To appear: Proceedings of the IJCAI-01 Workshop on E-business and the Intelligent Web, edited by Alun Preece. To be held August 5, 2001, in Seattle, Washington, USA, in conjunction with the International Joint Conference on Artificial Intelligence (IJCAI-01) (http://www.ijcai.org).

Standardizing XML Rules: Preliminary Outline of Invited Talk

Benjamin N. Grosof
MIT Sloan School of Management
Room E53-317, 50 Memorial Drive, Cambridge, MA 02142, USA
`bgrosof@mit.edu` (alt.: `grosof@cs.stanford.edu`)
`http://www.mit.edu/~bgrosof/`

Abstract

I give an overview of current efforts to standardize rules knowledge representation in XML. I focus especially on the design approach and criteria of RuleML, an emerging standard that I am helping to lead.

1 Overview of RuleML

I am leading, with Harold Boley of DFKI (Germany) and Said Tabet of Nisus Inc. (USA), an early-phase standards effort on a markup language for exchange of rules in XML, called RuleML (Rule Markup Language)[1]. The goal of this effort is eventual adoption as a Web standard, e.g., via the World Wide Web Consortium (W3C[2] within its new Semantic Web Activity. Along the way there are some interesting new research issues.

RuleML is, at its heart, an XML syntax for rule knowledge representation (KR), that is inter-operable among major commercial rule systems. It is especially oriented towards four commercially important families of rule systems: SQL (relational database), Prolog, production rules (cf. OPS5, CLIPS, Jess) and Event-Condition-Action rules (ECA). These kinds of rules today are especially found embedded in Object-Oriented (OO) systems, and are often used for business process connectors / workflow. These four families of rule systems all have common core abstraction: declarative logic programs (LP). “Declarative” here means in the sense of KR theory. Note that this supports both backward inferencing and forward inferencing.

RuleML is actually a family (lattice) of rule KR expressive classes: each with a DTD (syntax) and an associated KR semantics (KRsem). These expressive classes form a generalization hierarchy (lattice). The KRsem specifies what set of conclusions are sanctioned for any given set of premises. Being able to define an XML syntax is relatively straightforward. Crucial is the semantics (KRsem) and the choice of expressive features.

The motivation to have syntax for several different expressive classes, rather than for one most general expressive class, is that: precision facilitates and maximizes effective interoperability, given heterogeneity of the rule systems/applications that are exchanging rules.

The kernel representation in RuleML is: Horn declarative logic programs. Extensions to this representation are defined for several additional expressive features:

- negation: negation-as-failure and classical negation;
- prioritized conflict handling: e.g., cf. *courteous* logic programs [Grosof *et al.*, 1999];
- disciplined procedural attachments for queries and actions: e.g., cf. *situated* logic programs [Grosof, 1997];
- equivalences, equations, and rewriting;

and other features as well. In addition, RuleML defines some useful expressive restrictions (e.g., Datalog, facts-only, binary-relations-only), not only expressive generalizations.

In January 2001, we released a first public version of a family of DTD’s for several flavors of rules in RuleML. This was presented at the W3C’s Technical Plenary Meeting[3] held Feb. 26 to Mar. 2, 2001. Especially since then, RuleML has attracted a considerable degree of interest in the R&D community. Meanwhile, the design has been evolving to further versions.

[1] 1) http://www.dfki.de/ruleml ; or 2) http://www.mit.edu/~bgrosof and search for “XML Rules” or “RuleML”

[2] http://www.w3.org

[3] a large convocation of most of its face-to-face standards working group meetings

RuleML largely grows out of the design approach and design criteria of Business Rules Markup Language (BRML) which was developed in my previous work at IBM Research and which is implemented in IBM CommonRules[4] available under free trial license from IBM alphaWorks. The design approach and design criteria of CommonRules and BRML are described in [Grosof *et al.*, 1999] [Grosof and Labrou, 2000], and in the documentation in the CommonRules download package. BRML's expressive class is situated courteous logic programs, i.e., declarative logic programs with negation-as-failure, (limited) classical negation, prioritized conflict handling, and disciplined procedural attachments for queries and actions.

RuleML differs in several significant respects from its BRML predecessor, however. One respect is that it defines a family of DTD's. More deeply, however, these differences largely revolve around "Webizing" the KR:

- URI's[5] for logical vocabulary and knowledge subsets
- labels for rules/rulebases, import/export
- headers: meta-data describes the XML document's expressive class
- procedural attachments using Web protocols/services; queries or actions via CGI/servlets/SOAP/...

Such Webizing, and interoperability of KR on the Web, involve several kinds of practical mechanics beyond the representation proper. These include to:

- build on existing W3C standards: namespaces, . . .
- share mechanisms with other emerging/extant standards for KR and ontologies on the Web, including especially RDF/RDFS[6], DAML+OIL[7]; and perhaps also ISO Topic Maps and IEEE Upper Ontologies
- use ontologies for rules, and rules for ontologies
- support ontology tags in: rulebase, predicate symbol, . . .

Further directions include to:

- support definition of syntax in terms of XML Schema as well as in terms of DTD's
- support additional XML syntaxes: RDF; surface/"stylesheeted"
- express more KR's: KIF[8]/classical-logic, Bayesian, fuzzy logic, rewriting systems, temporal, . . .
- provide a Rule mechanism to (other) emerging W3C[9] standards: Semantic Web / RDF, P3P (privacy), . . .

[4] http://www.research.ibm.com/rules and http://alphaworks.ibm.com

[5] Uniform Resource Identifiers, a generalization of Uniform Resource Locators (URL's), a W3C standard

[6] http://www.w3.org, search for Resource Description Format (RDF) and RDF Schema

[7] http://www.daml.org

[8] ANSI Knowledge Interchange Format: http://logic.stanford/kif/ and http://www.cs.umbc.edu/kif/

[9] http://www.w3.org

Acknowledgements

Harold Boley of DFKI (Germany) and Said Tabet of Nisus Inc. (USA) are my close collaborators in leading the RuleML Initiative. Hoi Chan of IBM T.J. Watson Research Center contributed much to the design details of Business Rules Markup Language.

References

[Grosof and Labrou, 2000] Benjamin N. Grosof and Yannis Labrou. An Approach to using XML and a Rule-based Content Language with an Agent Communication Language. In Frank Dignum and Mark Greaves, editors, *Issues in Agent Communication*. Springer-Verlag, 2000. Slightly revised version of paper that appeared in Proceedings of the IJCAI-99 (http://www.ijcai.org) Workshop on Agent Communication Languages (ACL-99) (http://wwwis.win.tue.nl/acl99), held in Stockholm, Sweden, Aug. 1, 1999. Also available as IBM Research Report RC 21491 (May 28, 1999).

[Grosof *et al.*, 1999] Benjamin N. Grosof, Yannis Labrou, and Hoi Y. Chan. A Declarative Approach to Business Rules in Contracts: Courteous Logic Programs in XML. In Michael P. Wellman, editor, *Proceedings of the 1st ACM Conference on Electronic Commerce (EC-99)*. ACM Press, 1999. Held in Denver, CO.

[Grosof, 1997] Benjamin N. Grosof. Building Commercial Agents: An IBM Research Perspective (Invited Talk). In *Proceedings of the Second International Conference and Exhibition on Practical Applications of Intelligent Agents and Multi-Agent Technology (PAAM97)*, P.O. Box 137, Blackpool, Lancashire, FY2 9UN, UK. http://www.demon.co.uk./ar/PAAM97, April 1997. Practical Application Company Ltd. Held London, UK. Also available as IBM Research Report RC 20835 at World Wide Web http://www.research.ibm.com .