RuleML 1.0:
The Overarching Specification of Web Rules

Harold Boley¹, Adrian Paschke², and Omair Shafiq³

¹ Institute for Information Technology, National Research Council Canada
Fredericton, NB, Canada
harold.boley@nrc.gc.ca

² Freie Universitaet Berlin, Germany
paschke@mi.fu-berlin.de

³ University of Calgary, AB, Canada
moshafiq@ucalgary.ca

Abstract. RuleML is a family of languages, whose modular system of XML schemas permits high-precision Web rule interchange. The family’s top-level distinction is deliberation rules vs. reaction rules. Deliberation rules include modal and derivation rules, which themselves include facts, queries (incl. integrity constraints), and Horn rules (incl. Datalog). Reaction rules include Complex Event Processing (CEP), Knowledge Representation (KR), and Event-Condition-Action (ECA) rules, as well as Production (CA) rules. RuleML rules can combine all parts of both derivation and reaction rules. This allows uniform XML serialization across all kinds of rules. After its use in SWRL and SWSL, RuleML has provided strong input to W3C RIF on several levels. This includes the use of ‘striped’ XML as well as the structuring of rule classes into sublanguages with partial mappings between, e.g., Datalog RuleML and RIF-Core, Hornlog RuleML and RIF-BLD, as well as Production RuleML and RIF-PRD. We discuss the rationale and key features of RuleML 1.0 as the overarching specification of Web rules that encompasses RIF RuleML as a subfamily, and takes into account corresponding OASIS, OMG (e.g., PRR, SBVR), and ISO (e.g., Common Logic) specifications.

1 Introduction

Rules on the Web come in various formats and with diverse packaging. Often, however, the semantics of Web-distributed rule content are compatible. In such cases, rulebases can be reused with an interchange technology consisting of a family of canonical rule languages and bi-directional translators between canonical languages and the languages to be interchanged. The need for Web rule interchange has been increasing with the amount of business rules (incl. policies, regulations, laws, ...) in many domains (e.g. finance, engineering, healthcare, ...) on the Web 1.0, 2.0 (Social), and 3.0 (Social Semantic).

RuleML has been designed for the interchange of the major kinds of Web rules in an XML format that is uniform across various rule languages and platforms. It has broad coverage and is defined as an extensible family of languages, whose
modular system of XML schemas permits rule interchange with high precision, as follows.

When a rulebase is prepared for interchange by a sender,

- it is translated to RuleML if the source document is not in the RuleML format already,
- the Most Specific Schema (MSS) is determined against which the RuleML document can be validated,
- the Internationalized Resource Identifier (IRI) of the MSS is pointed to from the rulebase or is otherwise transmitted along with the rulebase.

When a rulebase is obtained by a receiver,

- it is validated against the same RuleML schema to exclude any too specific MSS assignments and transmission errors,
- it is converted to the local format if the target document is not to be in RuleML anyway.

The RuleML family constitutes a taxonomy of subfamilies, languages, and sub-languages classified through the syntactic power of rules, as reflected by their XML Schema Definitions (XSDs), and through their semantic power, as reflected by their model-theoretic, proof-theoretic, and operational semantics. Often, more syntactic power leads to more semantic power (e.g., the introduction of Expression syntax pushes Datalog to Horn Logic (Hornlog) models in Section 3.2). Syntactically neutral aspects of semantic power will be expressed by semantic attributes (e.g., by a negation attribute for the semantics of Negation-as-failure in Section 3.3).

Fig. 1, a simplified version of the RuleML taxonomy, shows the semantic subfamilies of Deliberation rules for inference and Reaction rules for (re)action. Deliberation rules, via Higher Order Logic (HOL) and First Order Logic (FOL), subsume Derivation rules. Derivation rules subsume Hornlog and Datalog languages and (syntactically) specialize to the condition-less Fact and conclusion-less Query languages (subsuming Integrity Constraint (IC) languages). Reaction rules subsume Complex Event Processing (CEP) and Knowledge Representation (KR) rules, as well as Event-Condition-Action-Postcondition (ECAP) rules. ECAP rules specialize to Event-Condition-Action (ECA) rules, which themselves specialize to Condition-less Trigger (EA) rules and to the rule subfamily of Event-less Production (CA) rules. The RuleML family also has ‘mix-ins’ for Equality and (oriented) Rewriting, as well as for Naf. The Reaction subfamily has mix-ins for Event Algebra, Action Algebra, etc.

While not shown in Fig. 1, RuleML languages make use of ‘pluggable’ libraries of built-ins such as from the Semantic Web Rule Language (SWRL) [HPSB+04] and the Rule Interchange Format (RIF) [PBK10]. There are also entire RuleML languages we cannot further discuss in the confines of this paper, including for uncertainty and fuzzy rules [DPSS08] and for defeasible rules [KBA08].

1 Fuzzy RuleML: http://www.image.ntua.gr/FuzzyRuleML
2 Defeasible RuleML: http://defeasible.org

---

1 Fuzzy RuleML: http://www.image.ntua.gr/FuzzyRuleML
2 Defeasible RuleML: http://defeasible.org