How Ontologies and Rules Help to Advance Automobile Development

RuleML 2007
Juergen Angele, ontoprise GmbH
Onoprise is a leading semantic software company. Our goal is to make a company's know-how visible and reusable.

**Founded:** 1999  
**Team:** approx. 50 Employees  
**Headquarter:** Karlsruhe  
**Market:** 9 out of the 20 largest German companies are our customers  
Strategic Partner for Oracle and Software AG
Karlsruhe: Location for Semantic Technologies

AIFB

Basic Research
Application-oriented Research

Application-oriented Research
Know-how Transfer
Realizing new Scenarios

Application-oriented Research
Product Development
Innovative Solutions

© 2007 ontoprise GmbH

www.ontoprise.de
„If I knew that we have already implemented this component from this vendor somewhere else, I had calculated differently.“

„With this error message, I always exchange the complete PC. Is there another way?“

„While I am still busy collecting all the information from all systems most of the customers already ran out of patience“
Semantics = All About Meaning

"An Ontology is a shared conceptualization of a domain [Tom Gruber]."

- social semantics (meaning)

Ontology

Railroad Object

- Train
- Track

Steam Train

- Mogul

Mogul Train

Mogul Emperor

Mogul Ski Race
Parts of an Ontology

An Ontology is a formal and defined System of Concepts and Relations between these Concepts used to describe complex domains of knowledge.

- Semantics (Meaning)
- Relations
- Rules

IF gage width = 1435 THEN suitable for standard gage ELSE narrow gage
Semantic Web Layer

OntoBroker OWL

SparQL

OWL

DLP bit of OWL/Rul

RDF Schema

RDF Core

XML

Namespaces

URI

Unicode

Challenges

- Majority of innovations is in electronical equipment
- Increasing complexity in development and integration
- Shorter development cycles
- Increasing quality measures

Goals

- Introduction of efficient testing methods to reduce manual work
- Manage complexity
- Increase transparency

Solution

- Extraction of rules from requirement specification and functional frames
- Collection of expert knowledge from engineers in free text
- Carve out of logic by means of an ontology
- Automatic validation of test results by means of specifications
Use Case

• Audi Valvelift System (AVS)

• 2 different Cam Contours for small and large Valve Lift

• Increases Engine Efficiency (more Power, lower Fuel Consumption)

• Controlled by Engine Management System
  • Deterministic Finite Automaton
  • $S1, S4$ - small, large Valve Lift
  • Transition Functions
Use Case

- Observable Variables during HiL Tests
- Snapshots at different Times
- Ontology Reflects Data Structure recorded during HiL Tests
- Introduces Terms as known to Experts (e.g., engineSpeed)
Collection of rules

Documentation
Requirement Specification, Descriptions, Functional Framework (Word, Excel, PDF)

Reverse Engineering

Expert knowledge

Interviews

Collection of rules (natural language)
Ontology- and rule development

advantages:
• Set of rules is extendable in an incremental way
• Rules are not hidden in program code.
• Rules are automatically explainable.
Collecting rules
Ontology- and rule development

ECU Specification: „If the engine speed is greater than 4000, the valve lift system must switch to S4 if it is in S1.”

?S[nextState->S4] <-
?S:Situation[state->S1, engineSpeed->?V] and
?V > 4000.

Experts: „At idle speed the small valve lift must be used.“

ERROR(?S) <-
?S:Situation[state->S4, idle->1].
Testing measurements

Ontology and rules

OntoBroker

Analysis result

In situation s2 still state 4 holds. In the situation before state 4 hold as well. The rotary speed was larger than the threshold. Therefore a state transition should have happened.
Functionality

- Support of internal order processing for building and rebuilding testcars (AVx)
- Integrate cross department dependencies into AVx
- The knowledge about functional, geometrical and processual dependencies is spread over many engineers

Goal

- Reduce time for testcar lifecycle and therefore for the whole development cycle
- Utilize expertise of engineers to improve testcar process
- Prevent time-lags in testcar process

„There is no other technology to both describe this level of complexity and being flexible enough to adapt to changes. ontoprise's technology enables us to describe and make executable our complex domain in a flexible and maintainable manner.“
Ontology combines rules, structures and information

Ontology

Structures
Dependencies, rules

Mapping of existing information
Ontologies represent the meaning of information.

Rules:
- Parts must be approved before you can test them embedded.
- Only the Person responsible for a part can approve its testing.

Part needs filtering before testing.
**Example Rule:** The maximum power of the motor must not exceed the one of the brakes: \( P_{\text{motor}} < P_{\text{brakes}} \)
What our customer likes on Flogic

F-Logic

- declarative (logic-based)
- clear semantics (well-founded semantics)

- powerful (rules, functions, negations)
- ontology based structuring (frame)
- schema reasoning
- simple human readable syntax
- homogenous rule and query syntax
- logical model of a domain

- modelling environment
- close integration into databases
- IT infrastructure
- handling large amounts of data
- fast engine
What our customer dislikes on Flogic

F-Logic

- it is not a standard!
Conclusion

Ontologies and Rules ...

- increase the transparency
  - by carving out logics from applications and data
  - because all results are explained in natural language

- make complexity manageable
  - because informal and distributed knowledge is formalized and therefore made machine processable
  - because knowledge can be structured and re-used

- help to build flexible systems that can adapt to changes quickly

but we need a standard!
Thank you!

Jürgen Angele, angele@ontoprise.de