

ENTWICKLUNG
BESCHLEUNIGEN

SERVICE
VERBESSERN

VERTRIEB
OPTIMIEREN

The background of the slide is a cityscape with several skyscrapers. A large, semi-transparent silhouette of a person's head is overlaid on the city, with its top resting on a prominent, rounded building. The text "Know how to use Know-how" is centered in the left half of the image.

**Know how
to use Know-how**

www.ontoprise.de

Data Integration using Semantic Technology: A use case

**Jürgen Angele, ontoprise
Michael Gesmann, Software AG**

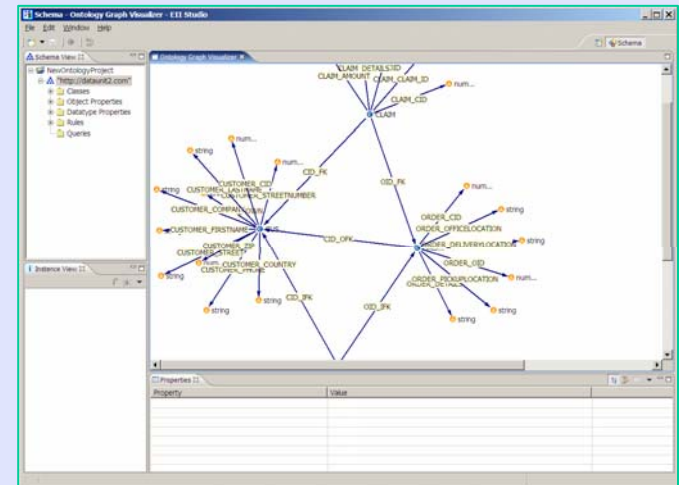
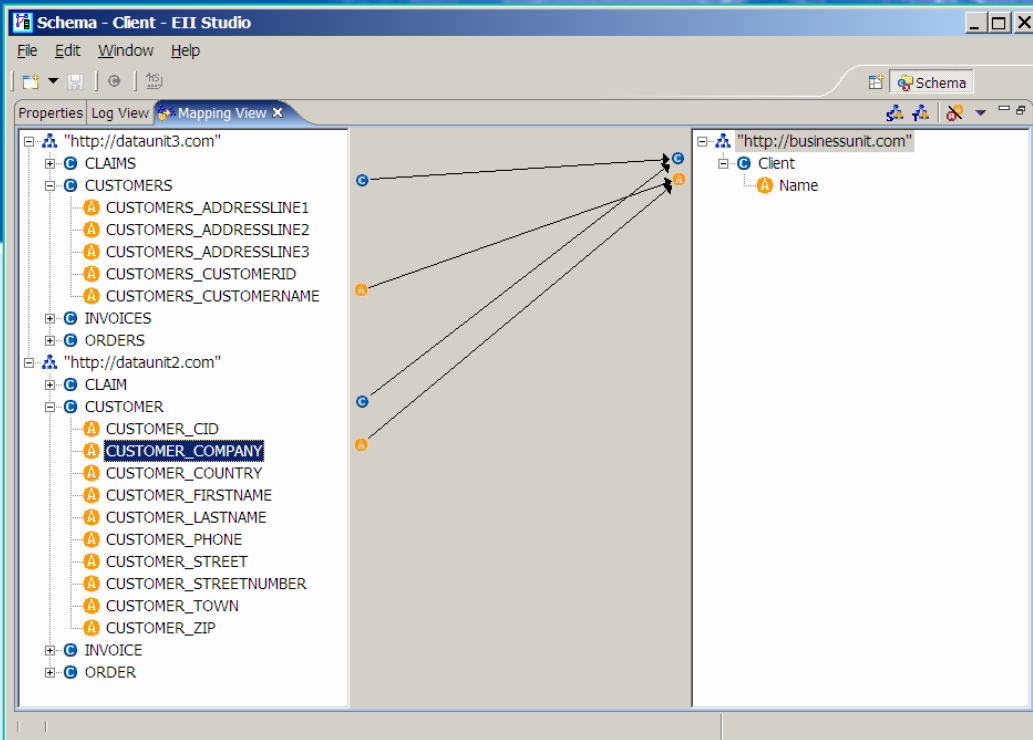
Software AG's Information Integrator



Information
Integrator
Studio



Powered by: ontoprise



Kinds of Integration Problems

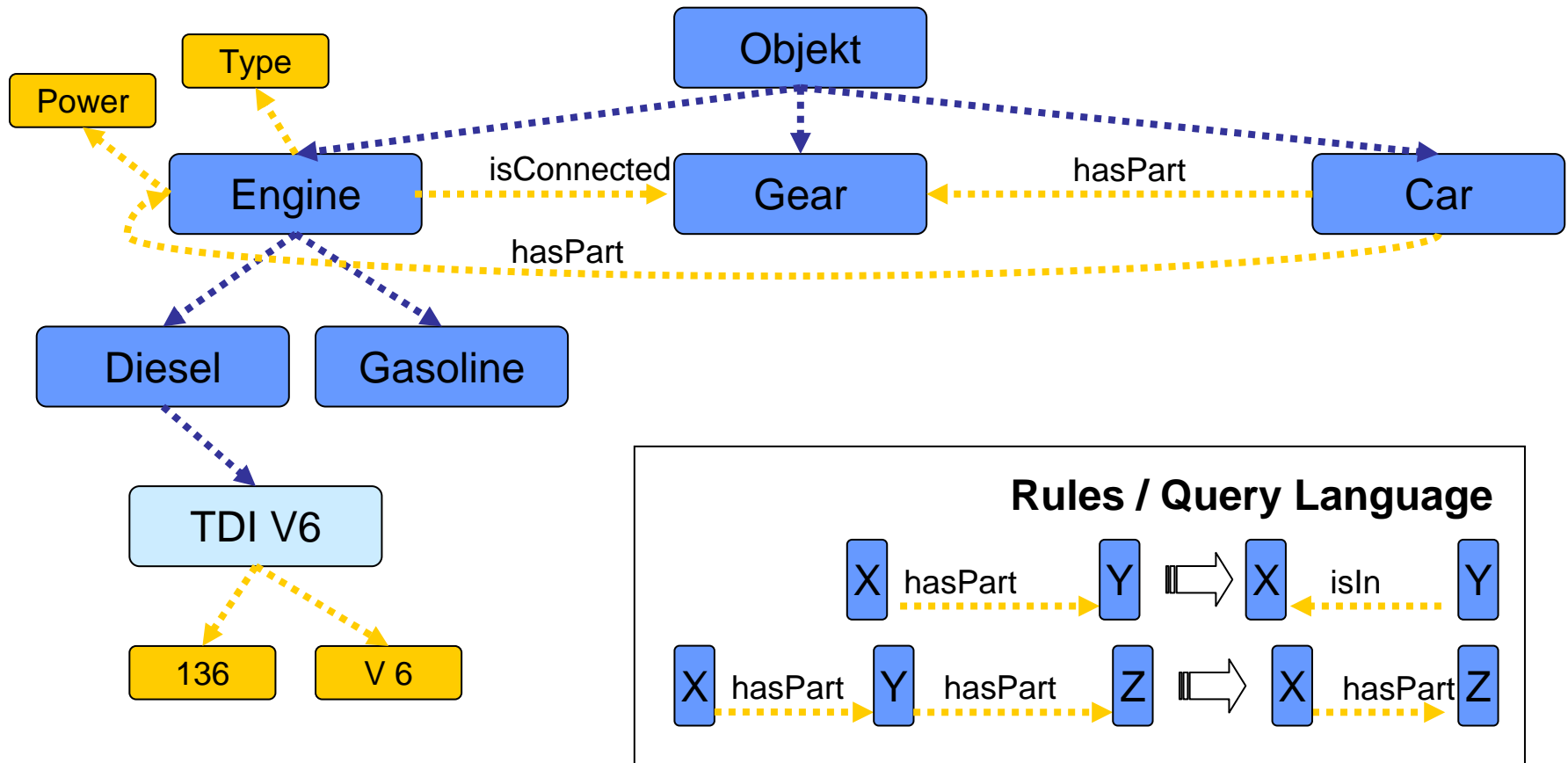
It is generally estimated that for each \$1 spent for an application, companies spend on average \$5 to \$9 for the integration.

© IBM, Nelson Mattos

What is the **problem** of information integration?

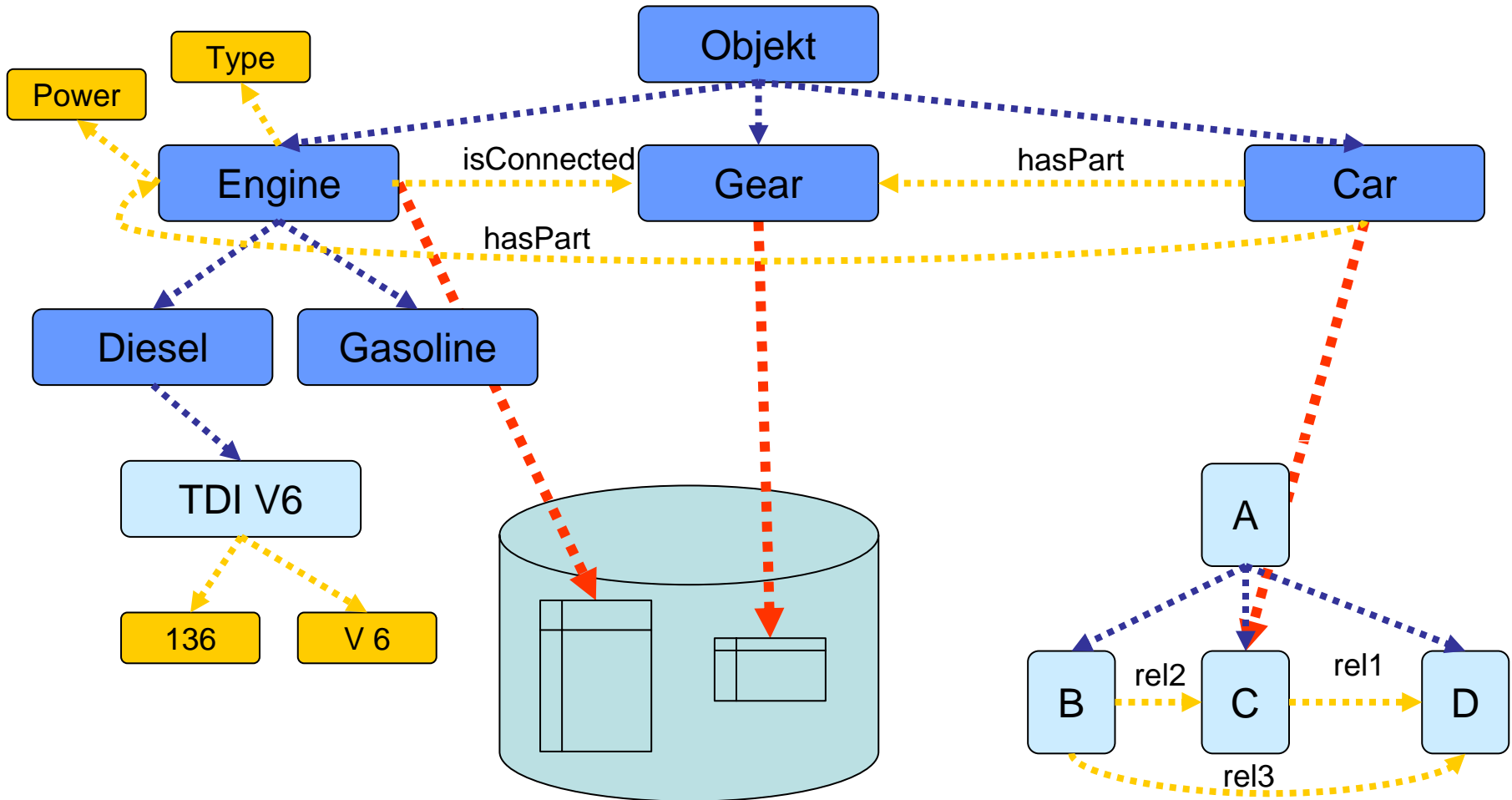
- **structural heterogeneity** – different application systems store their data in different structures
- **semantic heterogeneity** – intended meaning of information items is different in the various application systems
- **inconsistency and redundancy problems** – data in different application systems might be partially inconsistent or redundant

Ontology

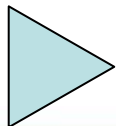


- Representation Language: F-Logic, WRL
- standards: RDF, OWL

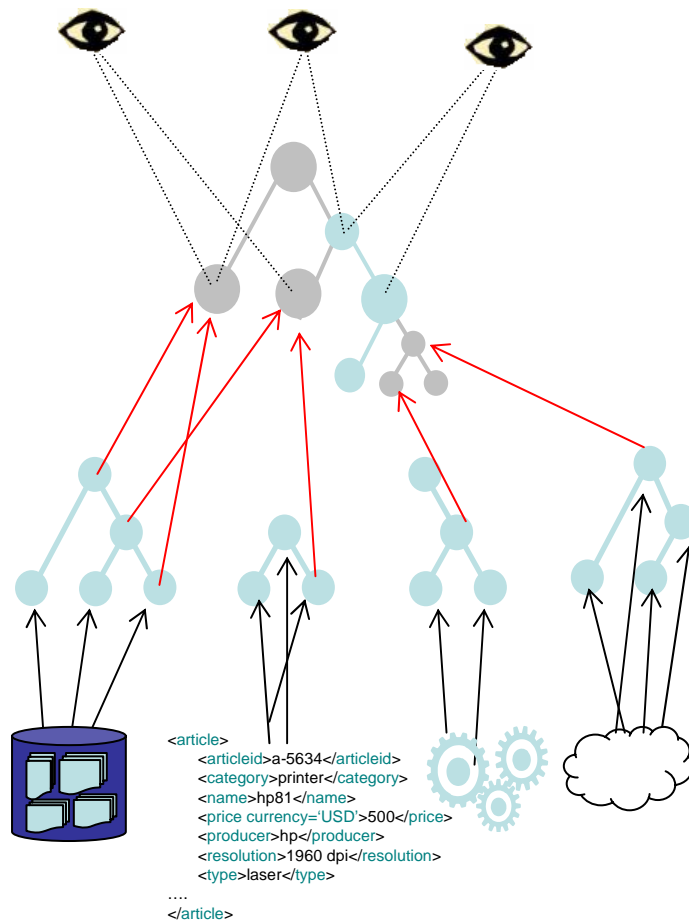
Mapping an Ontology with Data



- Mapping to Databases, Webservices, Search Engines
- Mapping to Ontologies



Concept



views

business ontology

manual mappings

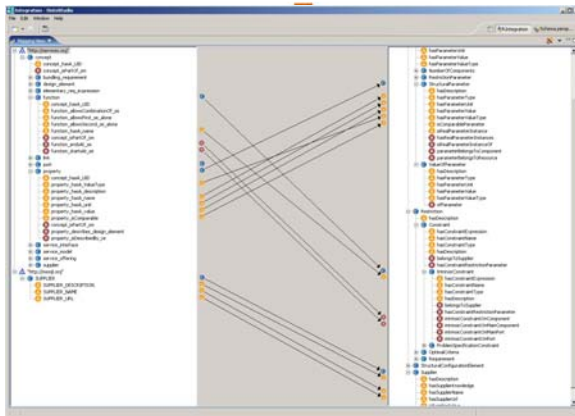
data source ontologies

automatic mappings

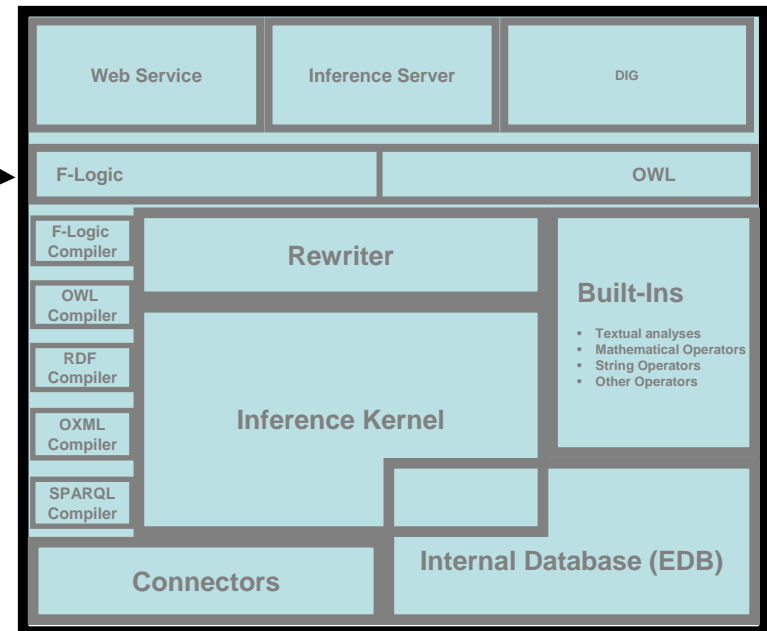
data sources

Information Integrator Studio / Semantic Server

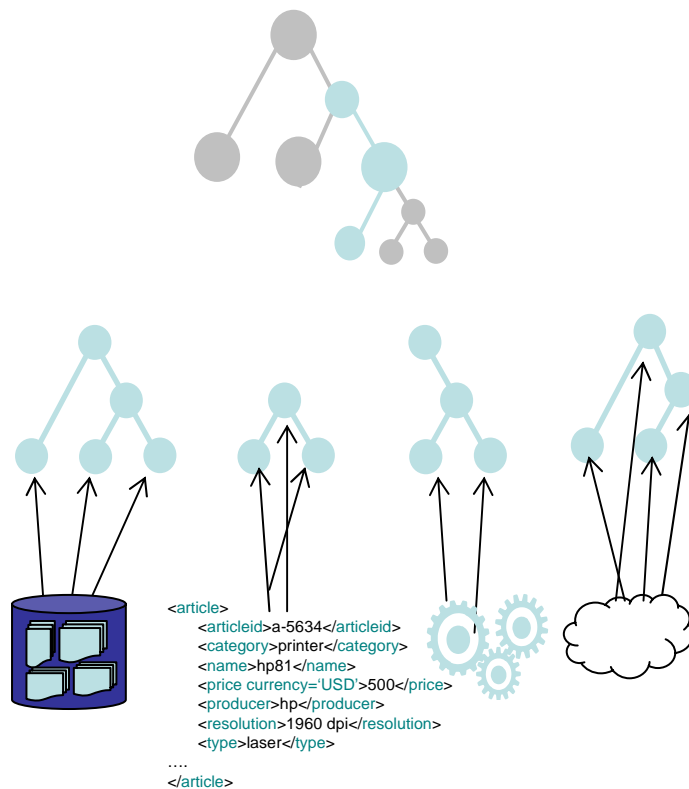
Studio



Semantic Server



Concept



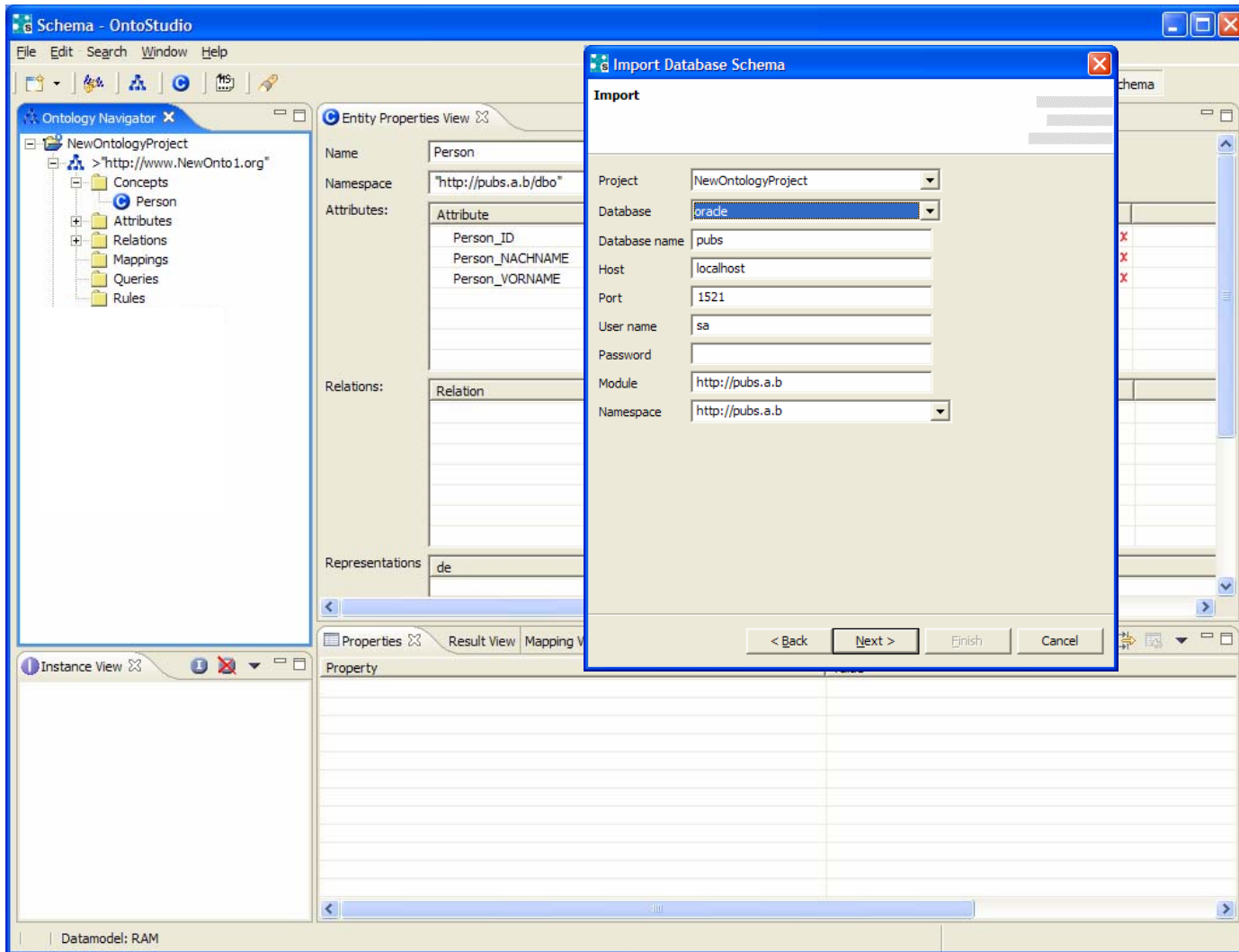
business ontology

data source ontologies

automatical mappings

data sources

Studio



Datasource Mapping Rules

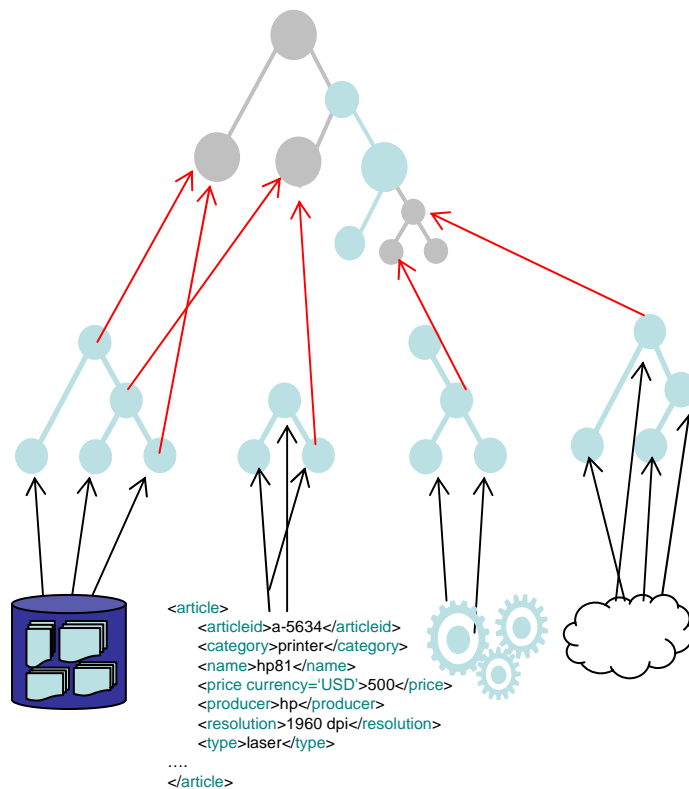
FORALL VAR1, VAR2, VAR3, VAR4, VAR5, VAR6, VAR7, VAR8

**c(authors_1347752484,VAR1):authors
[authors_au_id->>VAR1;authors_au_lname->>VAR2.....]**

<-

**dbaccessuser("\dbo\".\authors\"",
F("au_id", VAR1, "au_lname", VAR2,.....),
"mssqlserver2000", "pubs", "localhost:1433", "sa", "").**

Concept



business ontology

manual mappings

data source ontologies

automatic mappings

data sources

Studio

Schema - OntoStudio

File Edit Search Window Help

Ontology Navigator

- NewOntologyProject
 - >"http://www.NewOnto1.org"
 - Concepts
 - Person
 - Attributes
 - Relations
 - Mappings
 - Queries
 - Rules
 - >"http://pubs.a.b"
 - Concepts
 - Person
 - authors
 - Attributes
 - Relations
 - Mappings
 - Queries
 - Rules

Entity Properties View

Name: Person

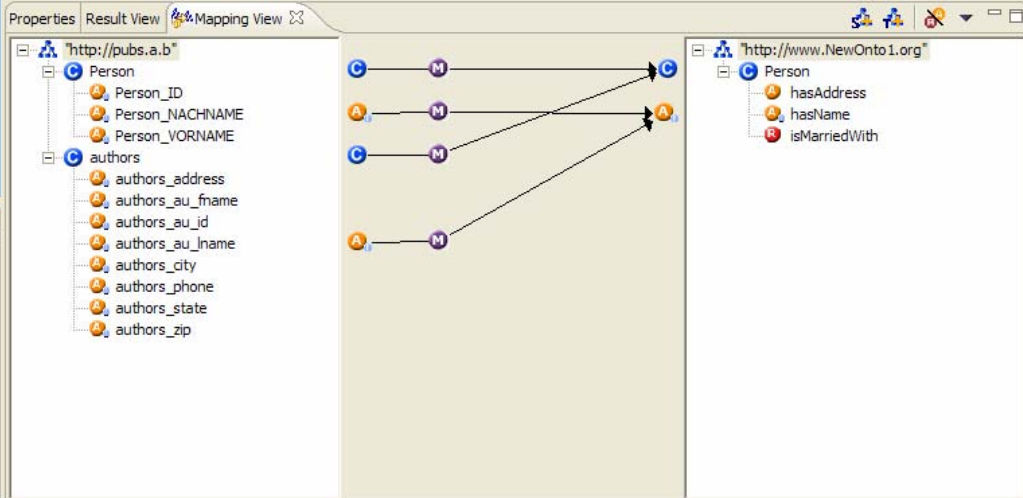
Namespace: "http://www.NewOnto1.org"

Attributes:

Attribute	Range	Min	Max	
hasAddress	string	0	N	X
hasName	string	0	N	X

Relations:

Relation	Range	Min	Max	
isMarriedWith	Person	0	N	X



Datamodel: RAM

Mapping Rules

FORALL VAR0, VAR1, VAR2, VAR3, VAR4, VAR5, VAR6, VAR7, VAR8

VAR0:Person

[hasName->>VAR2.....]

<-

VAR0:authors

[authors_au_id->>VAR1;authors_au_lname->>VAR2.....] .

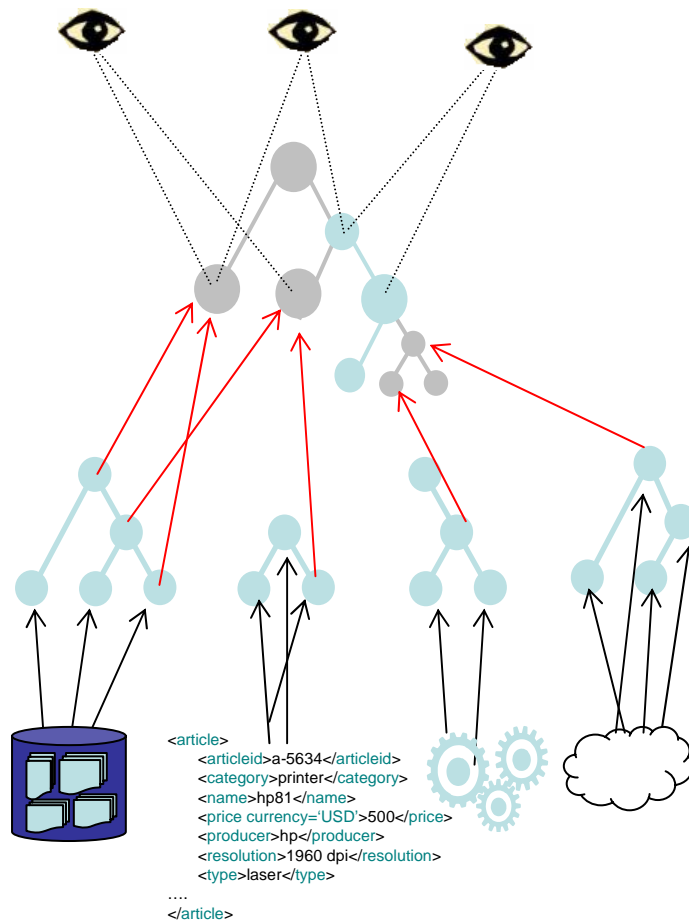
FORALL VAR0,VAR1,VAR2,VAR3

VAR0:LaserPrinter[hasResolution->>VAR1; hasPrice->>VAR3;...]

<-

**VAR0: Printer[ptype->>"Laser"; pres->>VAR1; pp->> VAR2]
 AND VAR3 is VAR2*1.3.**

Concept



views

business ontology

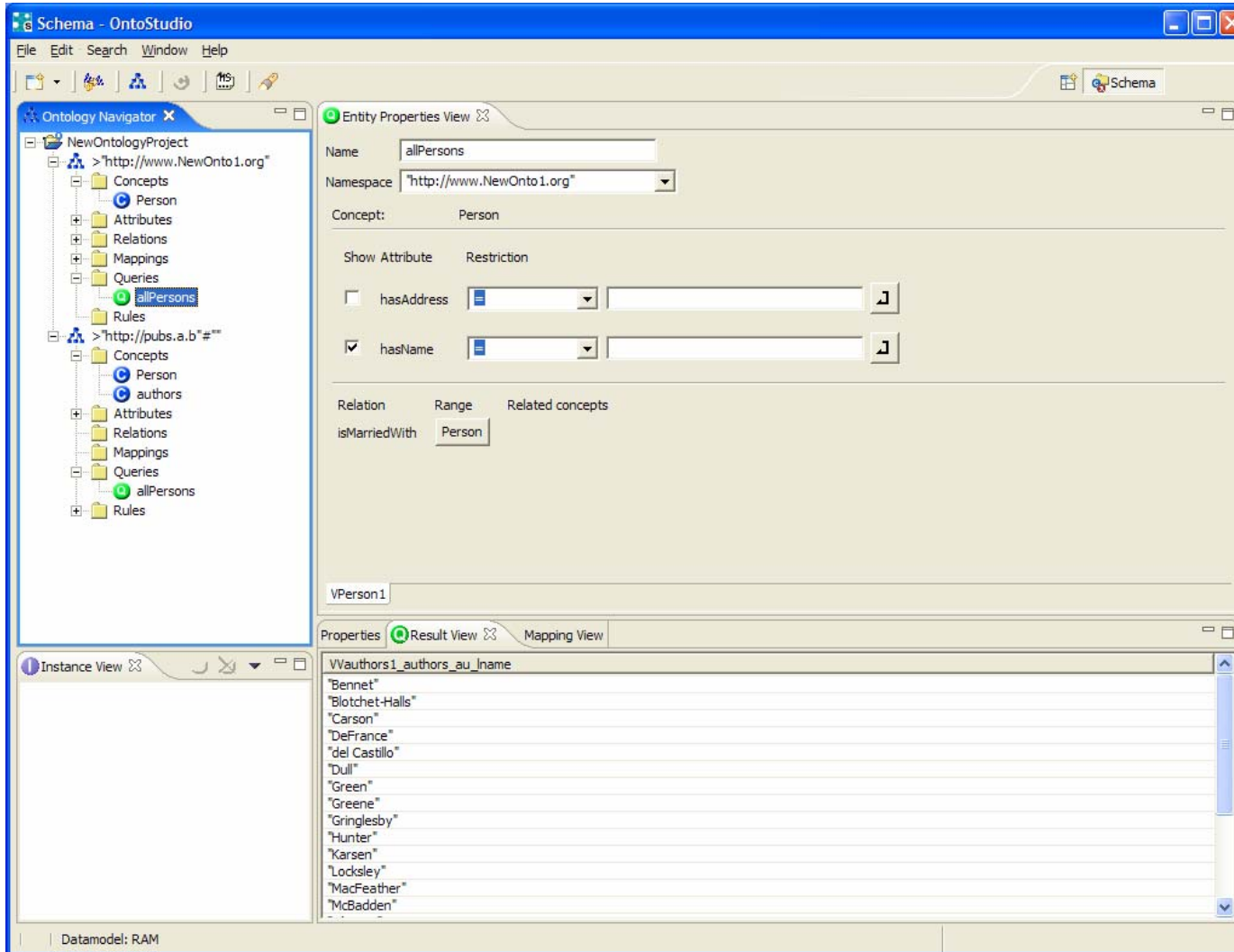
manual mappings

data source ontologies

automatic mappings

data sources

Studio



The screenshot displays the OntoStudio application window titled "Schema - OntoStudio". The interface is divided into several panes:

- Ontology Navigator:** A tree view on the left showing the ontology structure. It includes two namespaces:
 - `>http://www.NewOnto1.org*` containing folders for Concepts, Attributes, Relations, Mappings, Queries, and Rules, with a green circle icon next to `allPersons`.
 - `>http://pubs.a.b"#"` containing folders for Concepts, Attributes, Relations, Mappings, Queries, and Rules, with a green circle icon next to `allPersons`.
- Entity Properties View:** A central pane showing the configuration for the `allPersons` entity.
 - Name:** `allPersons`
 - Namespace:** `http://www.NewOnto1.org*`
 - Concept:** `Person`
 - Show Attribute Restriction:**
 - `hasAddress`: (disabled)
 - `hasName`: (enabled)
 - Relation Range Related concepts:**
 - `isMarriedWith`: `Person`
- Instance View:** A pane at the bottom left, currently empty.
- Properties / Result View / Mapping View:** A pane at the bottom right showing a list of instance values for the `Vauthors1_authors_au_lname` property:
 - "Bennet"
 - "Blotch-Halls"
 - "Carson"
 - "DeFrance"
 - "del Castillo"
 - "Dull"
 - "Green"
 - "Greene"
 - "Gringlesby"
 - "Hunter"
 - "Karsen"
 - "Locksley"
 - "MacFeather"
 - "McBadden"

The status bar at the bottom indicates "Datamodel: RAM".

Runtime Performance

Which is the right evaluation sequence for the rule bodies?

Constraint satisfaction problem:

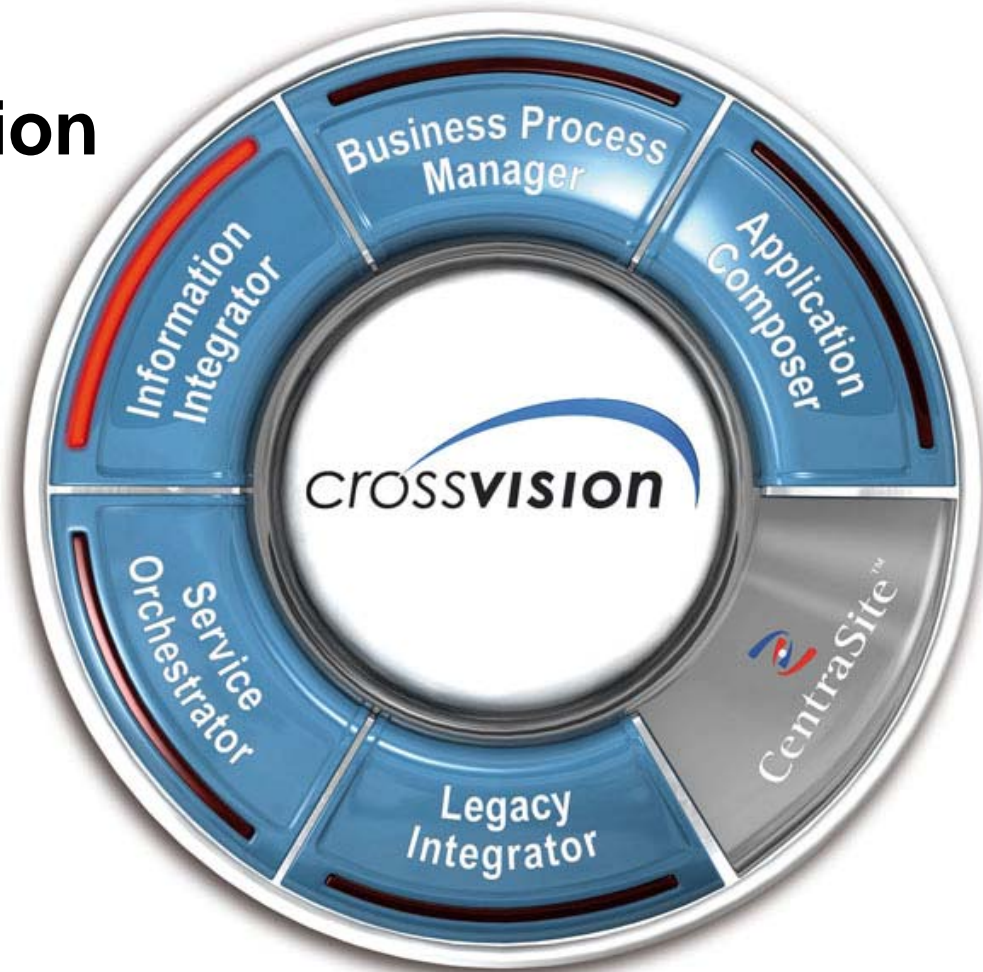
- **avoid table scans**
- **use key information, unique index information,....**

Solution:

- **heuristics**
- **optimizer**

- located in Darmstadt, Germany
- established in 1969
- about 2800 Employees
- customers in 73 countries
- core businesses:
 - ◆ Enterprise transaction systems (data management and processing)
 - ◆ crossvision – SOA integration suite

Semantic Integration



Customer Information Gateway

Does a customer raising a support issue for a specific project have a valid maintenance contract for that product?

What is a customer's experience with our products?



- **client**
- **contact information**
- **contract**



- **customer**
- **contact information**
- **support case**

Customer Information Gateway – Source ontologies

Class „File151“
property „AA“ type integer
property „AB“ type date
property „AC“ type date
property „AD“ type string
...
Class „File87“
...

Class „CUSTOMER“
property „cid“ type integer
...
Class „CASE“
property „desc“ type string
property „cust “ type integer
property „belongsTo“ type „CUSTOMER“
...

Customer Information System (Adabas)

File 151

Fields „AA“, „AB“, „AC“, „AE“

File 87

Fields „AA“, „AB“, „AC“

Support Information System (SQL)

Table „CUSTOMER“

Table „CASE“

Customer Information Gateway - Mapping

Class Customer

property id
property name
property address

Class Contract

property contractId
property contractEndDate
property contractor

Class SupportRequest

property id
property status
property issuedBy

File 151 -> Contract

AA -> contractId
AB -> contractStartDate
AC -> contractEndDate
AD -> contractStatus

File 87 -> Client

...

Class „File151“
Class „File 87“

Class „CUSTOMER“
Class „CASE“

Adabas

SQL

Customer Information Gateway - Mapping

Class Customer

property id
property name
property address

Class Contract

property contractId
property contractEndDate
property contractor

Class SupportRequest

property id
property status
property issuedBy

- **Object identity**
- **Data type transformations**
- **Encoding of values**
- **Generation of object references**
- **Generation of inverse references**
- **N:M relations**



Class „File151“
Class „File 87“

Customer Information Gateway - Queries

**For given SupportRequest
search for the
corresponding Contract**

Class Customer

property id
property name
property address

Class SupportRequest

property id
property status
property issuedBy

Class Contract

property contractId
property contractEndDate
property contractor



Customer Information Gateway

For given Customer
search for all their
SupportRequests

Class Customer

property id
property name
property address

Class Contract

property contractId
property contractEnd
property contractor

Class SupportRequest

property id
property status
property issuedBy



Customer Information Gateway – Query Performance

- **Avoid full scans on source data**
- **Exploit data sources' query capabilities**
 - **filter predicates (more than one)**
 - **joins**
 - **indexes**
 - **uniqueness**

Customer Information Gateway – Summary

- First (simple) use case
- Different data sources
- Mappings and queries
- Beside data integration not many additional rules
- Performance

- Continue work in context of EU projects
 - ◆ NEON
 - ◆ Semantic Gov

**Let's pump information
we have never seen before
from the data!**

