

RuleML2TPTP Project

Part 2

Meng Luan

m.luan AT unb.ca

University of New Brunswick

November 17, 2014

Introduction

<http://edmonl.github.io/RuleML2TPTP>

- Quick start
- Project releases
- Datalog⁺ RuleML examples
- Datalog⁺ schema
- Project wiki (FAQ)
- ...

XSLT Translation - Preprocessing

- XSLT-based normalizer
 - ▶ Produces a fully striped form
 - ▶ Performs canonical ordering

XSLT Translation

- Core Task: XSLT translator development
- Input: normalized Datalog⁺ RuleML (XML)
- Output: TPTP language (plain text)
- Supports: `<Assert>`, `<Query>`, `<Or>`, ...
- Ignores: `<Retract>`, `<Data>`

XSLT Translation

- Use push style, in contrast to pull style
 - ▶ Push style: apply templates recursively
 - ▶ Pull style: select nodes iteratively

- Trivial nodes

```
<xsl:template match="r:*">  
  <xsl:apply-templates/>  
</xsl:template>
```

- Pretty printing

XSLT Translation

- Generate formula name
 - ▶ `assert_act<N>_formula<M>`
 - ▶ `query_act<N>_formula<M>`
- Map formula role
 - ▶ `<Assert> --> axiom`
 - ▶ `<Query> --> conjecture`

```
top <formula> -->
  fof(<name>, <role>, (
    ...
  ))
```

XSLT Translation

```
<Forall> --> ![<declare-list>]:...  
<Exists> --> ?[<declare-list>]:...  
<Implies> --> (<if>) => (<then>)  
<Equal> --> <left> = <right>  
<Or/> --> $false  
<And/> --> $true
```

Variables, Constants and Functors

- Variables
 - ▶ Uppercase the first letter
- Constants and functors
 - ▶ Lowercase the first letter
 - ▶ Single-quote the whole string
- Warnings

Build and Run

```
$ ls -R
$ ant -projecthelp
$ ant jar
$ java -jar ruleml2tptp -h
$ java -jar ruleml2tptp \
  [-s <file>] [-o <file>]
```

Java Coding

- Call Saxon directly?
 - Too much memorizing and typing...
- Shell scripts?
 - Platform- (shell-) dependent
 - Maintenance of multiple sets of scripts
- Java!

Comments

- Retained by default
- Converted into proper commenting syntax
- Do not want comments in the output?

```
$java -jar ruleml2tptp -g v \  
    [-s <file>] [-o <file>]
```

Comments

- In XML:

```
<!--
```

```
        An example comment:
```

```
%      Indent
```

```
-->
```

- In the TPTP language:

```
% An example comment:
```

```
%      Indent
```

Miscellaneous Options

- -r: use CRLF instead of LF (default)
- -nt, -nn: only normalize or translate
- -t: specify factory class
- -c: select comments by regular expression
- -g: flags for selecting comments

Summary

<http://edmon1.github.io/RuleML2TPTP>

- Finished on schedule
- An open project
- Leave your comments on GitHub