Case Study in Rule Inferencing using Protégé and Jess

Eric Wang, Sung Ah Kim, and Yong Se Kim

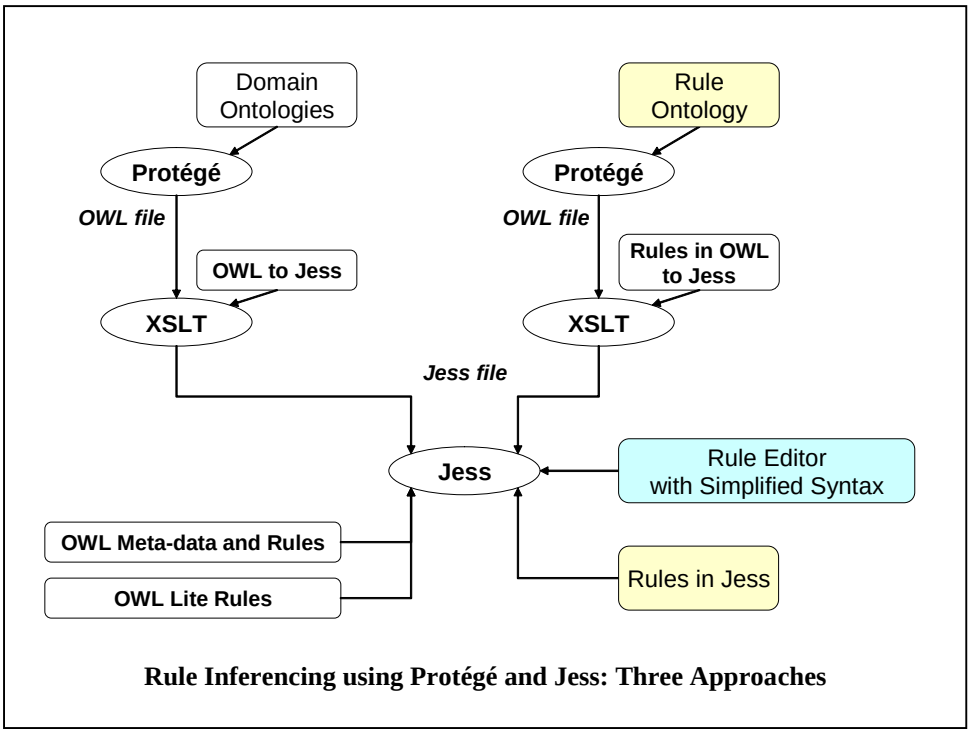
*Cre*ative *D*esign *I*ntelligent *T*utoring *S*ystems (*CreDits*) Research Center

Sungkyunkwan University

Abstract

We report on our research toward the development of an integrated environment for ontology editing and general rule inferencing, using Protégé and Jess. We use Protégé to capture conceptual and relational domain knowledge as ontologies, and Jess to provide "unrestricted" rule inferencing, allowing any expressiveness and any side-effects. This ontology environment is currently being used to support multiple research projects, including an intelligent tutoring system, a home service robot, and a design knowledge repository.

We have partially integrated rule knowledge representation with Protégé by exploring three approaches: (a) direct implementation of rules in Jess, (b) encoding of rule structure



Rule Inferencing using Protégé and Jess: Three Approaches

as a separate Protégé ontology, mimicking SWRL, with rules as instances; and (c) a simplified syntax, with automatic conversion to Jess. We integrate Protégé to Jess via an external conversion step using XSLT and Protégé's OWL output file.

Some significant findings we will discuss:

- *Unrestricted rules.* Our rules are not limited to Description Logic. Also, they must support arbitrary tests (e.g. numeric inequalities) and side-effects (e.g. communication with other applications). Hence, existing classifiers such as Racer, and existing Protégé plugins that support queries and inferencing, were judged to be insufficient for our purposes. This motivated us to adopt Jess as the rule inferencing engine. Subsequently, it introduced numerous challenges in representing our rules in any simpler format within Protégé itself.

- *Ruleflow idioms.* Our rules make use of idioms such as "unanimous vote" that, while simple to express in natural language, are challenging (and verbose) to implement in rule languages. Furthermore, their semantics imposes constraints in the sequence in which (sets of) rules must be evaluated, i.e. "ruleflow". This appears to be an intrinsic consideration in any rule system of sufficient complexity, which transcends rule languages and engines. To our knowledge, there is no consensus on a single mechanism to handle such idioms, even among the rule system communities. This poses a challenge for SWRL and similar efforts to capture rule knowledge, and suggests a need for a parallel specification of large-scale system behaviors such as ruleflow or conflict resolution.

- *Survey of existing rule editing environments.* We will briefly survey existing commercial business rule engines, including their rule expressiveness, conflict resolution and other large-scale control mechanisms, ease of use, and the degree to which they mimic an ontology editor such as Protégé in the course of rule definition.