ACCELERATE DEVELOPMENT | IMPROVE SERVICE | OPTIMISE SALES

ontoprise
know how to use Know-how

How Ontologies and Rules Help to Advance Automobile Development

www.ontoprise.de

RuleML 2007
Juergen Angele,
ontoprise GmbH

know how to use Know-how!

Ontoprise 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

Basic Research

Application-oriented Research

Application-oriented Research

Know-how Transfer

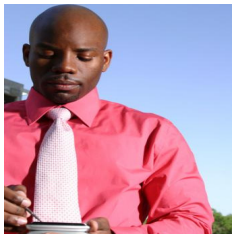
Realizing new Scenarios

Application-oriented Research

Product Development

Innovative Solutions

Field of Applications for ontoprise Technology



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

Support users in quickly finding information

SemanticMiner®

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

Reuse the know-how of your engineers

SemanticGuide®

„While I am still busy collecting all the information from all systems most of the customers already ran out of patience"

Single View on Multiple Sources

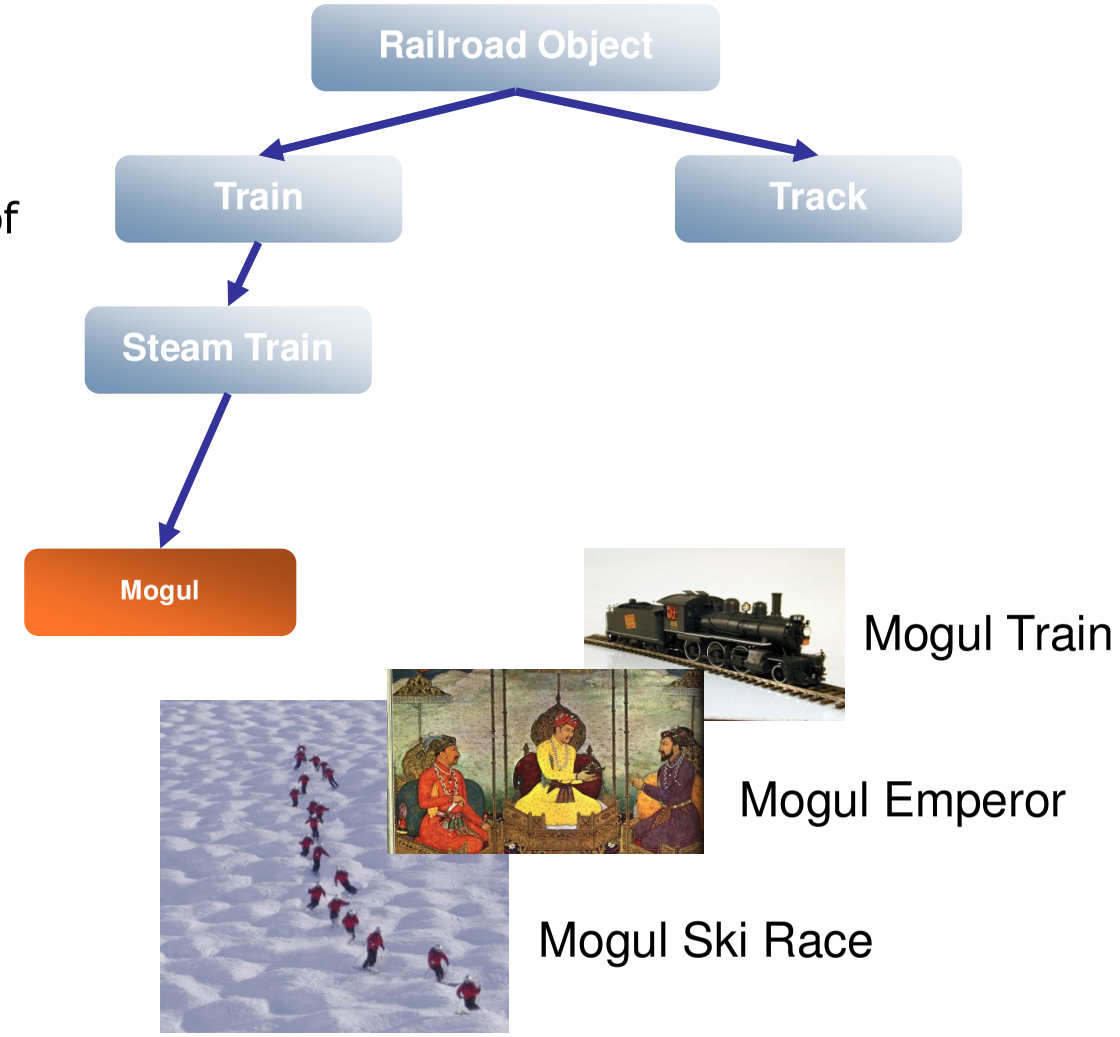
SemanticIntegrator®

Semantics = All About Meaning

Ontology

„An Ontology is a **shared** conceptualization of a domain [Tom Gruber].“

- social semantics (meaning)

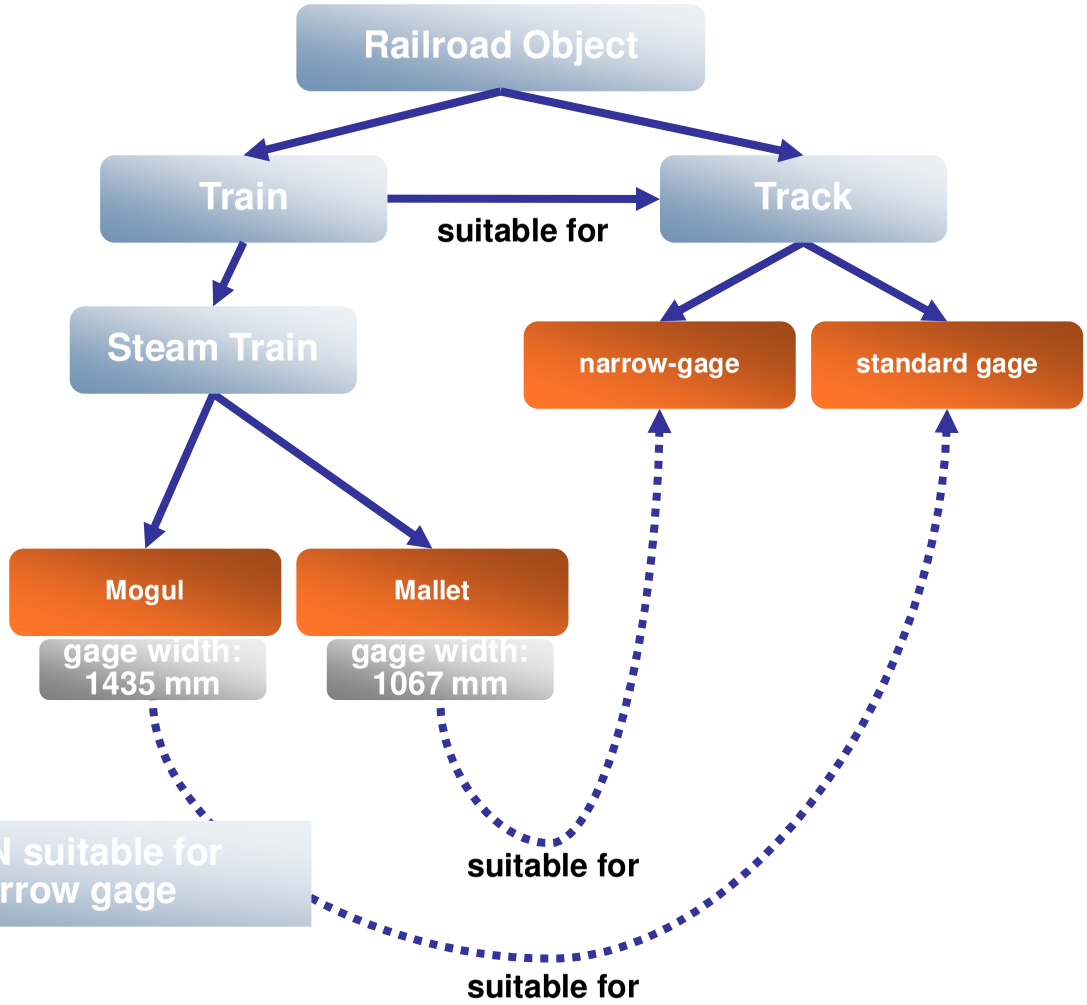


Parts of an Ontology

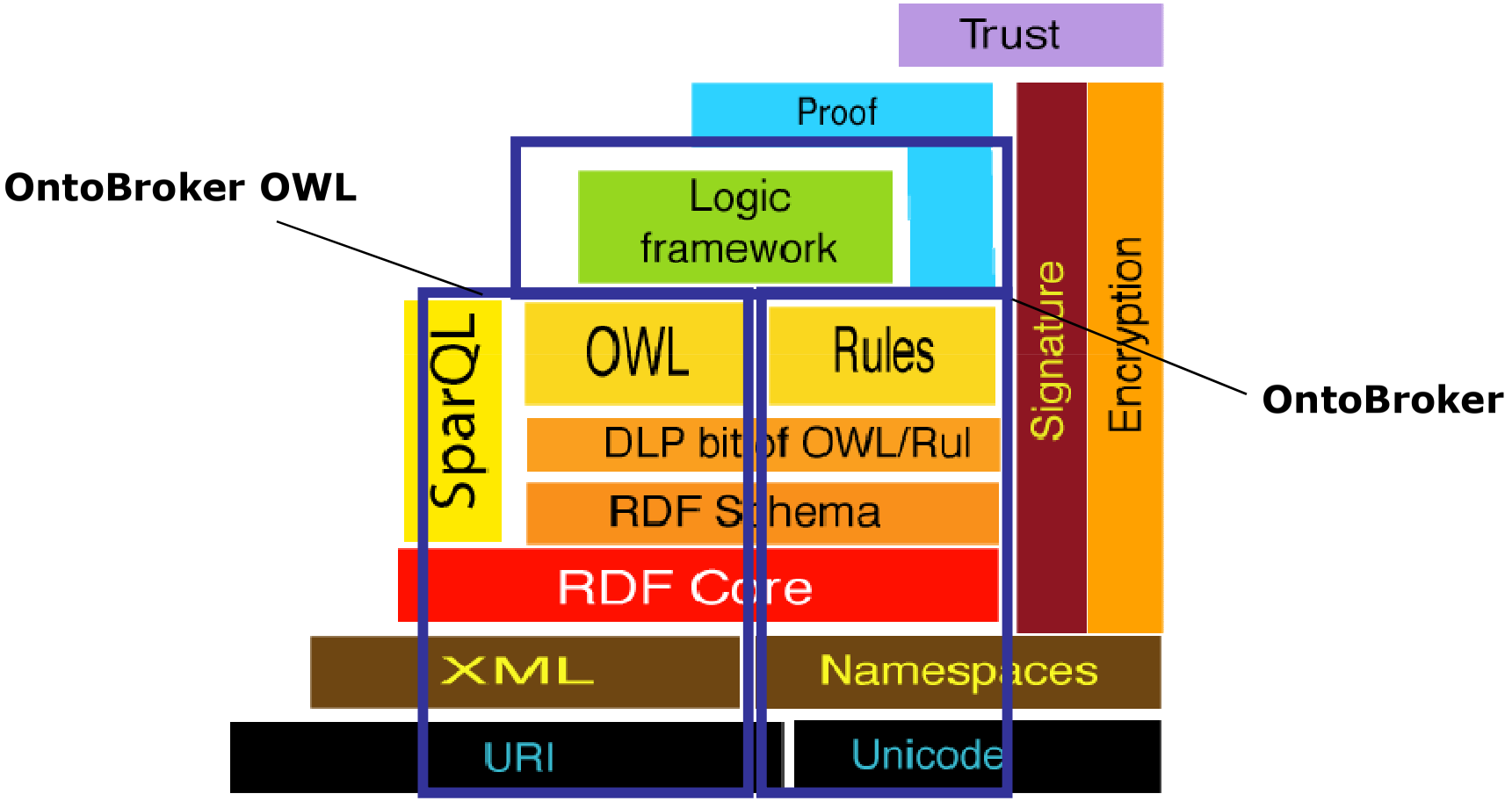
Ontology

„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



Semantic Web Layer



Tim Berners-Lee, ISWC November 2005, http://www.w3.org/2005/Talks/1110-iswc-tbl/#(12)

Automatic Analysis of Control Units for Audi

Challenges

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

Goals

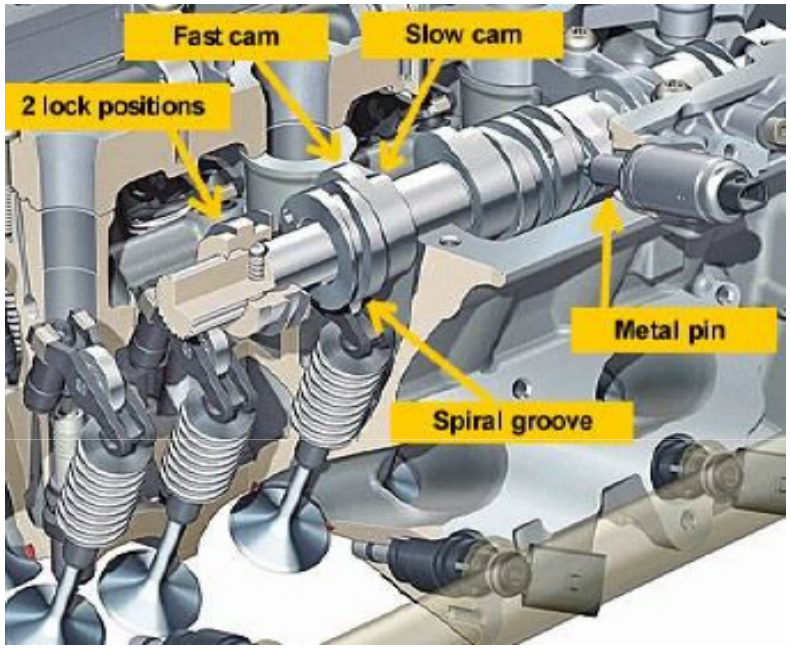
- Introduction of efficient testing methods to reduce manual work
- Manage complexity
- Increase transparency

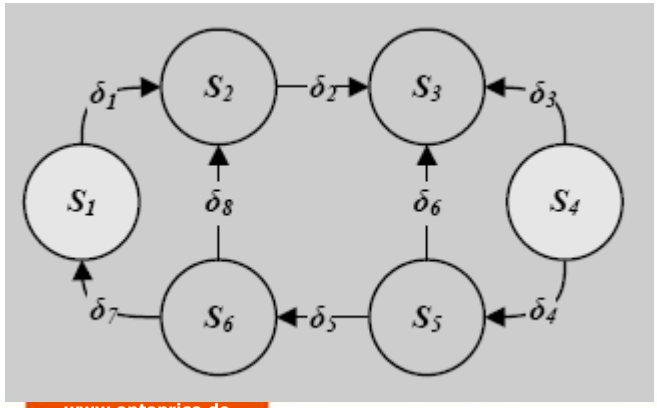
Solution

- 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

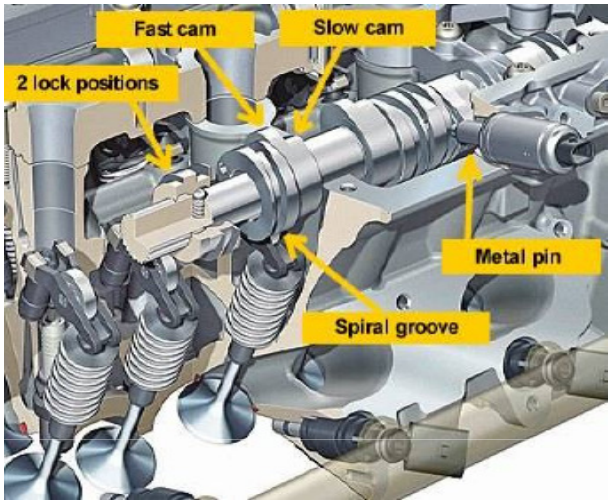
- Audi Valvelift System (AVS)
- 2 different Cam Contours for small and large Valve Lift
- Increases Engine Efficiency (more Power, lower Fuel Consumption)
- Controlled by Engine ManagementSystem
  - Deterministic Finite Automaton
  - *S1, S4 - small, large Valve Lift*
  - Transition Functions

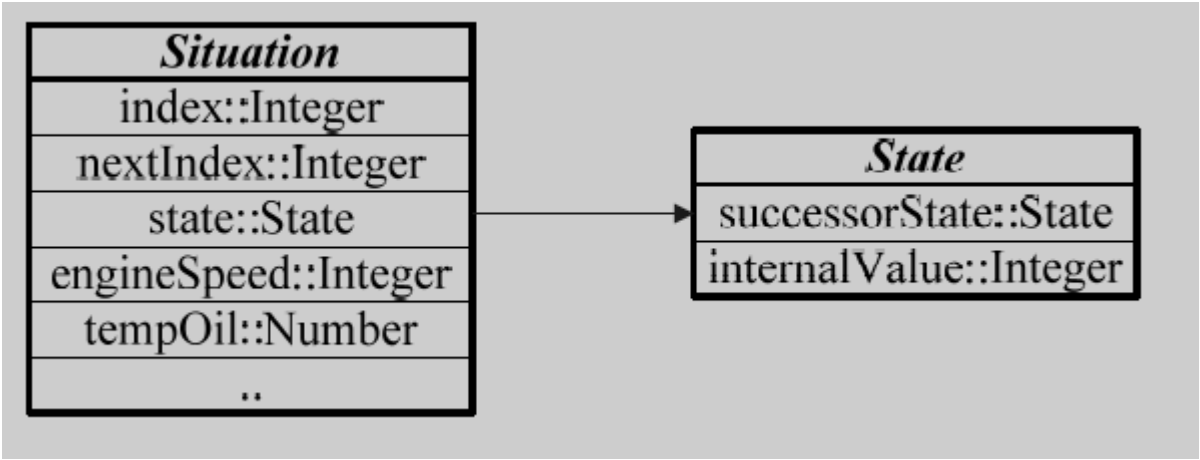




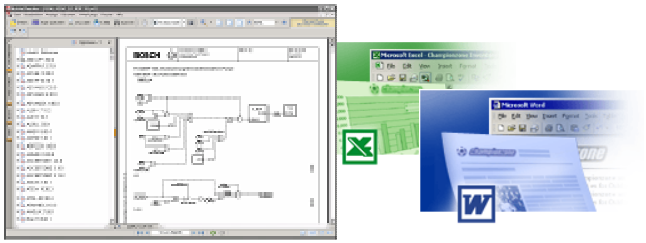
Use Case

- 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)

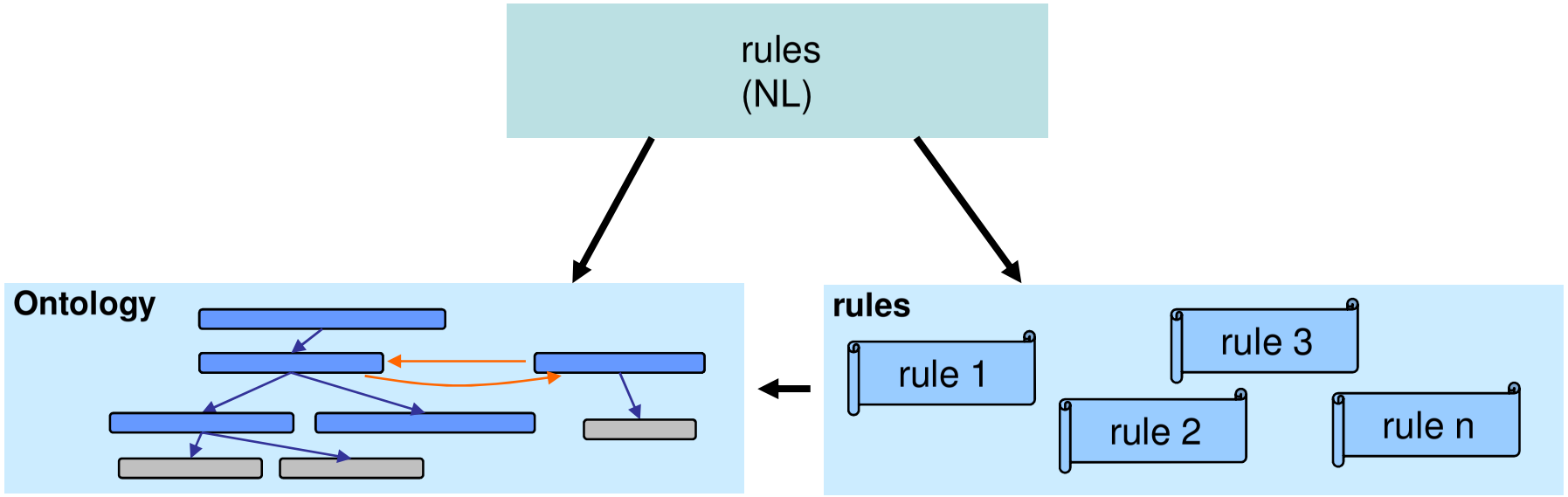
Expert knowledge

Reverse Engineering

Interviews

Collection of rules (natural language)

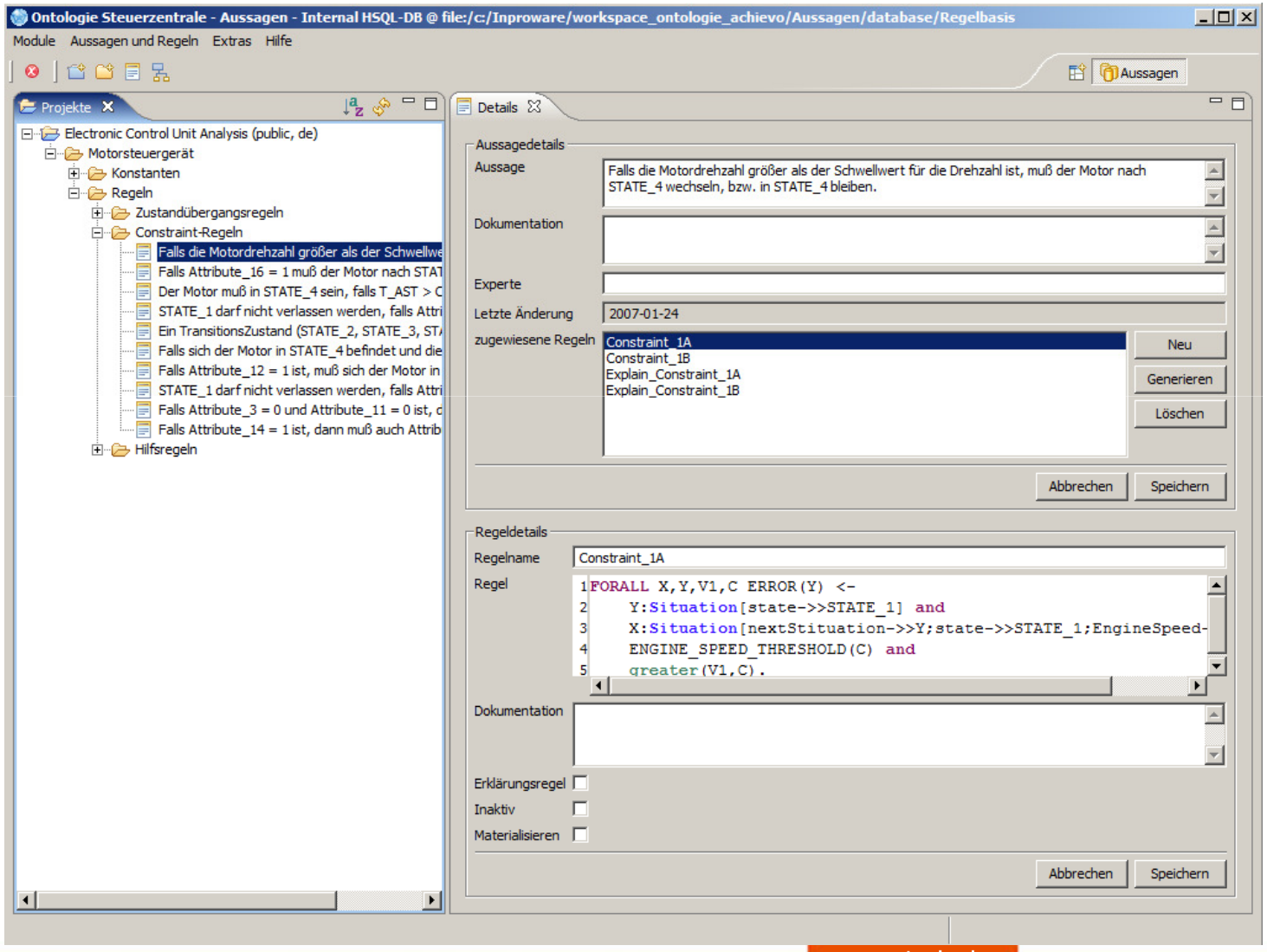
Ontology- and rule development



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

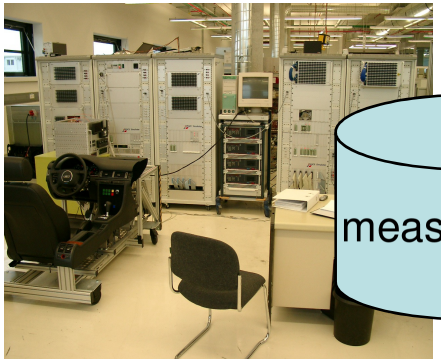
ECU Specification: *„If the engine speed is greater than 4000, the valve lift system must switch to S4 if it is in S1."*

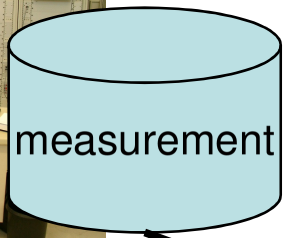
?S[nextState->S4] <-
?S:Situation[state->S1, engineSpeed->?V] and
?V > 4000.

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

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

Testing measurements



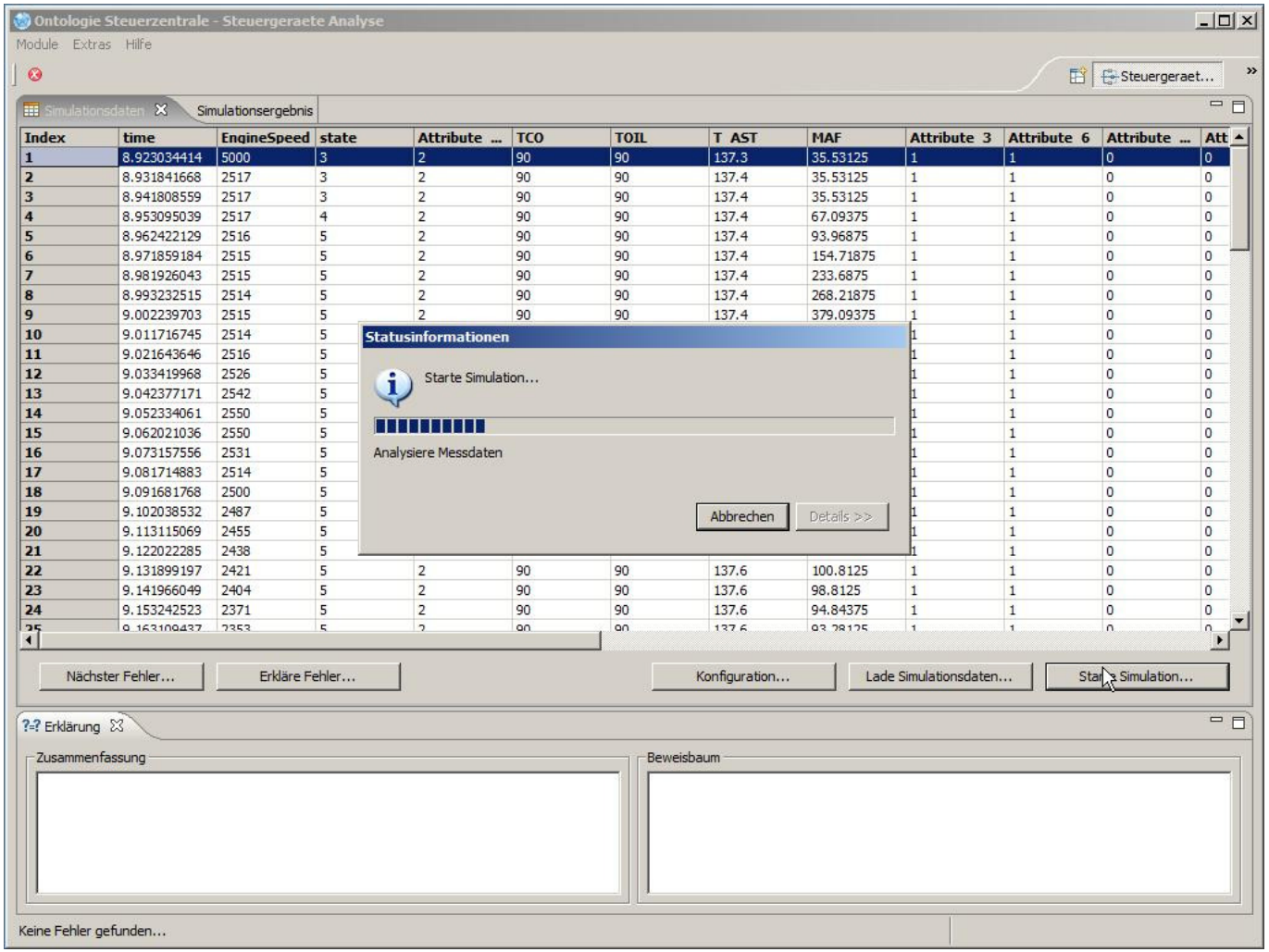


Ontology and rules

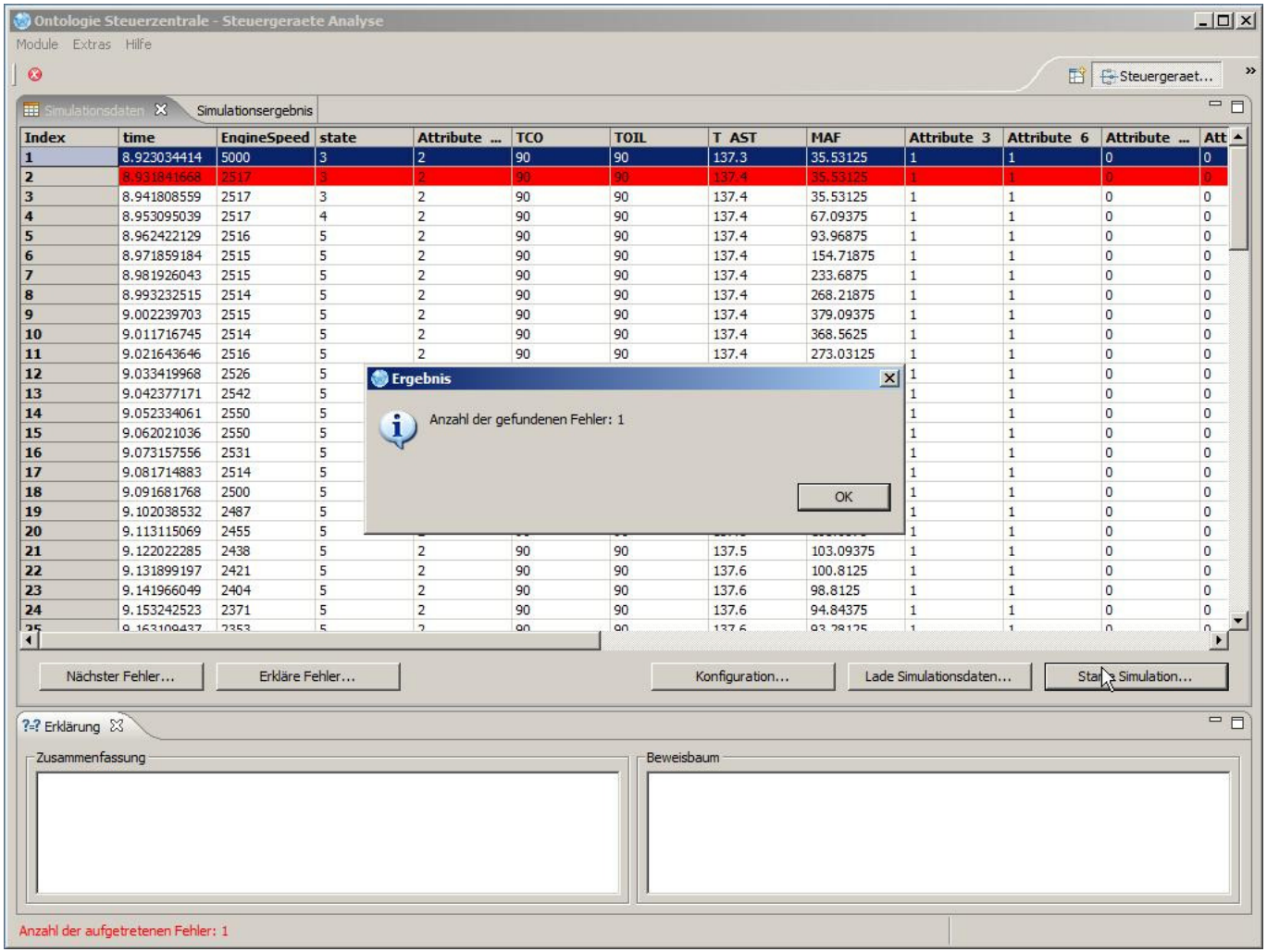
OntoBroker

Analysis result

Analyzing data

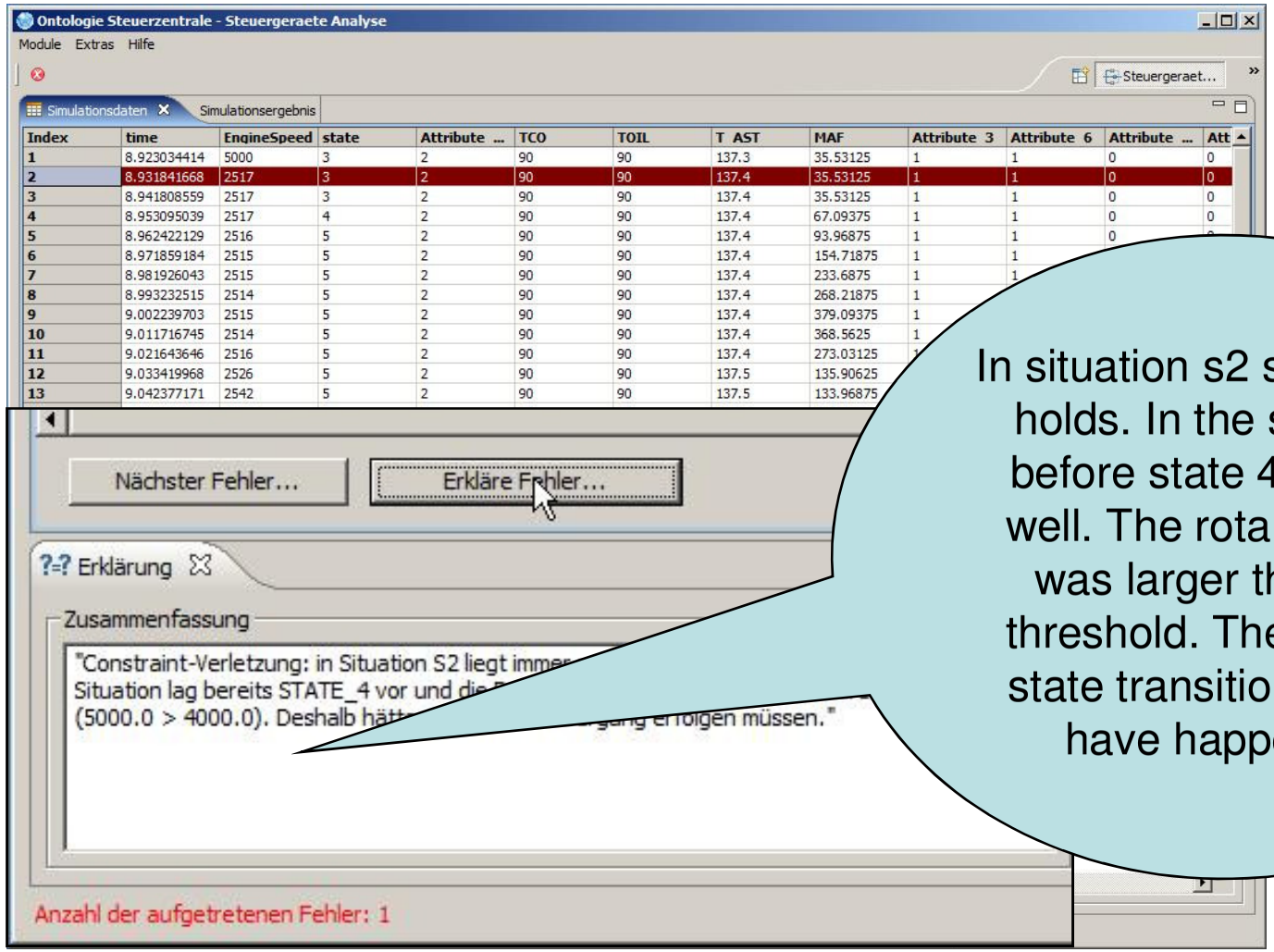


Found errors



Index	time	EngineSpeed	state	Attribute ...	TCO	TOIL	T_AST	MAF	Attribute_3	Attribute_6	Attribute ...	Att
1	8.923034414	5000	3	2	90	90	137.3	35.53125	1	1	0	0
2	8.931841668	2517	3	2	90	90	137.4	35.53125	1	1	0	0
3	8.941808559	2517	3	2	90	90	137.4	35.53125	1	1	0	0
4	8.953095039	2517	4	2	90	90	137.4	67.09375	1	1	0	0
5	8.962422129	2516	5	2	90	90	137.4	93.96875	1	1	0	0
6	8.971859184	2515	5	2	90	90	137.4	154.71875	1	1	0	0
7	8.981926043	2515	5	2	90	90	137.4	233.6875	1	1	0	0
8	8.993232515	2514	5	2	90	90	137.4	268.21875	1	1	0	0
9	9.002239703	2515	5	2	90	90	137.4	379.09375	1	1	0	0
10	9.011716745	2514	5	2	90	90	137.4	368.5625	1	1	0	0
11	9.021643646	2516	5	2	90	90	137.4	273.03125	1	1	0	0
12	9.033419968	2526	5						1	1	0	0
13	9.042377171	2542	5						1	1	0	0
14	9.052334061	2550	5						1	1	0	0
15	9.062021036	2550	5						1	1	0	0
16	9.073157556	2531	5						1	1	0	0
17	9.081714883	2514	5						1	1	0	0
18	9.091681768	2500	5						1	1	0	0
19	9.102038532	2487	5						1	1	0	0
20	9.113115069	2455	5						1	1	0	0
21	9.122022285	2438	5	2	90	90	137.5	103.09375	1	1	0	0
22	9.131899197	2421	5	2	90	90	137.6	100.8125	1	1	0	0
23	9.141966049	2404	5	2	90	90	137.6	98.8125	1	1	0	0
24	9.153242523	2371	5	2	90	90	137.6	94.84375	1	1	0	0

Explaining inferences



Index	time	EngineSpeed	state	Attribute ...	TCO	TOIL	T_AST	MAF	Attribute_3	Attribute_6	Attribute ...	Att
1	8.923034414	5000	3	2	90	90	137.3	35.53125	1	1	0	0
2	8.931841668	2517	3	2	90	90	137.4	35.53125	1	1	0	0
3	8.941808559	2517	3	2	90	90	137.4	35.53125	1	1	0	0
4	8.953095039	2517	4	2	90	90	137.4	67.09375	1	1	0	0
5	8.962422129	2516	5	2	90	90	137.4	93.96875	1	1	0	
6	8.971859184	2515	5	2	90	90	137.4	154.71875	1	1		
7	8.981926043	2515	5	2	90	90	137.4	233.6875	1	1		
8	8.993232515	2514	5	2	90	90	137.4	268.21875	1			
9	9.002239703	2515	5	2	90	90	137.4	379.09375	1			
10	9.011716745	2514	5	2	90	90	137.4	368.5625	1			
11	9.021643646	2516	5	2	90	90	137.4	273.03125				
12	9.033419968	2526	5	2	90	90	137.5	135.90625				
13	9.042377171	2542	5	2	90	90	137.5	133.96875				

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.

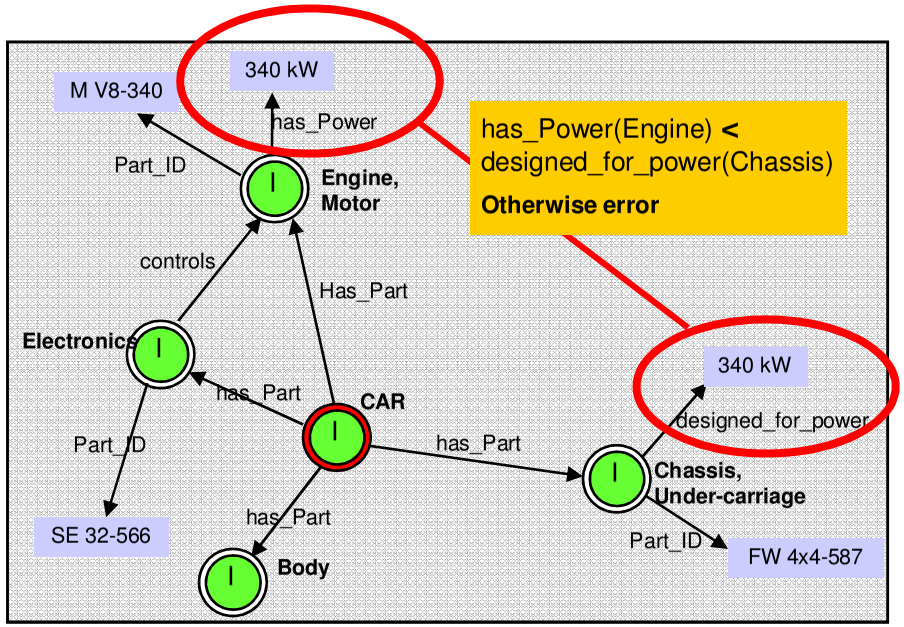
Ontology-based Testcar Configurator at Audi

Functionality

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

Goal

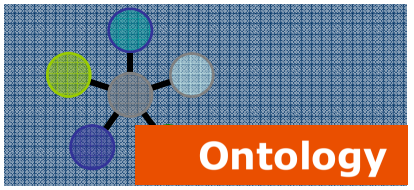
- Reduce time for testcar lifecycle and therefore for the whole development cycle
- Utilize expertise of engineers to improve testcar process
- prevent time-lags in testcar process



Sample Ontology (Source ontoprise)

„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."

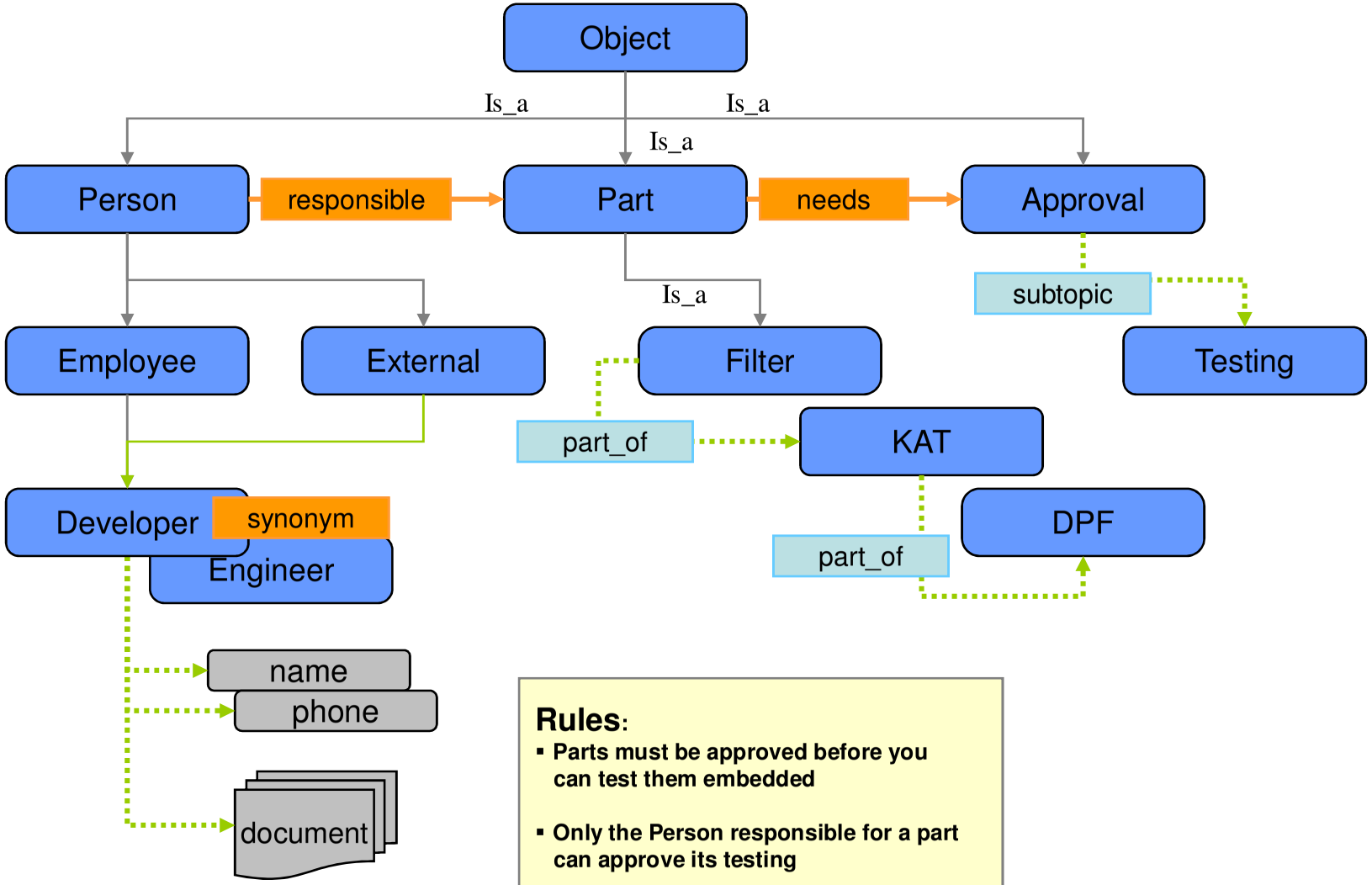
Ontology combines rules, structures and information



Structures
Dependencies, rules

Mapping of existing information

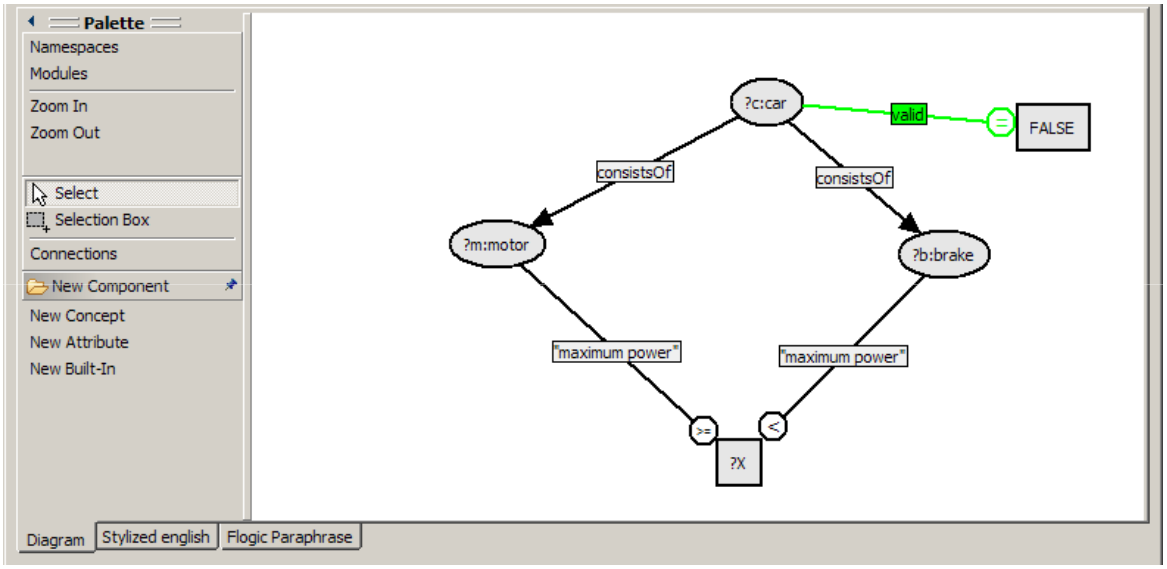
Ontologies represent the meaning of information

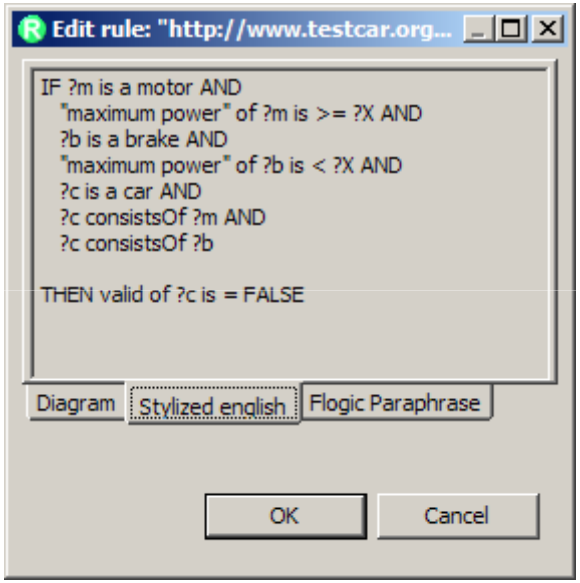


ontoprise
know how to use Know-how

Relationships/Constraints

Example Rule: *The maximum power of the motor must not exceed the one of the brakes: Pmotor < Pbrakes*






```
FORALL c, entity21, X, b, entity22, m
    c["http://www.testcar.org"#valid->>FALSE]@"http://www.testcar.org"#""
    <-
    m:"http://www.testcar.org"#motor@"http://www.testcar.org"#"" AND
    m["http://www.testcar.org"#"maximum power"->>entity21]@"http://www.testcar.org"#"" AND greaterorequal(entity21, X) AND
    b:"http://www.testcar.org"#brake@"http://www.testcar.org"#"" AND
    b["http://www.testcar.org"#"maximum power"->>entity22]@"http://www.testcar.org"#"" AND less(entity22, X) AND
    c:"http://www.testcar.org"#car@"http://www.testcar.org"#"" AND
    c["http://www.testcar.org"#consistsOf->>m]@"http://www.testcar.org"#"" AND
    c["http://www.testcar.org"#consistsOf->>b]@"http://www.testcar.org"#"".
```

What our customer likes on Flogic

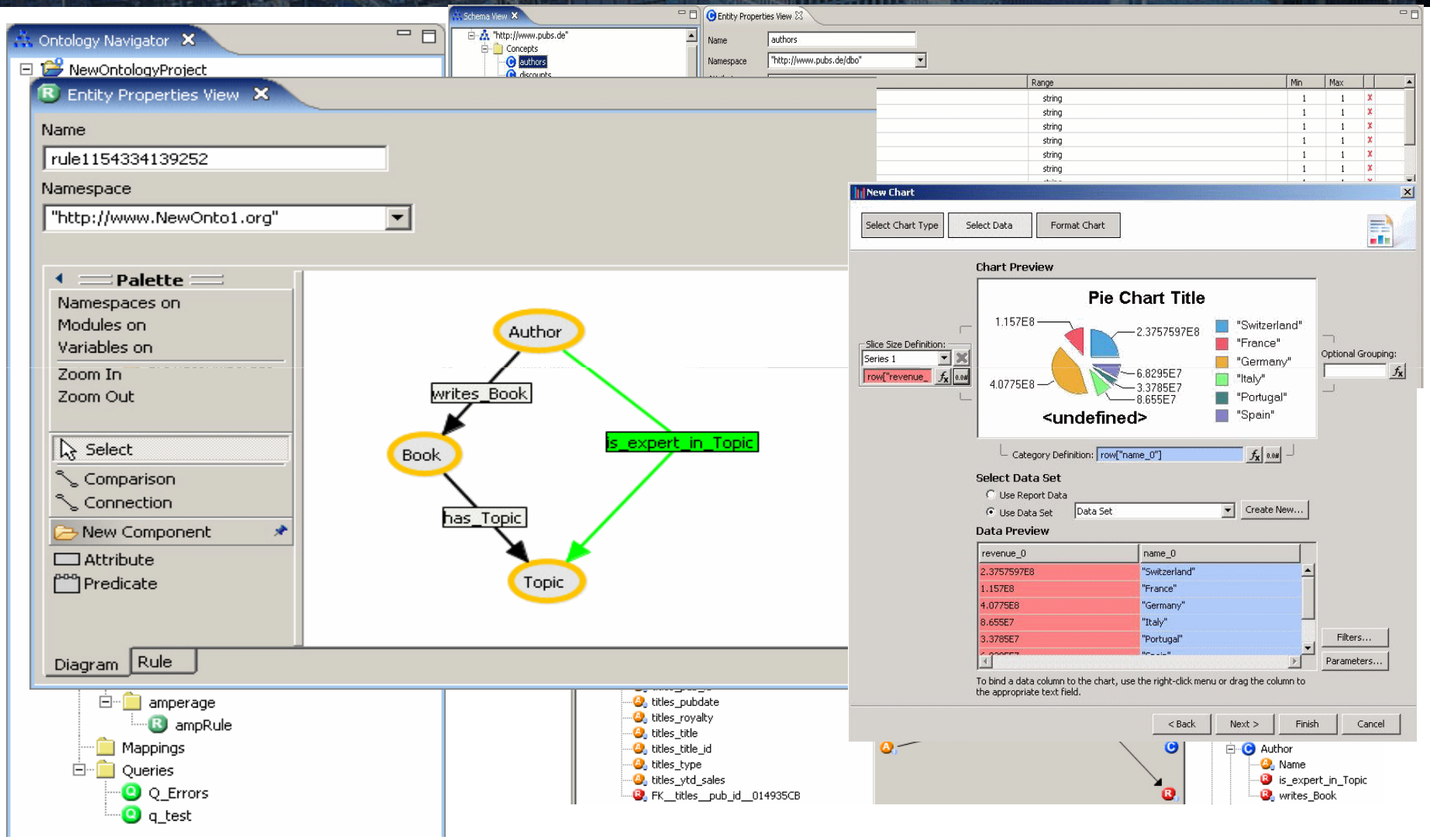
F-Logic

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

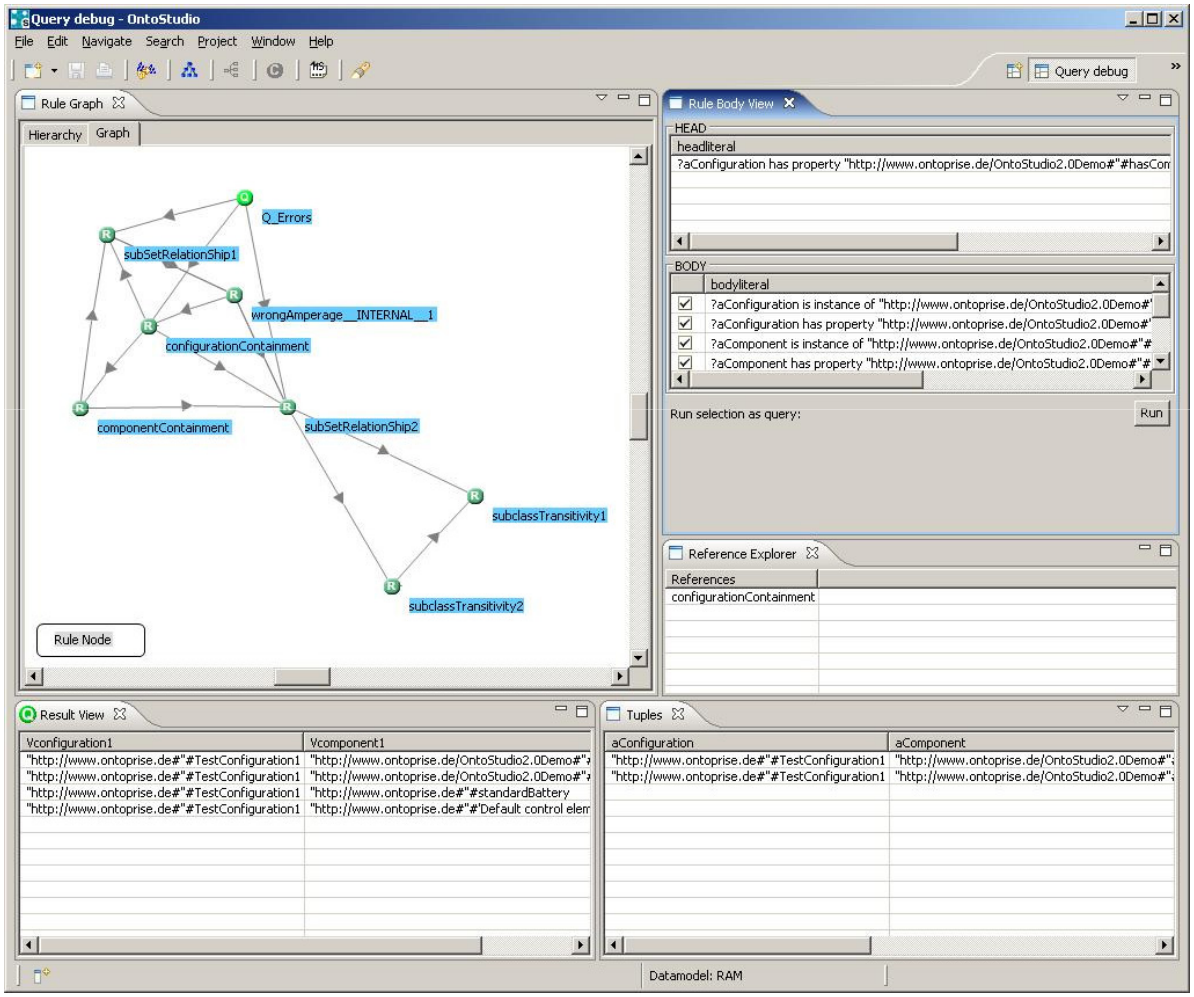
- powerful (rules, functions, negations)
- ontology based structuring (frame)
- schema reasoning
- simple human readable syntax
- homogenious rule and query syntax
- logical model of a domain

- modelling environment
- close integration into databases
- IT infrastructure
- handling large amounts of data
- fast engine

OntoStudio / NEON toolkit



OntoStudio / NEON toolkit



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

ontoprise