

For More Information:



Brian Ulicny
VISTology, Inc.
bulicny@vistology.com
508 788 5088

For Immediate Release

VISTology BaseVISor 2.0 Plugin for TopBraid Composer 3.4

Fast, Customizable, Forward-Chaining Inference Engine supports OWL 2 RL

San Francisco, CA – June 8, 2011 – VISTology, Inc., a leading semantic technology company, announced the availability of BaseVISor 2.0 as a plugin for TopQuadrant's TopBraid Composer 3.4 today at the 2011 Semantic Technology Conference. BaseVISor is a powerful semantic inference technology that enables organizations to implement an ontology-driven approach to knowledge representation and reasoning. BaseVISor 2.0 implements the W3C's OWL 2 RL standard.

BaseVISor 2.0 features a highly efficient forward-chaining inference engine, optimized for ontological and rule-based reasoning. Rules and facts are compiled into a Rete network and processed with respect to specified ontologies in order to generate all new inferable facts. In comparison with other leading inference engines, BaseVISor 2.0 has been demonstrated to be considerably faster in inference time and query time on standard benchmarking datasets.

TopQuadrant's TopBraid Composer is a professional development environment for W3C's Semantic Web standards RDF Schema, the OWL Web Ontology Language, the SPARQL Query Language and the Semantic Web Rule Language (SWRL). Composer also can be used as a run-time environment to execute rules, queries, reasoners and mash-ups.

BaseVISor 2.0 Plugin for TopBraid Composer Features:

Rule Editing: The plugin's BaseVISor editor provides rule templates and syntax checking for BaseVISor's rule language. Drag and drop functionality allows the developer to incorporate vocabulary from ontologies within TopBraid Composer. BaseVISor natively supports XML schema datatypes (XSD).

OWL 2 RL Processing: The axioms of OWL 2 RL provide scalable ontology-based reasoning without sacrificing expressive power.

Reasoning: Rules are compiled by BaseVISor and executed at runtime. The BaseVISor inference engine may be chained with other inference engines within the TopBraid Composer environment.

User-Defined Functions: User-defined procedural attachments can be invoked from within a rule to perform a complex calculation complementing BaseVISor's built-in functions. The plugin provides a simple way to deploy custom BaseVISor procedures as Java jar files via a configuration file.

Rule Salience: Enables the rule with the highest priority to be executed first. Used when more than one rule is activated at the same time.

Inclusion Mechanism: Provides a way for including rules from the local file system or on the Web into BaseVISor for reasoning.

A BaseVISor plugin for the recently released TopBraid Composer 3.5 is currently under development.

About TopBraid Composer

TopBraid Composer™ from TopQuadrant is an enterprise-class platform for developing Semantic Web ontologies and building semantic applications.

Fully compliant with W3C standards, Composer offers comprehensive support for developing, managing and testing configurations of knowledge models and their instance knowledge bases. Composer provides a flexible and extensible framework with a published API for developing semantic client/server or browser-based solutions that can integrate disparate applications and data sources.

About VISTology

VISTology, Inc. is a software research and development corporation based in Framingham, MA, committed to the innovative design and development of adaptive and semantically rich information solutions. VISTology's technical expertise is in formal semantics, information fusion, situation awareness, object-oriented modeling & design, and artificial intelligence. Since 1997, VISTology has led numerous R&D projects funded through contracts with clients that include the US Army, DARPA, AFRL/AFOSR, ONR, and MDA.

Licensing

BaseVISor 2.0 can be licensed for academic and research use free of charge; all other uses require a commercial license. Contact sales@vistology.com.

For more information, visit <http://www.vistology.com/BaseVISor/>