$, i.e., the semantics of rules follows the standard semantics of first-order predicate logic. We say that two sets of logical formulas $\mathcal{S}$ and $\mathcal{S}'$ are *equivalent* if and only if every interpretation $\mathcal{I}$ that entails $\mathcal{S}$ also entails $\mathcal{S}'$ and vice-versa. Furthermore, we say that $\mathcal{S}'$ is a *conservative extension* of $\mathcal{S}$ if and only if (i) every interpretation that entails $\mathcal{S}'$ also entails $\mathcal{S}$ and (ii) every interpretation that entails $\mathcal{S}$ and is only defined for the symbols in $\mathcal{S}$ can be extended to an interpretation entailing $\mathcal{S}'$ by adding suitable interpretations for additional signature symbols. It is well-known that a set of logical formulas can be replaced by another set without affecting the outcome of reasoning tasks if the latter set is a conservative extension of the former.

We now formally discuss the transformation of rules into axioms. We do not include a comprehensive description of this transformation, which was introduced in [15], but only present a simplified version in an attempt to make this publication more self-contained. Specifically, our presentation makes the following assumptions about rules, all without loss of generality.

1. Rules do not contain constants. Note that, an atom of the form $R(a, b)$ (resp. $A(a)$) with $a, b \in \mathbf{I}$ in the body of a rule may be replaced by an equivalent atom $\exists U.(\{a\} \sqcap \exists R.\{b\})(x)$ (resp. $\exists U.(\{a\} \sqcap A)(x)$) where $x$ is any arbitrarily chosen variable occurring in the body of the rule. Furthermore, atoms of the form $R(x, a)$ with $x \in \mathbf{V}$ and $a \in \mathbf{I}$ occurring in the body may be replaced by $\exists R.\{a\}(x)$. Similar transformations may be applied to the head of a rule in order to remove all occurrences of constants.
2. The head of a rule is of the form $C(x)$ or $S(x, y)$ with $C \in \mathbf{E}$, $S \in \mathbf{R}$ and $x, y \in \mathbf{V}$. Note that, a rule of the form $\beta \to \eta$ is equivalent to a set of rules $\{\beta \to \eta_1, \ldots, \beta \to \eta_n\}$ provided that $\eta = \eta_1 \cup \ldots \cup \eta_n$.
3. All of the variables in the body of a rule are connected. If two variables $x$ and $y$ are not connected in the body of a rule, we may simply add the atom $U(x, y)$ to the body of the rule resulting in a semantically equivalent rule.

The preprocessing implied by the aforementioned assumptions has been implemented in the ROWLTab plugin and thus, such constraints need not be con-

sidered by the end users. Moreover, such preprocessing is carefully implemented in an attempt to minimize the number of necessary modifications for a rule to satisfy assumptions (1-3). We now proceed with the definition of our translation.

**Definition 1.** *Given some rule $\rho = \beta \to \eta$, let $\delta(\beta \to \eta)$ be the rule that results from exhaustively applying transformations (1-3) where (1) and (2) should be applied with higher priority than (3).*

1. *Replace every atom of the form $R(x,x)$ in $\beta$ or in $\eta$ with $\exists R.\mathsf{Self}(x)$.*
2. *Replace every maximal subset of the form $\{C_1(y)\ldots,C_n(y)\} \subseteq \beta$ with the atom $C_1 \sqcap \ldots \sqcap C_n(y)$.*
3. *For every variable $y$ not occurring in $\eta$ that occurs in exactly one binary atom in $\beta$ of the form $R(z,y)$, do the following:*
   - *If there is some atom of the form $C(y) \in \beta$, then replace the atoms $R(z,y)$ and $C(y)$ in $\beta$ with the atom $\exists R.C(z)$.*
   - *Otherwise, replace the atom $R(z,y)$ in $\beta$ with $\exists R.\top(z)$.*

*Example 1.* Consider the rule $\rho = \mathrm{Person}(x) \wedge \mathrm{hasParent}(x,y) \wedge \mathrm{Female}(y) \to \mathrm{Mother}(y)$. Then, the transformation presented in the previous definition would sequentially produce the following sequence of rules

$$(\exists \mathrm{hasParent}^-.\mathrm{Person})(y) \wedge \mathrm{Female}(y) \to \mathrm{Mother}(y)$$

$$(\exists \mathrm{hasParent}^-.\mathrm{Person} \sqcap \mathrm{Female})(y) \to \mathrm{Mother}(y)$$

Rule $\delta(\rho)$ from the previous example can be directly transformed into an axiom as indicated in the following lemma.

**Lemma 1.** *Consider some rule $\rho$. If $\delta(\rho)$ is of the form $C(x) \to D(x)$, then $\rho$ is equivalent to the axiom $C \sqsubseteq D$.*

*Proof.* Let $\upsilon$ and $\upsilon'$ be some rules such that $\upsilon'$ results by applying some of the transformations (1-3) introduced in Definition 1 to $\upsilon$. Note that, by definition, we can conclude equivalency between $\upsilon$ and $\upsilon'$. Thus, we can show via induction that $\rho$ is equivalent to $\delta(\rho)$. Furthermore, if $\delta(\beta \to \rho)$ is of the form $C(x) \to D(x)$, then, by the definition of the semantics of rules and axioms, $C \sqsubseteq D$ is equivalent to $\delta(\beta \to \rho)$. Since the equivalence relation is transitive, we can conclude that $\rho$ is equivalent to $C \sqsubseteq D$.

As indicated by Lemma 1, rule $\rho$ from Example 1 is equivalent to the axiom $\exists \mathrm{hasParent}^-.\mathrm{Person} \sqcap \mathrm{Female} \sqsubseteq \mathrm{Mother}$.

**Lemma 2.** *Consider some rule $\rho$. If the rule $\delta(\rho)$ is of the form $\bigwedge_{i=2}^n (C_i(x_{i-1}) \wedge R_i(x_{i-1},x_i)) \wedge C_n(x_n) \to S(x_1,x_n)$, then the set of axioms $\{C_i \sqsubseteq \exists R_{C_i}.\mathsf{Self} \mid i = 1,\ldots,n\} \cup \{R_{C_1} \circ R_1 \circ \ldots \circ R_{C_{n-1}} \circ R_n \circ R_{C_n} \sqsubseteq S\}$ where all $R_{C_i}$ are fresh properties unique for every class $C_i$ is a conservative extension of the rule $\rho$.*

*Proof.* As shown in proof of Lemma 1, rules $\rho$ and $\delta(\rho)$ are indeed equivalent. Thus, the lemma follows from the fact that the set of rules presented in the statement of the lemma is a conservative extension of $\delta(\rho)$.
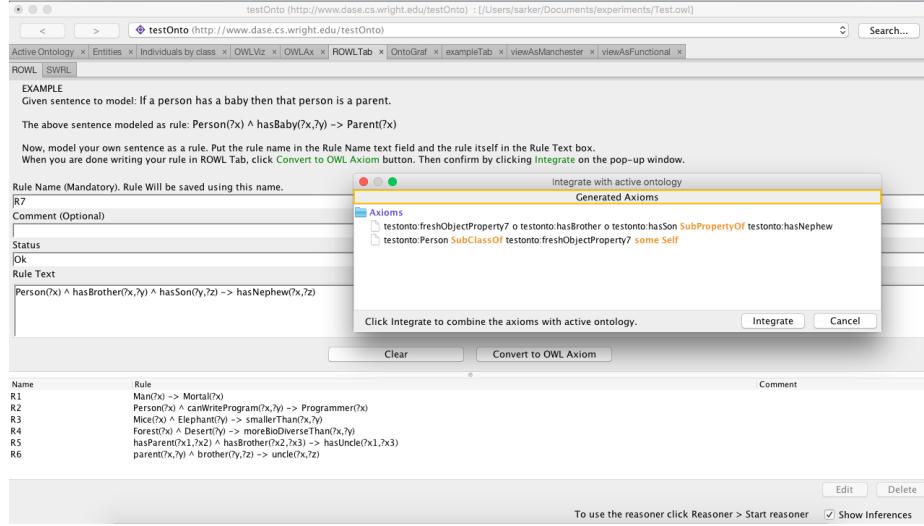
**Fig. 1.** The ROWLTab interface with generated axioms.

We finalize the section with some brief comments about the presented transformation. First, the transformation is sound, but has not been proven to be complete. That is, there may be some rules which are actually expressible as axioms, but cannot be handled by our translation algorithm. Moreover, the transformation of a given rule may in some cases produce axioms that violate some of the global syntactic restrictions of OWL 2 DL, such as regularity restriction on property inclusion/hierarchy, when added to an ontology.

## 3  Plugin Description and Features

Figure 1 shows the user interface of the ROWLTab plugin with generated axioms from a rule and also shows previously saved rules in the bottom part of the user interface. As seen in the figure, the plugin consists of two tabs: ROWL and SWRL. The latter is really the SWRLTab input interface, while the former is our implementation of rule-to-OWL conversion functionality. We have in fact reused the source code of the SWRLTab, kept its functionality intact and added extra functionality by means of the ROWL tab. The upper part of the interface is for rule insertion and the bottom part is for rule modification. At the top a modeling example is also shown.

A user can enter a rule in the "Rule Text" input box using the standard SWRL syntax, e.g.

```
Person(?x) ^ hasChild(?x,?y) ^ Female(?y) -> hasDaughter(?x,?y)
```

Every rule needs a distinct name to be eligible to be saved for later modification. A suggested rule name is automatically generated but the user also has the option

to change the rule name; this can be done in the "Rule Name" input box. The user can also give annotations to a rule in the "Comment" input box. The plugin does syntax checking of the rule, but nothing more sophisticated, e.g. tautologies can also be entered, and checks for global constraints like RBox regularity, which are required for the ontology to stay within OWL 2 DL, are not performed. Since rule-to-OWL conversion often results in the use of property chains, extra care is needed by the modeler to ensure compliance, if compliance is desired; this can also be checked by using a reasoner such as HermiT [17] from within Protégé. When the user is writing a rule it checks whether a predicate is already declared or not. If not declared it will show that the predicate is invalid and the user needs to declare it before the rule can be converted to OWL. Auto completion of predicate names is also supported. The user can use tab-key for auto-completion to existing class, object properties, data properties and individual names.

Figure 1 also shows a button "Convert to OWL Axiom" below the "Rule Text" input box. This button is initially deactivated and if the inserted rule is syntactically correct then this button becomes clickable. When the "Convert to OWL Axiom" button is clicked, ROWLTab will attempt to apply the rule-to-OWL transformation described in the previous section to the given rule. If successful, a pop-up will appear displaying one or more OWL axioms resulting from the transformation, presented in Manchester syntax. These axioms can then be integrated into the active ontology by clicking the "Integrate" button of the pop-up interface. If the given rule cannot be transformed into OWL axioms, ROWLTab will prompt the user if (s)he still want to insert the rule into the ontology as a SWRL rule. If the user agrees, ROWLTab will switch to its SWRL tab and proceed in the same way as adding a rule via the original SWRLTab.

As described in Section 2, the translation of a rule into OWL axioms may sometimes require the introduction of fresh object properties, which will be automatically created by the plugin when necessary. The namespace for these fresh object properties is taken from the default namespace. To create a unique fresh object property, the plugin counts the number of existing fresh object properties in the active ontology (including imports) then increments the counter by 1 and creates the new object property with the incremented counter as part of its identifier.

Once the axioms generated from a rule are added to the ontology, the rule will be saved with the ontology for later modification; the rule is in fact added as an annotation to every OWL axiom generated by the rule. Figure 1 shows saved rules displayed on the bottom left of the ROWLTab plugin. A user can modify or delete rules at any time. If a rule is modified or deleted, the axioms generated by that rule will be affected. That means if a rule is deleted then the axioms generated by the rule will also be deleted. To edit a rule which was previously used to generate axioms the user needs to select the rule first and then click the "Edit" button at the bottom right part of the interface. The rule will then appear in the "Rule Text" input box for modification. The user can also double click on the rule to edit that rule. To delete a rule, the user needs to select that rule and then click the "Delete" button on the bottom right of the interface.

A feature of ROWLTab not found in the SWRLTab is the possibility to automatically add declarations for classes and properties if the inserted rule contains classes or properties not yet defined in the ontology. For example, in the rule above, the original SWRLTab requires that hasChild and hasDaughter be already defined as object properties, and Person and Female as classes in the ontology. This means that the user does not need to first exit the plugin and declare the classes and properties outside the ROWLTab.

Another feature of the ROWLTab plugin is that it actually works as a superset of SWRLTab plugin. So if a user need to work with the SWRLTab plugin, the user does not need to install the SWRLTab plugin separately, as the ROWLTab contains a full instance of the SWRLTab plugin. If the ROWLTab plugin is installed the user only need to switch the tab from ROWL to SWRL to get the full SWRLTab functionality. This also creates a limitation that when a new version of SWRLTab is available the developer of ROWLTab has to embed the newer version of SWRLTab explicitly.

To manage the source code and to be able to modify the source code efficiently in the future we have separated the view module from the control module. The view module consist of the user interface and the control module implements the rule-to-OWL transformation. Besides those two modules we have a separate listener module, which acts as a bridge between the view and controller module. We have used Maven as our build system to easily manage the dependency of various APIs. This plugin is open source and the source code is available at the DaseLab website (`http://dase.cs.wright.edu/content/modeling-owl-rules`).

## 4   Evaluation

For the evaluation of the ROWLTab plugin, we conducted a user evaluation to answer the following three questions:

- Is writing OWL axioms into Protégé via the ROWLTab plugin quicker than writing them directly through the standard Protégé interface?
- Is writing OWL axioms into Protégé via ROWLTab plugin less error-prone than writing them directly through the standard Protégé interface?
- Do users view modeling OWL axioms via ROWLTab plugin to be an easier task than directly through the standard Protégé interface?

The evaluation was conducted by asking the participants to model a set of natural language sentences as rules using the ROWLTab or as OWL axioms using the standard Protégé interface. We recorded time and number of keyboard and mouse clicks required for each question (see Section 4.1) and also recorded the responses which we subsequently assessed for correctness (see Section 4.2). Finally, the participants answered a brief questionnaire (see Section 4.3).

Before describing the experiment in detail, we would like to encourage the readers to do it themselves; it should take less than an hour, the software can be obtained from the ROWLTab website already indicated.

**Table 1.** Evaluation Questions

| Group A | Group B | Difficulty |
|---|---|---|
| 1. Every father is a parent. <br> 2. Every university is an educational institution. | 7. Every parent is a human. <br> 8. Every educational institution is an organization. | easy |
| 3. If a person has a mother then that mother is a parent. <br> 4. Any educational institution that awards a medical degree is a medical school. | 9. If a person has a parent who is female, then this parent is a mother. <br> 10. Any university that is funded by a state government is a public university. | medium |
| 5. If a person's brother has a son, then that son is the first person's nephew. <br> 6. All forests are more biodiverse than any desert. | 11. If a person has a female child, then that person would have that female child as her daughter. <br> 12. All teenagers are younger than all twens. | hard |

For the experiment we recruited 12 volunteers from among the graduate students at Wright State University. Our sole selection criterion was that the participants had at least some basic knowledge of OWL, and had at least minimal exposure to Protégé. All participants were then given a half-hour briefing in which we explained, by means of examples, how to model natural language sentences with and without the ROWLTab in Protégé.

Each participant was given the same twelve natural language sentences to model. The sentences are listed in Table 1 where group A consists of sentence 1 to 6 and group B consists of sentence 7 to 12. As indicated in the table, each group contains two easy, two medium, and two hard sentences to model. Each participant modeled one of the sets of sentences using the ROWLTab and the other group without using the ROWLTab, and we randomly assigned whether the participant will model Group A using the ROWLTab or Group B using the ROWLTab. In order to minimize learning effects which may come from different sentences, we made sure that for each sentence in Group A there is a very similar sentence in Group B, and vice-versa: Each sentence number $n$ in Group A corresponds to sentence number $n+6$ in Group B. We furthermore randomized whether the participant will first model using ROWLTab, and then without the ROWLTab, or vice-versa, also to control for a possible learning effect during the course of the experiment. There was no time limit for the modeling; participants were informed that it should usually take no longer than an hour to model all twelve sentences. Participants were also informed that they cannot go back to earlier sentences during the course of the experiment.

Our categorization into easy, medium, and hard sentences was done as follows: Easy sentences expressed simple subclass relationships. Medium sentences required the use of property restrictions to model them in OWL; the medium sentence 9 was discussed in Section 1. Hard sentences could only be expressed using two or three OWL axioms, together with a technique called *rolification* [11,14].

**Table 2.** Average and standard deviation of time (in seconds), number of clicks (keyboard and mouse), and correctness score per difficulty category of sentences.

| Sentence | Time (in secs) | | # clicks | | Correctness | |
|---|---|---|---|---|---|---|
| Category | Protégé | ROWL | Protégé | ROWL | Protégé | ROWL |
| | avg/std | avg/std | avg/std | avg/std | avg/std | avg/std |
| easy | 79/ 41 | 47/ 9 | 44/ 38 | 59/ 19 | 2.9/0.3 | 2.9/0.3 |
| medium | 312/181 | 116/61 | 216/131 | 141/ 91 | 2.2/0.5 | 2.5/0.8 |
| hard | 346/218 | 160/66 | 351/318 | 228/168 | 0.9/0.7 | 2.5/0.7 |

For example, sentence 5 when expressed as a rule becomes

$$\text{Person}(x) \wedge \text{hasBrother}(x, y) \wedge \text{hasSon}(y, z) \rightarrow \text{hasNephew}(x, z).$$

In order to express this sentence as OWL axioms, one first has to *rolify* the class Person by adding the axiom $\text{Person} \sqsubseteq \exists R_{\text{Person}}.\text{Self}$, where $R_{\text{Person}}$ is a fresh property name, and to then add the property chain axiom

$$R_{\text{Person}} \circ \text{hasBrother} \circ \text{hasSon} \sqsubseteq \text{hasNephew}.$$

We informed all the participants regarding the total number of easy, medium, and hard sentences the participants would face. With each sentence, we also displayed the suitable class and property-names which had been pre-defined by us, i.e. the participants did not have to declare them in Protégé, and directed participants to use the displayed class and property names to the maximum extent possible. For example, the pre-defined classes and properties for sentence 9 were Person, hasParent, Female, Mother, while for sentence 5 they were Person, hasBrother, hasSon, hasNephew. An exception to this is when modeling the hard sentences (5, 6, 11, and 12) via standard Protégé. Those sentences contain class names that need to be rolified, which necessitates one to declare one or more fresh object properties. In this case, we informed the participants that the hard sentences may require them to declare additional object properties without disclosing that this is due to rolification.

### 4.1 Time Used For Modeling

Our hypothesis was that, on medium and hard sentences, participants would be able to model quicker with the ROWLTab than without it. Cumulated data is given in Table 2.

For the statistical analysis, our null hypothesis was that there is no difference between the time taken with ROWLTab versus Protégé. Since each participant had modeled sentences from each difficulty class, we could perform a paired (two-tailed) t-test – note that assuming normal distributions appears to be perfectly reasonable for this data. For the medium sentences the null hypothesis was rejected with $p \approx 0.002 < 0.01$. For the hard questions the null hypothesis was rejected with $p \approx 0.020 < 0.05$. Both results are statistically significant with $p < 0.05$, thus confirming our hypotheses.

Interestingly, if we run the same t-test also on the easy sentences, the same null hypothesis is also rejected with $p \approx 0.019 < 0.05$. We will reflect on this further below.

In order to aid us in interpreting the results, we also recorded the number of clicks (keyboard plus mouse) required for modeling each sentence; cumulative data is provided also in Table 2. If we run the number of clicks through the same t-test as before, the null hypothesis being that there is no difference between the two interfaces. The corresponding $p$-values are 0.092 (easy), 0.030 (medium) and 0.173 (hard), i.e. we have $p < 0.05$ only for the medium sentences. The click analysis may provide us with a partial answer to the better performance of the ROWLTab regarding time used: Fewer clicks may in this case simply translate into less time required. However, this observation does not explain the data for easy and hard questions. We will return to this discussion at the end of Section 4.2, after we have looked at answer correctness.

### 4.2   Correctness of Modeled Axioms

Our hypothesis was that for medium and hard questions, participants would provide more correct answers with the ROWLTab than without it. To see if we can confirm this hypothesis, we verify the correctness of the axioms in the OWL files obtained from the participants. The correct set of axioms for each modeling question is given in Table 3. Since the sentences are short and the resulting OWL axioms are relatively simple, the verification was done manually. Also, for modeling tasks where the participants were asked to model the sentence via ROWLTab, we check the correctness of the rules by examining the OWL axioms obtained after translation, which are annotated with information regarding the actual rule input given to ROWLTab. We then assign a score of 0, 1, 2, or 3 to each answer from the participants as follows:

- Modeling a sentence via ROWLTab, the score is:
  - 3 if the participant's rule is fully correct (equivalent to the answer key),
  - 2 if the participant's mistakes are only in the incorrect use of variables (wrong placement, missing/spurious variables), i.e., the rule still employs the correct predicates in the rule body and head and no spurious predicates are used,
  - 1 if there's a missing predicate in the participant's rule or spurious predicates are used that makes the rule not equivalent to the correct answer,
  - 0 if the participant provides no answer.
- Modeling a sentence via the standard Protégé interface, the score is:
  - 3 if the participant's OWL axioms are fully correct,
  - 2 if the participant's OWL axioms employ the correct set of class and property names, but there is a mistake in the use of logical constructs,
  - 1 if there is a missing or spurious class names or property names, or even missing some necessary OWL axioms,
  - 0 if the participant provides no answer.

**Table 3.** Answers to Evaluation Questions – Ru$i$ and Ax$i$ are answers for question $i$ in the form of rule and OWL axioms, resp. where $R_1, \ldots, R_7$ are fresh (object) properties generated due to rolification

---

Ru1: $\text{Father}(x) \rightarrow \text{Parent}(x)$     Ax1: $\text{Father} \sqsubseteq \text{Parent}$

---

Ru2: $\text{University}(x) \rightarrow \text{EducationalInstitution}(x)$

Ax2: $\text{University} \sqsubseteq \text{EducationalInstitution}$

---

Ru3: $\text{Person}(x) \wedge \text{hasMother}(x, y) \rightarrow \text{Parent}(y)$

Ax3: $\exists \text{hasMother}^-.\text{Person} \sqsubseteq \text{Parent}$

---

Ru4: $\text{EducationalInstitution}(x) \wedge \text{awards}(x, y) \wedge \text{MedicalDegree}(y)$
$$\rightarrow \text{MedicalSchool}(x)$$

Ax4: $\text{EducationalInstitution} \sqcap \exists \text{awards}.\text{MedicalDegree} \sqsubseteq \text{MedicalSchool}$

---

Ru5: $\text{Person}(x) \wedge \text{hasBrother}(x, y) \wedge \text{hasSon}(y, z) \rightarrow \text{hasNephew}(x, z)$

Ax5: $\text{Person} \sqsubseteq \exists R_1.\mathsf{Self}$,    $R_1 \circ \text{hasBrother} \circ \text{hasSon} \sqsubseteq \text{hasNephew}$

---

Ru6: $\text{Forest}(x) \wedge \text{Desert}(y) \rightarrow \text{moreBiodiverseThan}(x, y)$

Ax6: $\text{Forest} \sqsubseteq \exists R_2.\mathsf{Self}$,    $\text{Desert} \sqsubseteq \exists R_3.\mathsf{Self}$,    $R_2 \circ U \circ R_3 \sqsubseteq \text{moreBiodiverseThan}$

---

Ru7: $\text{Parent}(x) \rightarrow \text{Human}(x)$     Ax7: $\text{Parent} \sqsubseteq \text{Human}$

---

Ru8: $\text{EducationalInstitution}(x) \rightarrow \text{Organization}(x)$

Ax8: $\text{EducationalInstitution} \sqsubseteq \text{Organization}$

---

Ru9: $\text{Person}(x) \wedge \text{hasParent}(x, y) \wedge \text{Female}(y) \rightarrow \text{Mother}(x)$

Ax9: $\text{Person} \sqcap \exists \text{hasParent}.\text{Female} \sqsubseteq \text{Mother}$

---

Ru10: $\text{University}(x) \wedge \text{fundedBy}(x, y) \wedge \text{StateGovernment}(y) \rightarrow \text{PublicUniversity}(x)$

Ax10: $\text{University} \sqcap \exists \text{fundedBy}.\text{StateGovernment} \sqsubseteq \text{PublicUniversity}$

---

Ru11: $\text{Person}(x) \wedge \text{hasChild}(x, y) \wedge \text{Female}(y) \rightarrow \text{hasDaughter}(x, y)$

Ax11: $\text{Person} \sqsubseteq \exists R_4.\mathsf{Self}$,    $\text{Female} \sqsubseteq \exists R_5.\mathsf{Self}$,    $R_4 \circ \text{hasChild} \circ R_5 \sqsubseteq \text{hasDaughter}$

---

Ru12: $\text{Teenager}(x) \wedge \text{Twen}(y) \rightarrow \text{youngerThan}(x, y)$

Ax12: $\text{Teenager} \sqsubseteq \exists R_6.\mathsf{Self}$,    $\text{Twen} \sqsubseteq \exists R_7.\mathsf{Self}$,    $R_6 \circ U \circ R_7 \sqsubseteq \text{youngerThan}$

---

The average and standard deviation of the correctness score for easy, medium, and hard questions can be found in Table 2. Here, our null hypothesis was that there is no difference in correctness of the answers given with ROWLTab versus Protégé. For the same reasons as before, we thus performed a paired (two-tailed) t-test, the null hypothesis being that there be no difference whether using ROWLTab or not. For the medium sentences we obtained $p \approx 0.18 > 0.05$, so the null hypothesis could not be rejected. But for the hard questions the null hypothesis was rejected with $p \approx 0.0001 < 0.01$. The latter result is statistically significant with $p < 0.01$. If we run the same t-test also on the easy sentences, the same null hypothesis cannot be rejected; we in fact obtain $p \approx 1.0000$.

We thus confirm our hypothesis that ROWLTab helps users in modeling hard sentences correctly; however we could not confirm this for medium sentences on this population of participants. It could be hypothesized that the participants

were sufficiently familiar with Protégé to perform well on medium difficulty, thus use of the ROWLTab only had an effect on time used, as shown in Section 4.1.

At the same time, the correctness analysis also sheds further light on the hard questions: While participants used less time for these on the ROWLTab, they did not use significantly fewer clicks; however answer correctness was much higher on the ROWLTab. This seems to indicate that the additional time using Protégé was spent thinking (and indeed, rather unsuccessfully) about the problem, while this additional thinking was not required when using the ROWLTab.

### 4.3   Participant Survey

We finally used a questionnaire with four questions to assess the subjective value which the use of the ROWLTab had to the participants. For this, we asked all participants to indicate to what extent they agree with each of the following statements.

1. ROWLTab is a useful tool to help with ontology modeling.
2. Modeling rules with ROWLTab was easier for me than modeling without it.
3. Given some practice, I think I will find modeling rules with the ROWLTab easier than modeling without it.
4. The ROWLTab is better for ontology modeling than the SWRLTab.

Participants were asked to click, on screen whether they agree with each statement, on a scale from -3 (strongly disagree) to +3 (strongly agree). It turns out that participants agreed highly with all three statements:

| Question Number | mean | standard deviation |
|---|---|---|
| 1 (ROWL is a useful tool.) | 2.83 | 0.39 |
| 2 (ROWL makes modeling easier.) | 3.00 | 0.00 |
| 3 (Modeling with ROWL easier with some practice.) | 2.75 | 0.45 |
| 4 (ROWLTab better than SWRLTab) | 1.75 | 1.22 |

In assessing these responses, we need to be aware that the pool of participants came from the investigators' institution, and many of them were either associated with the investigators' lab or had attended classes by one of the investigators. Hence the scores should be interpreted with caution. Nevertheless, the scores for the first three questions indicate strong agreement with the usefulness of the ROWLTab.

Regarding the fourth question, it should be noted that our briefing did not include a briefing on the SWRLTab. As discussed in Section 3, the user interaction of the ROWLTab is very similar to that of the SWRLTab, so the only substantial difference would be in the fact that the ROWLTab produces OWL axioms, while the SWRLTab produces SWRL axioms with a different semantics. We do not know to what extent the participants were aware of this difference. A quarter of the participants answered this question with "0" (neutral). Results of the experiment can be found at `http://dase.cs.wright.edu/content/rowl` and raw result can be found at `https://github.com/md-k-sarker/ROWLPluginEvaluation/tree/master/results` .

**Table 4.** Summary of evaluation results. Entries indicate whether the difference between using the ROWLTab and not using it were statistically significant.

| category | time | clicks | correctness |
|---|---|---|---|
| easy | significant ($p < 0.05$) | not significant | not significant |
| medium | significant ($p < 0.01$) | significant ($p < 0.05$) | not significant |
| hard | significant ($p < 0.05$) | not significant | significant ($p < 0.01$) |

## 5    Conclusions and Further Work

We have presented the Protégé ROWLTab plugin for rule-based OWL modeling in Protégé, and its underlying algorithms. We have furthermore reported on a user evaluation for assessing the improvements arising from the use of ROWLTab.

The evaluation results are summarized in Table 4: We have a significant time improvement in all three categories (it was hypothesized by us only for medium and hard sentences). In the medium category, where answer correctness was not significantly different, the ROWLTab required significantly less clicks. In the hard category, the difference in answer correctness was also significant.

The evaluation results are rather encouraging, and we also already received direct feedback from users that the ROWLTab is considered very useful. But while basic functionality is already in place, we already see further improvements that can be made to the plugin:

– When rolification is used for the transformation of a rule to OWL, the ROWLTab currently invents an artificial property name for the fresh object property. It may be helpful to more directly support a renaming of these properties, or to come up with a standard naming scheme for properties arising out of rolification. Note, however, that it is not sufficient to have one fresh property for each defined atomic class, as in some cases complex classes need to be rolified [11].

– The translation of rules into OWL often leads to the use of property chains, which may result in a non-regular property hierarchy, thus violating a global syntactic restriction of OWL 2 DL. While standard tools such as reasoners, which can be called from within Protégé, can detect this issue, it may be helpful to catch this earlier, e.g. directly at the time when a rule is translated.

– Currently, if a rule is input which cannot be translated to OWL, it is simply saved as a SWRL rule, i.e., with a significantly modified (and, in a sense, restricted) semantics. However, through the use of so-called *nominal schemas* [14] it is possible to recover more of the first-order semantics of the input rules, and it has even been shown that the use of such nominal schemas can lead to performance improvements of reasoners compared to SWRL [22].

More substantial possible future work would carry the ROWLTab theme beyond the basic rule paradigm currently supported:

– The rule syntax could be extended to allow for capturing OWL features which cannot be expressed by means of the basic rules currently supported. In particular, these would be right-hand side (head) disjunctions and existentials as well as cardinality restrictions, as well as left-hand side universal quantifiers. It would even be conceivable to add additonal shortcut notation, e.g. for witnessed universals [3], or for nominal schemas [14].

– The development of a full-blown rule syntax for all of OWL 2 DL would then also make it possible to perform all ontology modeling using rules, i.e., to establish an interface where the user would get a pure rules view on the ontology, if desired.

We are looking forward to feedback by ontology modelers on the route which we should take with the plugin in the future.

# References

1. Baader, F., et al. (eds.): The Description Logic Handbook: Theory, Implementation, and Applications. Cambridge University Press, 2nd edn. (2010)
2. Carral, D., Hitzler, P.: Extending description logic rules. In: Simperl, E., Cimiano, P., Polleres, A., Corcho, Ó., Presutti, V. (eds.) The Semantic Web: Research and Applications - 9th Extended Semantic Web Conference, ESWC 2012, Heraklion, Crete, Greece, May 27-31, 2012. Proceedings. Lecture Notes in Computer Science, vol. 7295, pp. 345–359. Springer (2012), http://dx.doi.org/10.1007/978-3-642-30284-8_30
3. Carral, D., Krisnadhi, A., Rudolph, S., Hitzler, P.: All but not nothing: Left-hand side universals for tractable OWL profiles. In: Keet, C.M., Tamma, V.A.M. (eds.) Proceedings of the 11th International Workshop on OWL: Experiences and Directions (OWLED 2014), Riva del Garda, Italy, October 17-18, 2014. CEUR Workshop Proceedings, vol. 1265, pp. 97–108. CEUR-WS.org (2014)
4. Cuenca Grau, B., Motik, B., Wu, Z., Fokoue, A., Lutz, C.: OWL 2 Web Ontology Language Profiles (Second Edition). W3C Recommendation (11 December 2012), http://www.w3.org/TR/owl2-profiles/
5. Hitzler, P., Krötzsch, M., Rudolph, S.: Foundations of Semantic Web Technologies. CRC Press/Chapman & Hall (2010)
6. Hitzler, P., et al. (eds.): OWL 2 Web Ontology Language Primer (Second Edition). W3C Recommendation (11 December 2012), http://www.w3.org/TR/owl2-primer/
7. Horrocks, I., Parsia, B., Patel-Schneider, P.F., Hendler, J.A.: Semantic web architecture: Stack or two towers? In: Fages, F., Soliman, S. (eds.) Principles and Practice of Semantic Web Reasoning, Third International Workshop, PPSWR 2005, Dagstuhl Castle, Germany, September 11-16, 2005, Proceedings. Lecture Notes in Computer Science, vol. 3703, pp. 37–41. Springer (2005)
8. Horrocks, I., et al.: SWRL: A Semantic Web Rule Language Combining OWL and RuleML. W3C Member Submission (21 May 2004), http://www.w3.org/Submission/SWRL/
9. Kifer, M., Boley, H. (eds.): RIF Overview (Second Edition). W3C Working Group Note (5 February 2013), https://www.w3.org/TR/rif-overview/
10. Knorr, M., Hitzler, P., Maier, F.: Reconciling OWL and non-monotonic rules for the semantic web. In: Raedt, L.D., et al. (eds.) ECAI 2012 – 20th European Conference on Artificial Intelligence, Montpellier, France, August 27-31 , 2012. pp. 474–479. IOS Press (2012)

11. Krisnadhi, A., Maier, F., Hitzler, P.: OWL and rules. In: Polleres, A., et al. (eds.) Reasoning Web. Semantic Technologies for the Web of Data - 7th International Summer School 2011, Galway, Ireland, August 23-27, 2011, Tutorial Lectures. Lecture Notes in Computer Science, vol. 6848, pp. 382–415. Springer (2011)
12. Krötzsch, M.: Description Logic Rules, Studies on the Semantic Web, vol. 8. IOS Press (2010)
13. Krötzsch, M., Hitzler, P., Vrandecic, D., Sintek, M.: How to reason with OWL in a logic programming system. In: Eiter, T., et al. (eds.) Proceedings of the Second International Conference on Rules and Rule Markup Languages for the Semantic Web, RuleML2006. pp. 17–26. IEEE Computer Society, Athens, Georgia (2006)
14. Krötzsch, M., Maier, F., Krisnadhi, A., Hitzler, P.: A better uncle for OWL: nominal schemas for integrating rules and ontologies. In: Srinivasan, S., et al. (eds.) Proceedings of the 20th International Conference on World Wide Web, WWW 2011, Hyderabad, India, March 28 - April 1, 2011. pp. 645–654. ACM (2011)
15. Krötzsch, M., Rudolph, S., Hitzler, P.: Description Logic Rules. In: Ghallab, M., et al. (eds.) Proceeding of the 18th European Conference on Artificial Intelligence, Patras, Greece, July 21-25, 2008. vol. 178, pp. 80–84. IOS Press, Amsterdam, The Netherlands (2008)
16. Krötzsch, M., Rudolph, S., Hitzler, P.: ELP: Tractable Rules for OWL 2. In: Sheth, A.P., et al. (eds.) Proceedings of the 7th International Semantic Web Conference, ISWC 2008, Karlsruhe, Germany, October 26-30, 2008. Lecture Notes in Computer Science, vol. 5318, pp. 649–664. Springer (2008)
17. Motik, B., Shearer, R., Horrocks, I.: Hypertableau Reasoning for Description Logics. Journal of Artificial Intelligence Research 36, 165–228 (2009)
18. Musen, M.A.: The Protégé project: A look back and a look forward. AI Matters 1(4), 4–12 (2015)
19. Patel-Schneider, P.F., Horrocks, I.: A comparison of two modelling paradigms in the semantic web. In: Proc. of the Fifteenth International World Wide Web Conference (WWW 2006). pp. 3–12. ACM (2006)
20. Rudolph, S., et al.: Efficient OWL Reasoning with Logic Programs – Evaluations. In: Marchiori, M., et al. (eds.) Proceedings of The First International Conference on Web Reasoning and Rule Systems 2007 (RR2007), Innsbruck, Austria, 7-8 June 2007. Lecture Notes in Computer Science, vol. 4524, pp. 370–373. Springer (2007)
21. Sarker, M.K., Carral, D., Krisnadhi, A.A., Hitzler, P.: Modeling OWL with rules: The ROWL protege plugin. In: Kawamura, T., Paulheim, H. (eds.) Proceedings of the ISWC 2016 Posters & Demonstrations Track, Kobe, Japan, October 19, 2016. CEUR Workshop Proceedings, vol. 1690. CEUR-WS.org (2016)
22. Steigmiller, A., Glimm, B., Liebig, T.: Reasoning with nominal schemas through absorption. Journal of Automated Reasoning 53(4), 351–405 (2014)
23. W3C Workshop on Rule Languages for Interoperability, 27-28 April 2005, Washington, DC, USA. W3C (2005), https://www.w3.org/2004/12/rules-ws/accepted