MANAGING RISK

ISO 15926 Templates

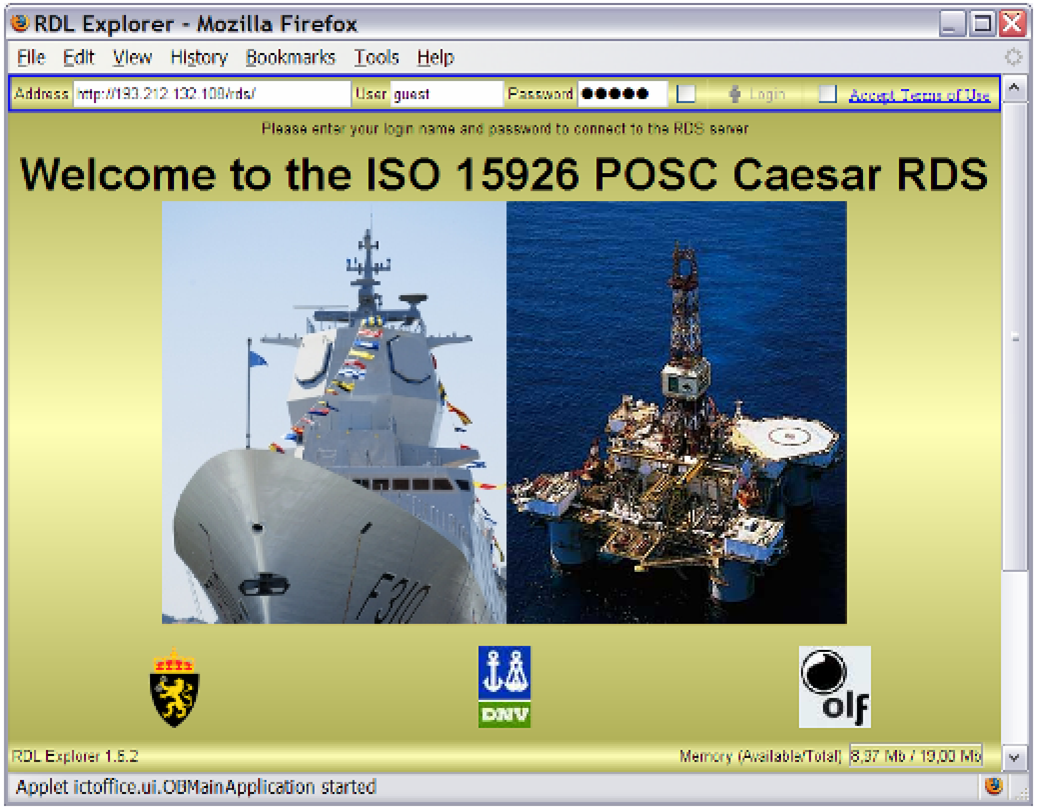
Building a rich ontology on the basis of ISO 15926 Part 2

Johan W. Klüwer

ISO 15926 and Semantic Web technologies, Sogndal , September 12, 2008

Current practice and tools I: *RDE*

- Reference Data Editor
- Compliant with ISO 15926-2
- Available at **rds.posccaesar.com**


RDL Explorer - Mozilla Firefox
File Edit View History Bookmarks Tools Help
Login
Accept Terms of Use
DNV
olf

Address http://193.212.132.108/rds/ User guest Password ●●●●● □

Please enter your login name and password to connect to the RDS server

Welcome to the ISO 15926 POSC Caesar RDS

RDL Explorer 1.5.2 Memory (Available/Total) 8,37 Mb / 19,00 Mb

Applet ictoffice.ui.OBMainApplication started

- A tool for ISO 15926 experts, not so much for Oil & Gas domain experts

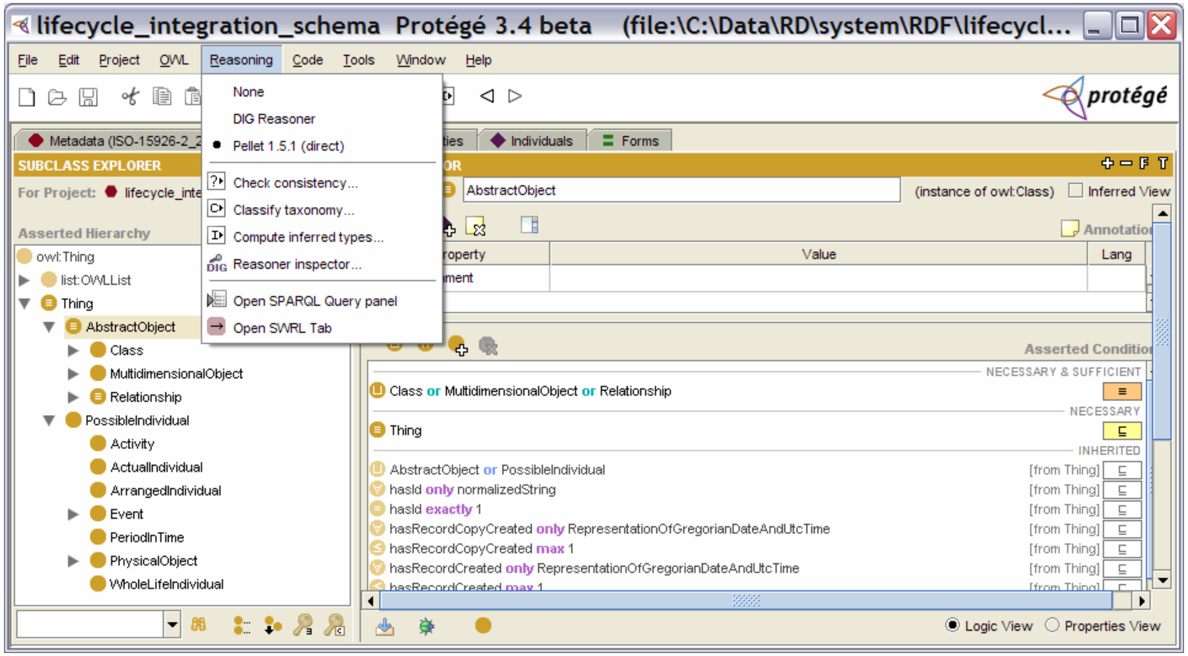
Current practice and tools II: *MS Excel*

- Spreadsheets are still in wide use for defining reference data
- Advantage: Domain experts tend to like tables
- Disadvantage: Complete lack of support for building correct structures

- Creating a good ontology using spreadsheets is practically impossible

Current practice and tools III: *Protégé*

- A generic ontology editor
- Supports loading ISO 15926 Part 2 in OWL format
- Supports loading reference data in OWL format



Class **or** MultidimensionalObject **or** Relationship

Thing

AbstractObject **or** PossibleIndividual [from Thing]

hasId **only** normalizedString [from Thing]

hasId **exactly** 1 [from Thing]

hasRecordCopyCreated **only** RepresentationOfGregorianDateAndUtcTime [from Thing]

hasRecordCopyCreated **max** 1 [from Thing]

hasRecordCreated **only** RepresentationOfGregorianDateAndUtcTime [from Thing]

hasRecordCreated **max** 1 [from Thing]

- Like the RD Editor, an expert tool

Wanted: A simple, compliant interface

- Standardized modelling practice
- Tools that are familiar to domain experts
  - Protégé, RDE are out
- Tools that support the user and check correctness
  - Excel is out
- Tools that provide for working at a suitable level of abstraction
  - Most ontology editors expose the user to too much ”assembly code”

Building domain ontologies with *templates*

- An ontology is used to record *statements.* That’s semantics.
- To build the RDL, we need to represent facts about a given domain using the language of ISO 15926
- Ideally, a domain expert states the facts, and the machine interprets the facts automatically

A template is a pattern for stating facts

- A Template for ISO 15926 is a *predicate*, a *statement form*, a *pattern for facts*
- A template has a *signature* defining the form of a statement
  - What arguments need to be given
  - What are their types
- Each template has an *interpretation rule* that interprets facts that fit the pattern
  - Reducing a complex statement into simpler ones
  - Eventually, to atomic statements in ISO 15926
  - Yielding an expression of the fact in the ontology language
- Current prototype developed in the Intelligent Data Sets (IDS) project

Template example I

- Constraint: A car has 3 or more wheels
- Express the constraint with a suitable template

Parts-at-least (Car, Wheel, 3) *!*

- Rules generate a set of ISO 15926 statements

“ The statement

Parts-at-least (C, D, i)

means that

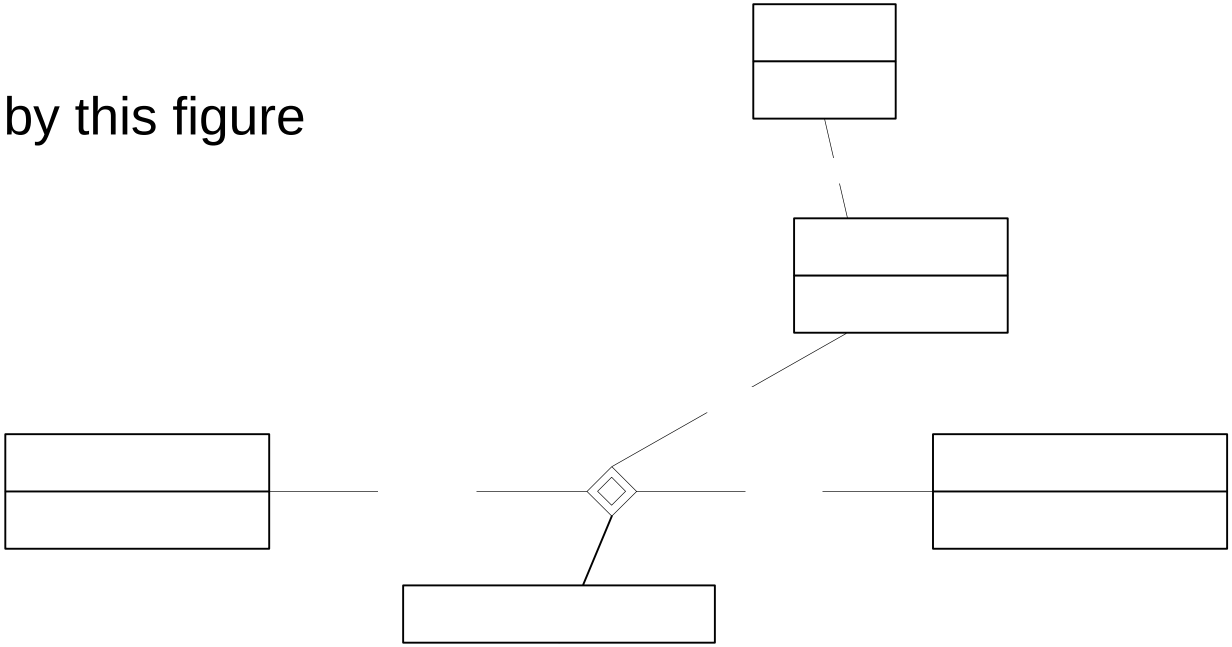
Any *C* has at least *i D’s* as parts ”

Template example I

- Assume that any car has 3 or more wheels
- Expressed with a suitable template *Parts*

Parts-at-least (Car, Wheel, 3) *!*

- Rules generate a set of ISO 15926 statements
- … as illustrated by this figure



A familiar interface

- Making the statement

  Parts-at-least (Car, Wheel, 3)

  requires no detailed knowledge about modelling

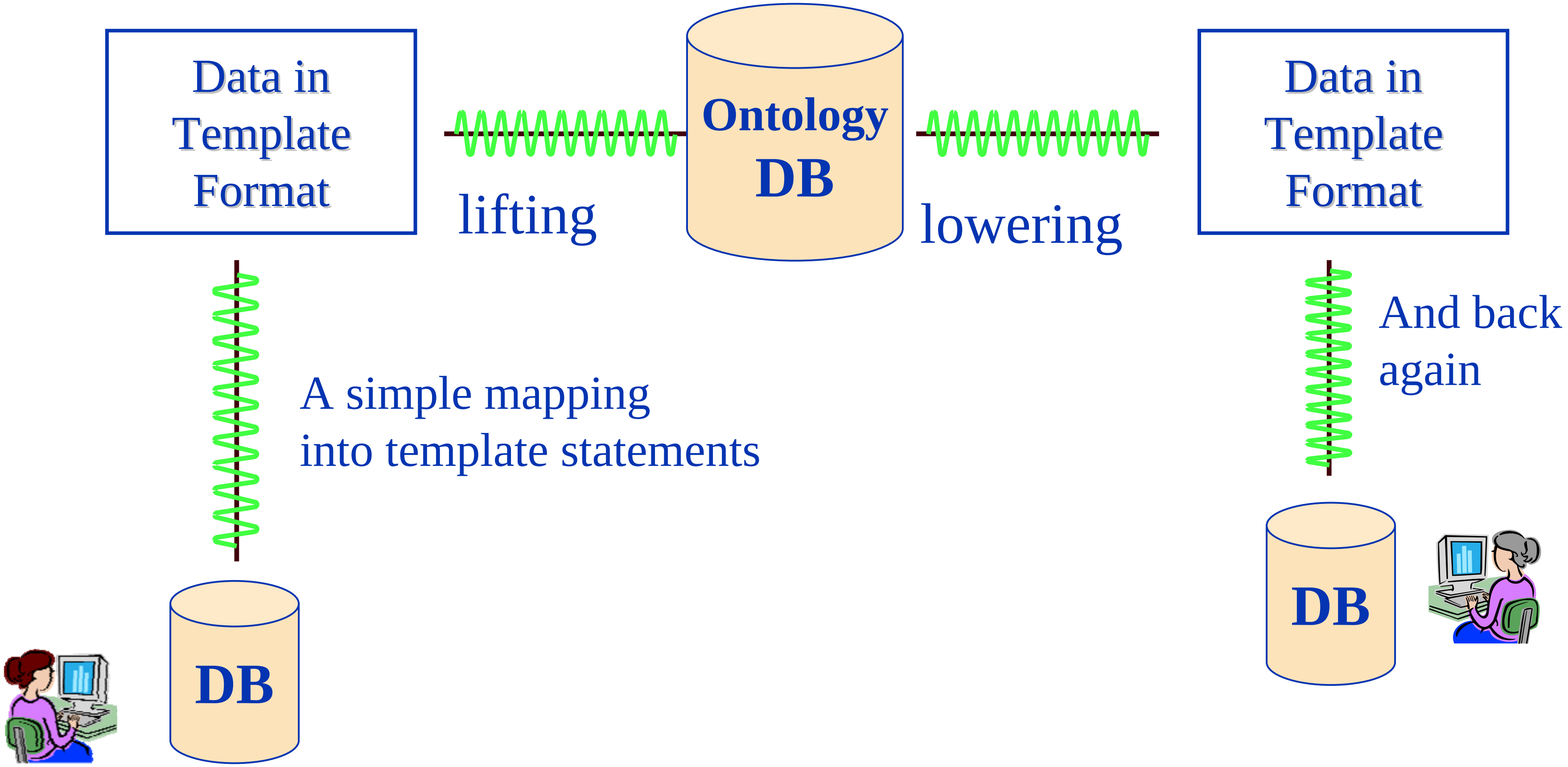
- A list of arguments can easily be stored in a table (Excel!)

Car	Wheel	3
Bicycle	Wheel	1

- Correctness of the generated ontology structure can be checked using generic ontology tools

And this can be used for ...

Translation by means of templates



Data in Template Format

lifting

Ontology DB

lowering

Data in Template Format

A simple mapping into template statements

And back again

DB

DB

Template example II

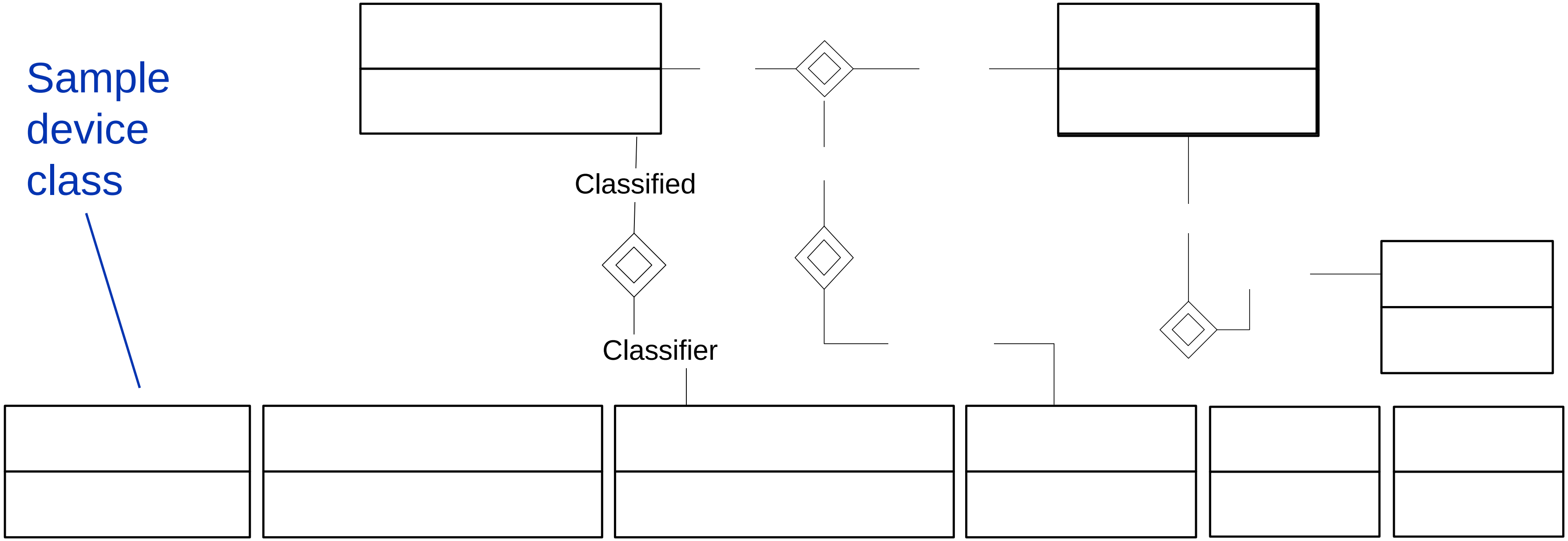
- A fairly complex claim (from IDS prototype)

  “The ambient temperature during operation of a 3051CG pressure transmitter should be within -40 and 85 degrees Celsius.”

- Six arguments are required for a precise statement

  “The *body height* of a *human* is a *length* property which varies from *50* to *250* *cm*”

Property with Scale and Quantification

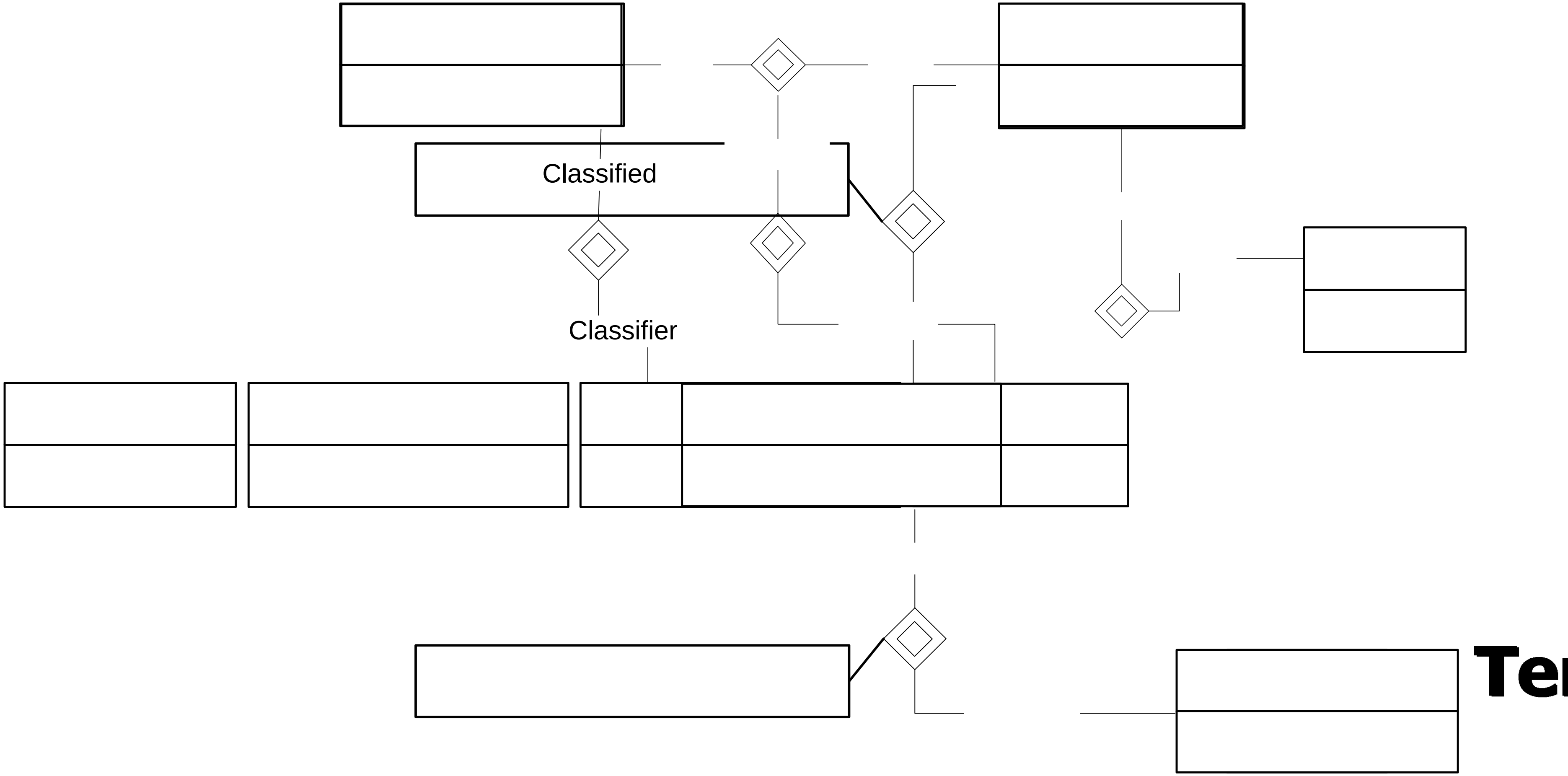


Sample device class

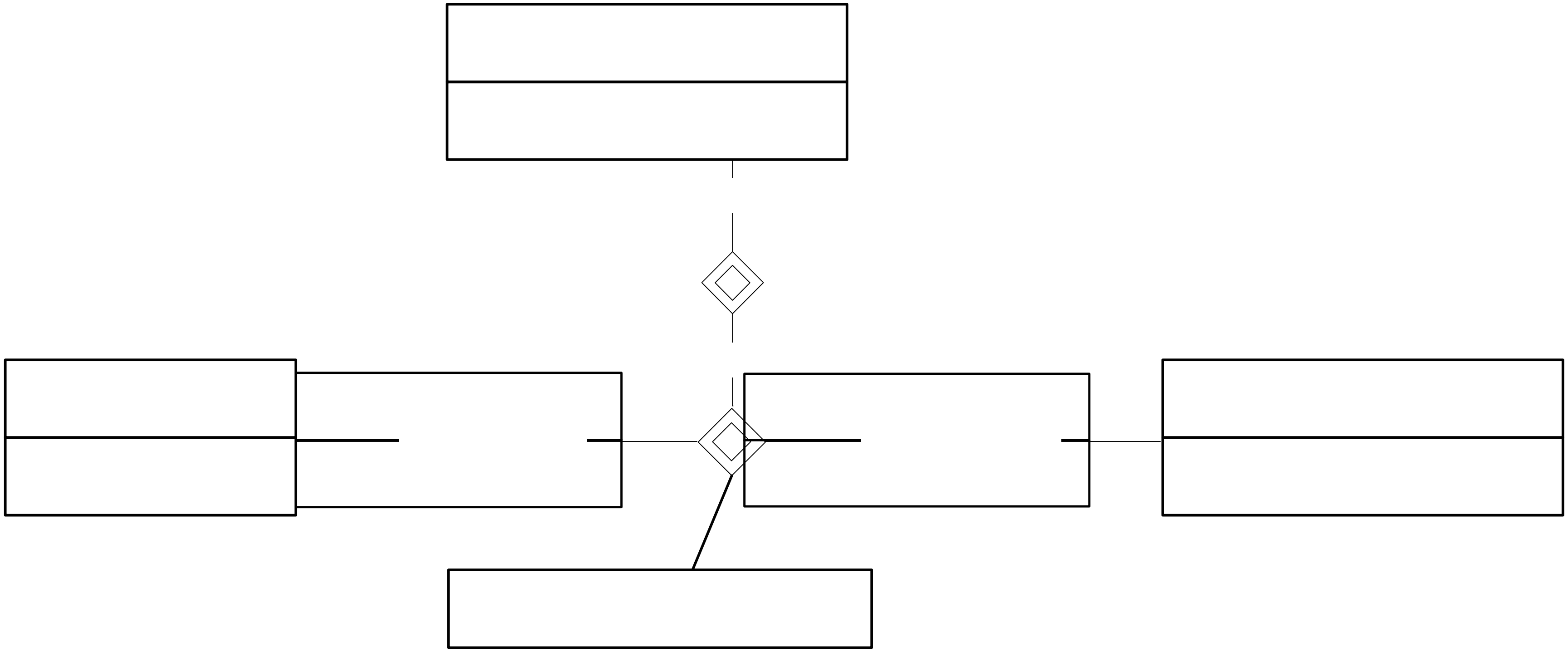
“The ambient temperature during operation of a 3051CG pressure transmitter should be within -40 and 85 degrees Celsius.”

Te

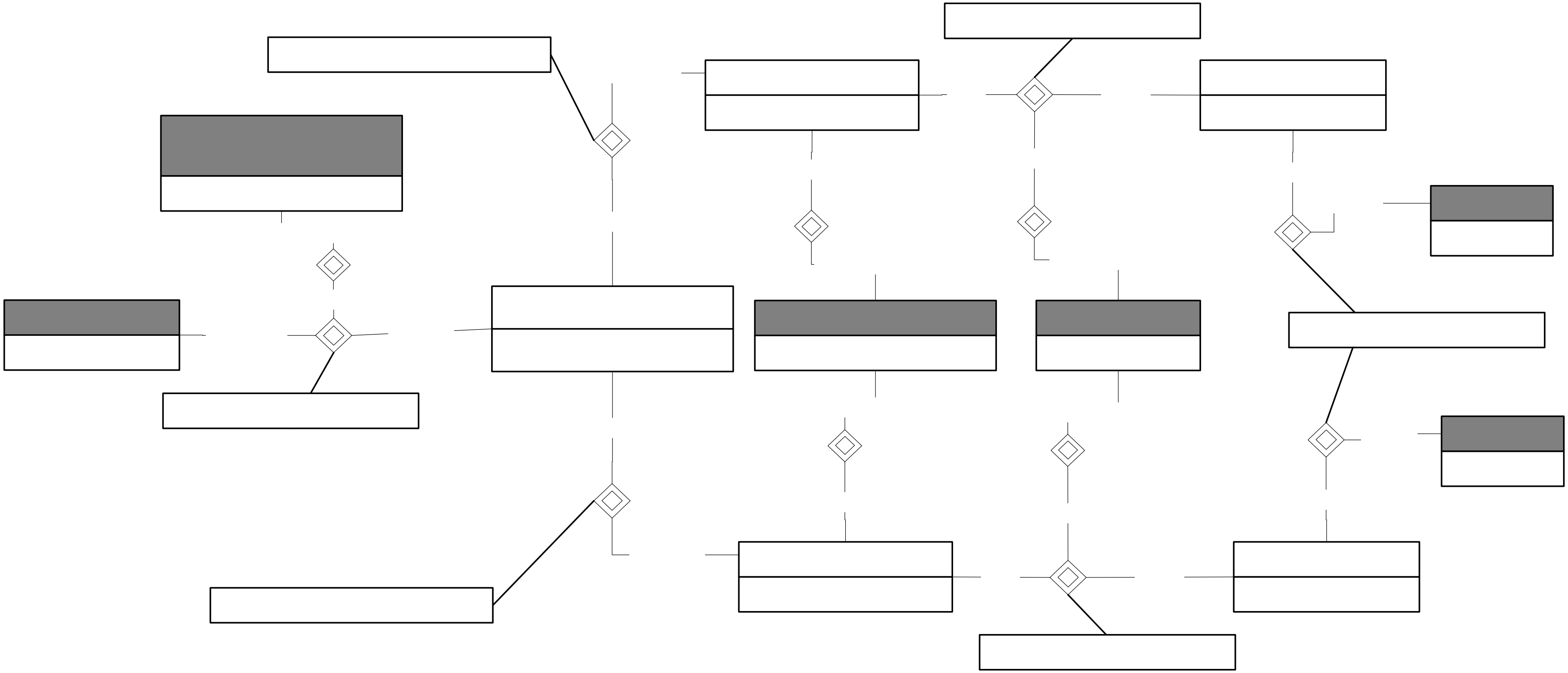
Property Range



Property Range Restriction



Model: Ambient Temperature Range



3051CG ambient temperature: -40°C – 85°C

Upper Bound Of Pr

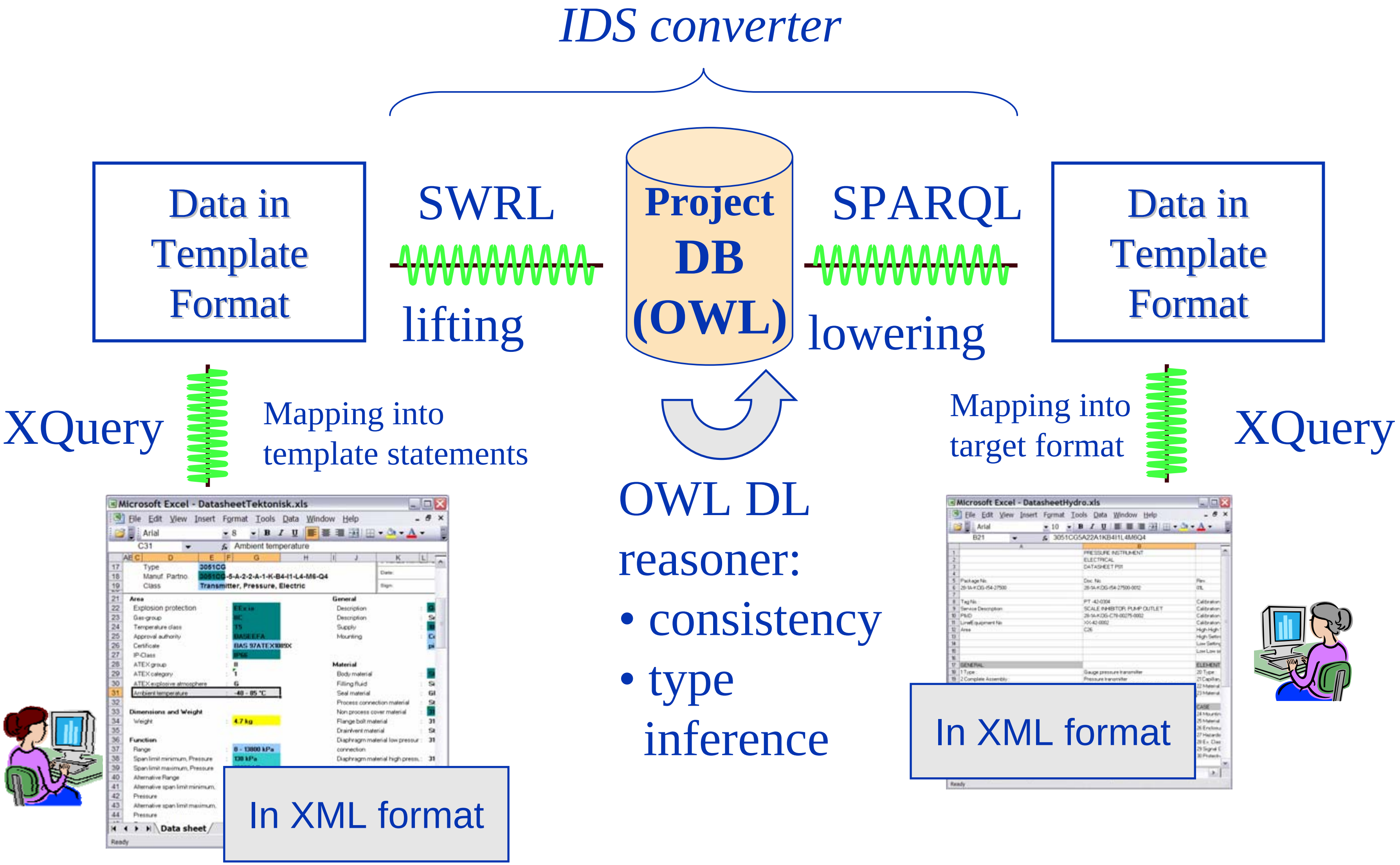
Templates for ontology development

- A flexible and precise language for ontology building
- Let the compiler handle the “assembly language”
- Creating rich semantic structure becomes practical
- Standardization of templates makes standardized modelling patterns possible
- Results can be consistency checked using automated reasoning*

**i.e., experimental verification*

And we have even tried it out ...

Translation in practice



IDS converter

Data in Template Format

SWRL

lifting

Project DB (OWL)

SPARQL

lowering

Data in Template Format

XQuery

Mapping into template statements

OWL DL reasoner:

- consistency
- type inference

Mapping into target format

XQuery

Microsoft Excel - DatasheetTektonisk.xls
File Edit View Insert Format Tools Data Window Help
Arial
8
C31
Ambient temperature
Data sheet
Ready

17	Type	3051CG
18	Manuf. Partno.	3051CG-5-A-2-2-A-1-K-B4-I1-L4-M6-Q4
19	Class	Transmitter, Pressure, Electric
21	**Area**	
22	Explosion protection	EEx ia
23	Gas-group	IIC
24	Temperature class	T5
25	Approval authority	BASEEFA
26	Certificate	BAS 97ATEX1089X
27	IP-Class	IP66
28	ATEX group	II
29	ATEX category	1
30	ATEX explosive atmosphere	G
31	Ambient temperature	-40 - 85 °C
33	**Dimensions and Weight**	
34	Weight	4.7 kg
36	**Function**	
37	Range	0 - 13800 kPa
38	Span limit minimum, Pressure	138 kPa
39	Span limit maximum, Pressure	
40	Alternative Range	
41	Alternative span limit minimum,	
42	Pressure	
43	Alternative span limit maximum,	
44	Pressure	

General
Description
Description
Supply
Mounting

Material
Body material
Filling fluid
Seal material
Process connection material
Non process cover material
Flange bolt material
Drain/vent material
Diaphragm material low pressur connection
Diaphragm material high press.

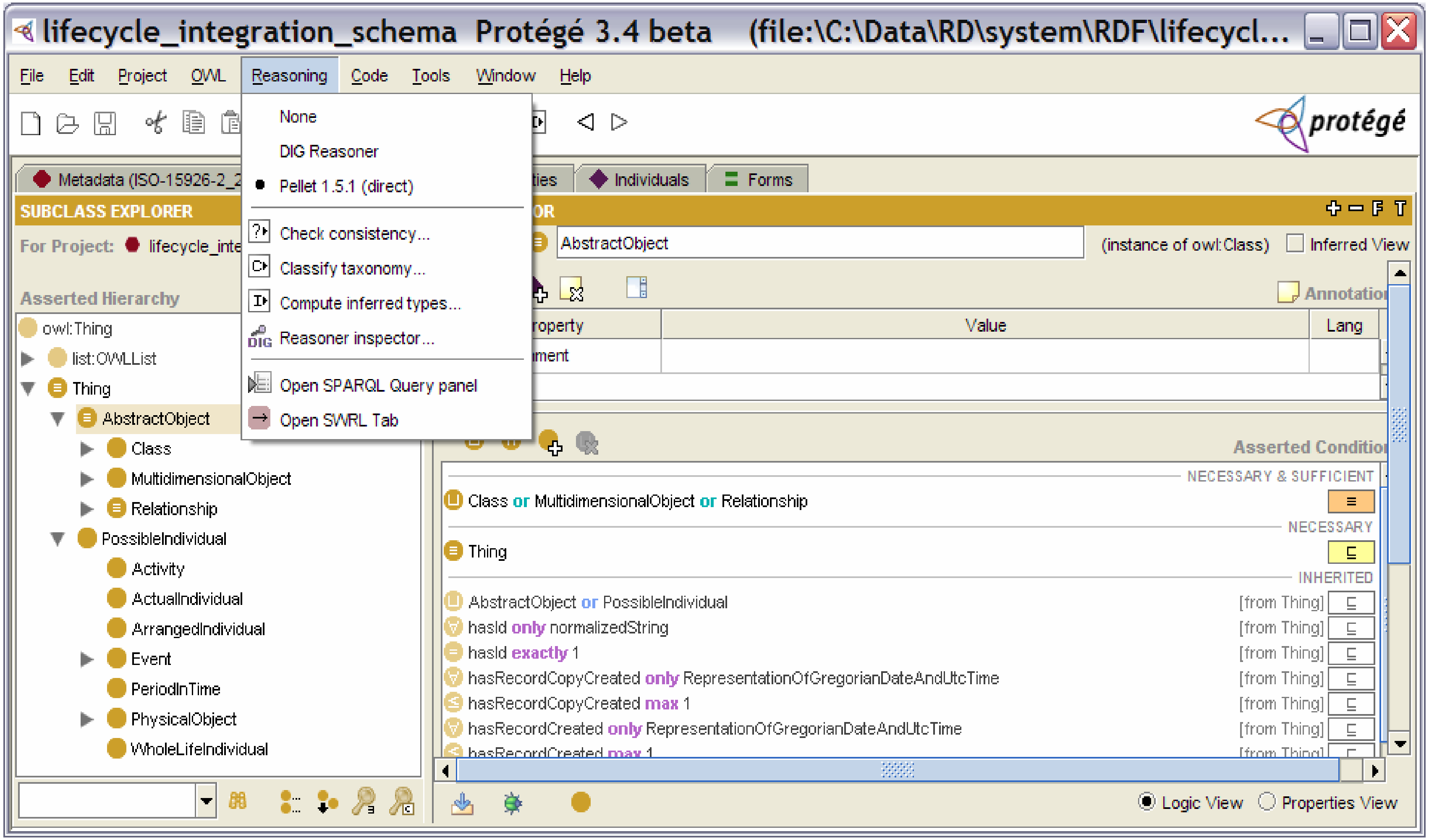
In XML format

Microsoft Excel - DatasheetHydro.xls
File Edit View Insert Format Tools Data Window Help
Arial
10
B21
3051CG5A22A1KB4I1L4M6Q4
Ready

	A	B
1		PRESSURE INSTRUMENT
2		ELECTRICAL
3		DATASHEET PSI
5	Package No	Doc. No
6	29-1A-KOG-I54-27500	29-1A-KOG-I54-27500-0002
8	Tag No.	PT -42-0304
9	Service Description	SCALE INHIBITOR, PUMP OUTLET
10	P&ID	29-1A-KOG-C78-00275-0002
11	Line/Equipment No	XX-42-0002
12	Area	C26
17	GENERAL	
18	1 Type :	Gauge pressure transmitter
19	2 Complete Assembly :	Pressure transmitter

In XML format

Our Protégé-OWL laboratory



For Project: lifecycle_inte

- owl:Thing
  - list:OWLList
  - Thing
    - AbstractObject
      - Class
      - MultidimensionalObject
      - Relationship
    - PossibleIndividual
      - Activity
      - ActualIndividual
      - ArrangedIndividual
      - Event
      - PeriodInTime
      - PhysicalObject
      - WholeLifeIndividual

AbstractObject (instance of owl:Class) □ Inferred View

roperty	Value	Lang
ment		

NECESSARY & SUFFICIENT

Class or MultidimensionalObject or Relationship ≡

NECESSARY

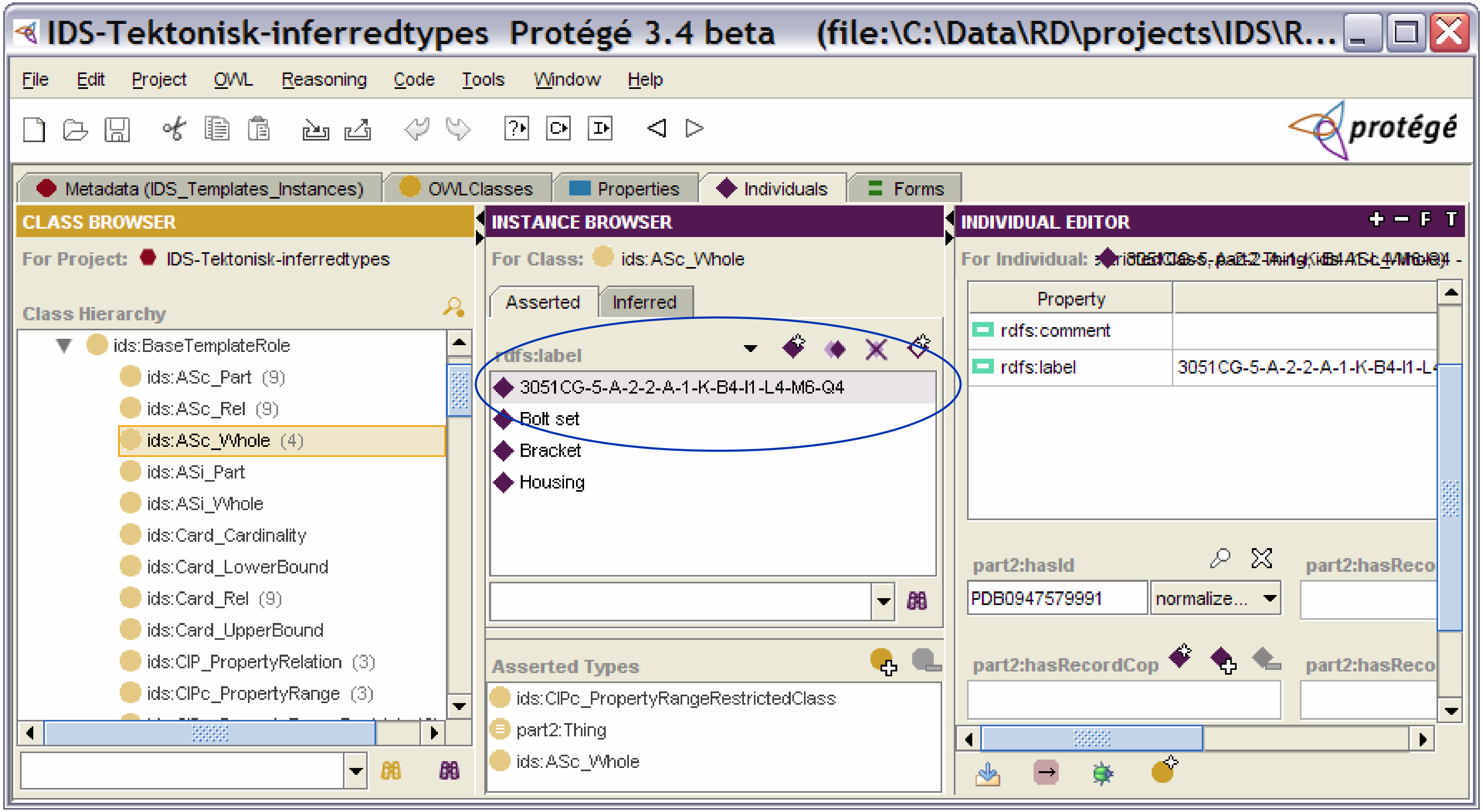
Thing ⊑

INHERITED

AbstractObject or PossibleIndividual	[from Thing]	⊑
hasId only normalizedString	[from Thing]	⊑
hasId exactly 1	[from Thing]	⊑
hasRecordCopyCreated only RepresentationOfGregorianDateAndUtcTime	[from Thing]	⊑
hasRecordCopyCreated max 1	[from Thing]	⊑
hasRecordCreated only RepresentationOfGregorianDateAndUtcTime	[from Thing]	⊑
hasRecordCreated max 1	[from Thing]	⊑

● Logic View ○ Properties View

Transmitter 3051CG


IDS-Tektonisk-inferredtypes Protégé 3.4 beta (file:\C:\Data\RD\projects\IDS\R...
File Edit Project OWL Reasoning Code Tools Window Help
protégé
Metadata (IDS_Templates_Instances)
OWLClasses
Properties
Individuals
Forms
CLASS BROWSER
INSTANCE BROWSER
INDIVIDUAL EDITOR
Asserted
Inferred

For Project: IDS-Tektonisk-inferredtypes

Class Hierarchy

- ids:BaseTemplateRole
  - ids:ASc_Part (9)
  - ids:ASc_Rel (9)
  - ids:ASc_Whole (4)
  - ids:ASi_Part
  - ids:ASi_Whole
  - ids:Card_Cardinality
  - ids:Card_LowerBound
  - ids:Card_Rel (9)
  - ids:Card_UpperBound
  - ids:ClP_PropertyRelation (3)
  - ids:ClPc_PropertyRange (3)

For Class: ids:ASc_Whole

rdfs:label

- 3051CG-5-A-2-2-A-1-K-B4-I1-L4-M6-Q4
- Bolt set
- Bracket
- Housing

Asserted Types

- ids:ClPc_PropertyRangeRestrictedClass
- part2:Thing
- ids:ASc_Whole

For Individual: [illegible]

Property	
rdfs:comment	
rdfs:label	3051CG-5-A-2-2-A-1-K-B4-I1-L4

part2:hasId

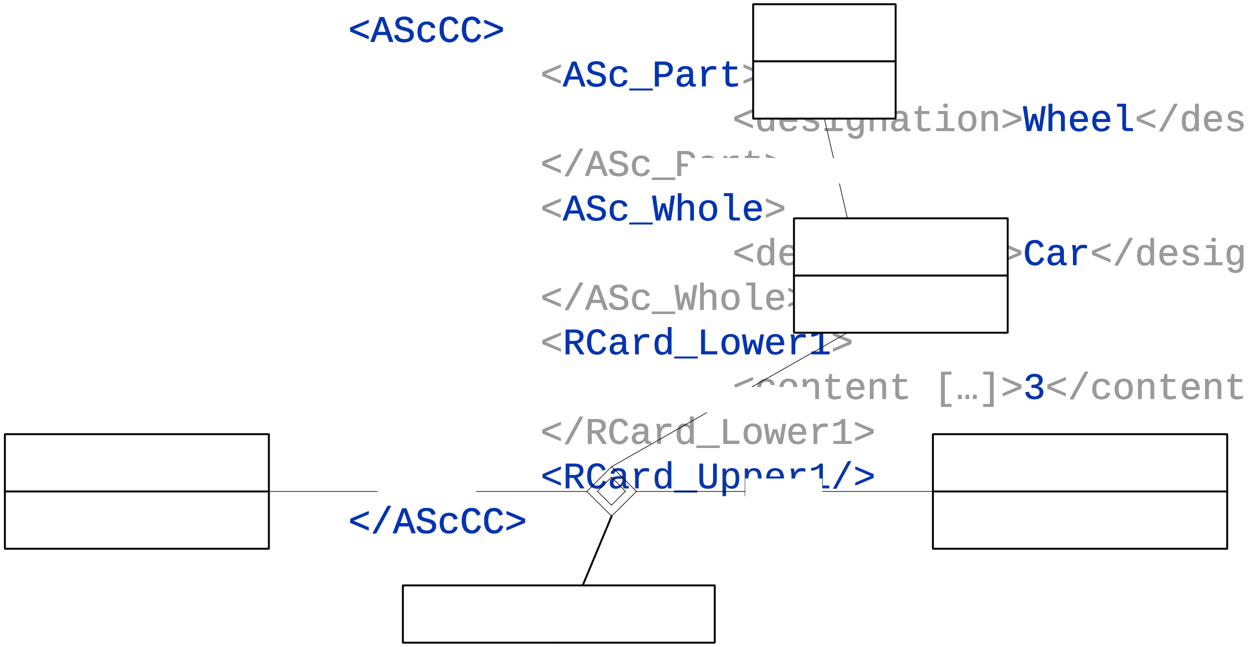
PDB0947579991 normalize...

part2:hasReco

part2:hasRecordCop

part2:hasReco

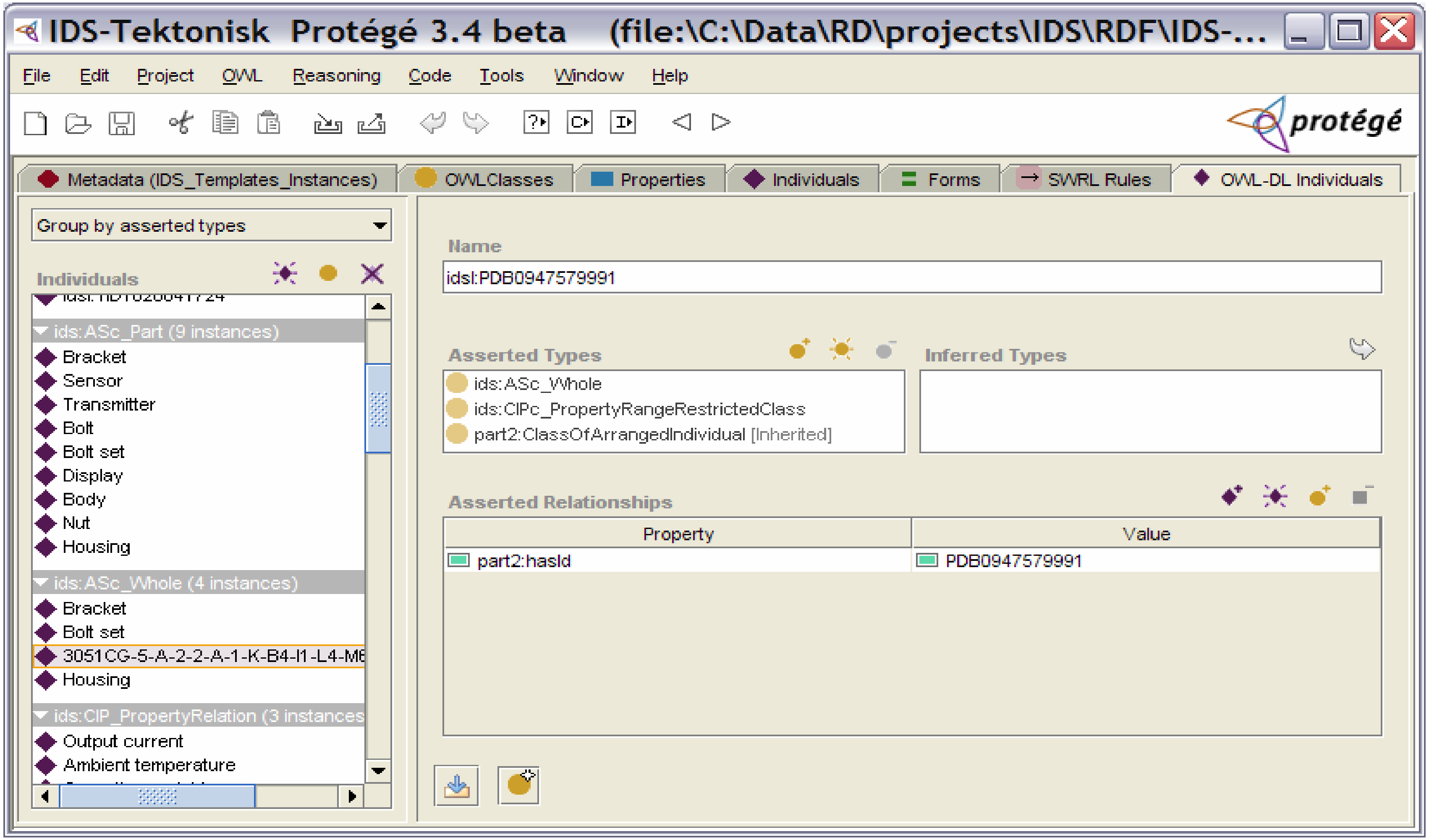
Input in a straightforward XML format



```
<AScCC>
        <ASc_Part>
                <designation>Wheel</designation>
        </ASc_Part>
        <ASc_Whole>
                <designation>Car</designation>
        </ASc_Whole>
        <RCard_Lower1>
                <content […]>3</content>
        </RCard_Lower1>
        <RCard_Upper1/>
</AScCC>
```

A car has at least three wheels.

Input data in Protege


IDS-Tektonisk Protégé 3.4 beta (file:\C:\Data\RD\projects\IDS\RDF\IDS-...
File Edit Project OWL Reasoning Code Tools Window Help
protégé
Metadata (IDS_Templates_Instances)
OWLClasses
Properties
Individuals
Forms
SWRL Rules
OWL-DL Individuals

Group by asserted types

Individuals

- ids:ASc_Part (9 instances)
  - Bracket
  - Sensor
  - Transmitter
  - Bolt
  - Bolt set
  - Display
  - Body
  - Nut
  - Housing
- ids:ASc_Whole (4 instances)
  - Bracket
  - Bolt set
  - 3051CG-5-A-2-2-A-1-K-B4-I1-L4-M6
  - Housing
- ids:ClP_PropertyRelation (3 instances
  - Output current
  - Ambient temperature

Name: idsl:PDB0947579991

Asserted Types

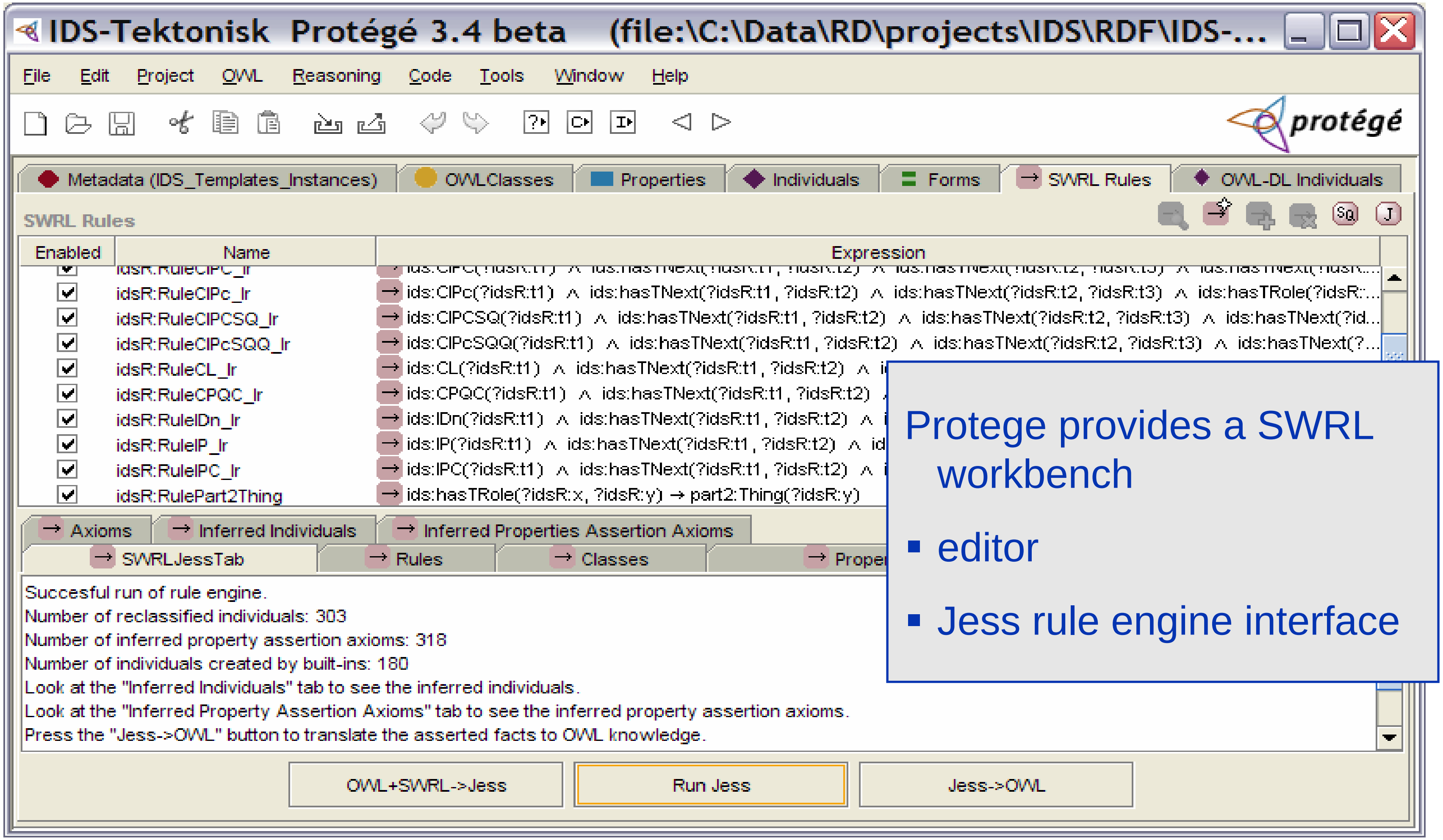
- ids:ASc_Whole
- ids:ClPc_PropertyRangeRestrictedClass
- part2:ClassOfArrangedIndividual [Inherited]

Inferred Types

Asserted Relationships

Property	Value
part2:hasId	PDB0947579991

Executing template rules



IDS-Tektonisk Protégé 3.4 beta (file:\C:\Data\RD\projects\IDS\RDF\IDS-...

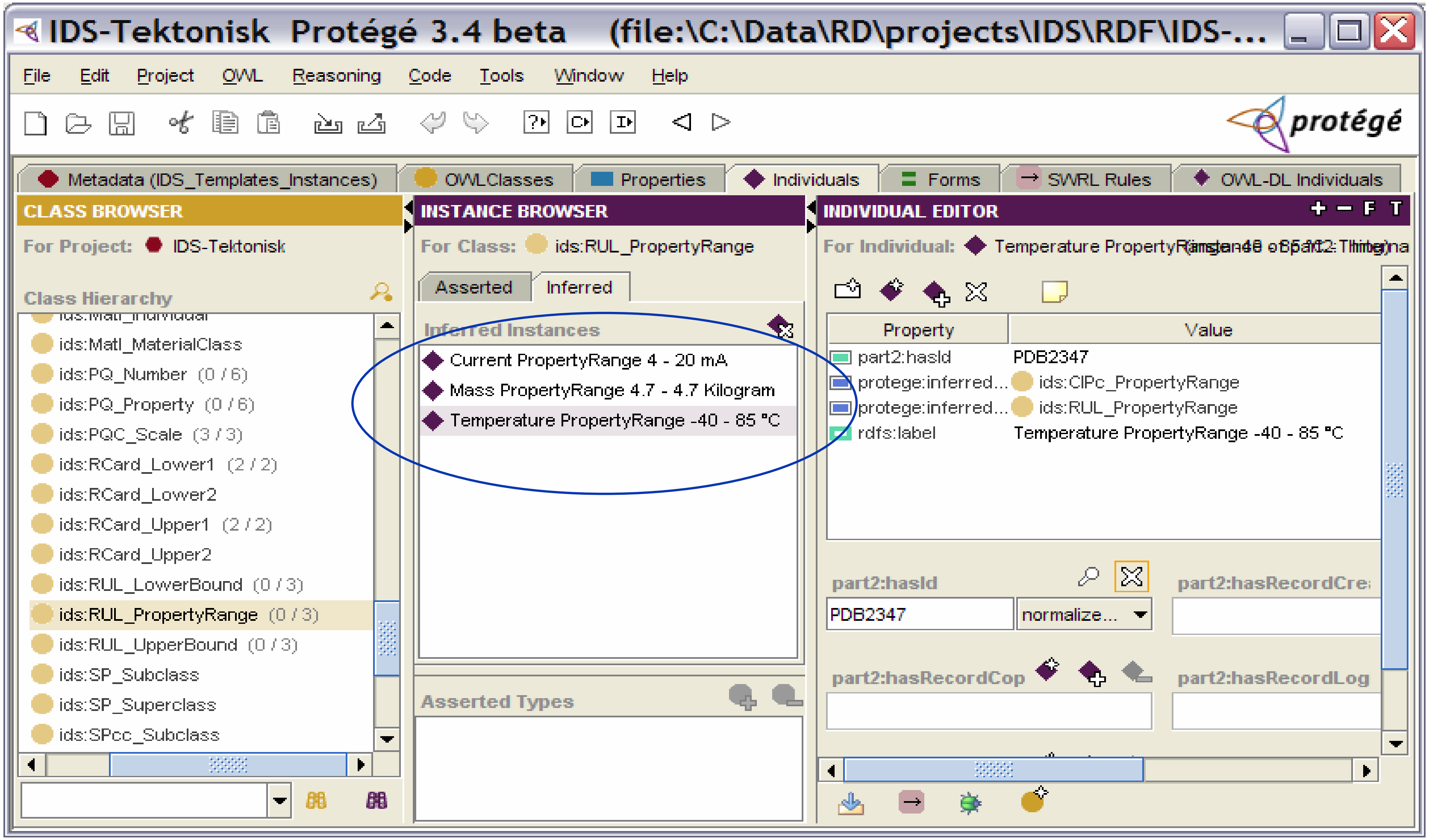
Enabled	Name	Expression
☑	idsR:RuleClPc_lr	ids:ClPc(?idsR:t1) ∧ ids:hasTNext(?idsR:t1, ?idsR:t2) ∧ ids:hasTNext(?idsR:t2, ?idsR:t3) ∧ ids:hasTRole(?idsR:...
☑	idsR:RuleClPCSQ_lr	ids:ClPCSQ(?idsR:t1) ∧ ids:hasTNext(?idsR:t1, ?idsR:t2) ∧ ids:hasTNext(?idsR:t2, ?idsR:t3) ∧ ids:hasTNext(?id...
☑	idsR:RuleClPcSQQ_lr	ids:ClPcSQQ(?idsR:t1) ∧ ids:hasTNext(?idsR:t1, ?idsR:t2) ∧ ids:hasTNext(?idsR:t2, ?idsR:t3) ∧ ids:hasTNext(?...
☑	idsR:RuleCL_lr	ids:CL(?idsR:t1) ∧ ids:hasTNext(?idsR:t1, ?idsR:t2) ∧ i
☑	idsR:RuleCPQC_lr	ids:CPQC(?idsR:t1) ∧ ids:hasTNext(?idsR:t1, ?idsR:t2) ∧
☑	idsR:RuleIDn_lr	ids:IDn(?idsR:t1) ∧ ids:hasTNext(?idsR:t1, ?idsR:t2) ∧ i
☑	idsR:RuleIP_lr	ids:IP(?idsR:t1) ∧ ids:hasTNext(?idsR:t1, ?idsR:t2) ∧ id
☑	idsR:RuleIPC_lr	ids:IPC(?idsR:t1) ∧ ids:hasTNext(?idsR:t1, ?idsR:t2) ∧ i
☑	idsR:RulePart2Thing	ids:hasTRole(?idsR:x, ?idsR:y) → part2:Thing(?idsR:y)

```
Succesful run of rule engine.
Number of reclassified individuals: 303
Number of inferred property assertion axioms: 318
Number of individuals created by built-ins: 180
Look at the "Inferred Individuals" tab to see the inferred individuals.
Look at the "Inferred Property Assertion Axioms" tab to see the inferred property assertion axioms.
Press the "Jess->OWL" button to translate the asserted facts to OWL knowledge.
```

Protege provides a SWRL workbench

- editor
- Jess rule engine interface

Individuals from rules and inference



For Project: IDS-Tektonisk

- ids:Matl_Individual
- ids:Matl_MaterialClass
- ids:PQ_Number (0 / 6)
- ids:PQ_Property (0 / 6)
- ids:PQC_Scale (3 / 3)
- ids:RCard_Lower1 (2 / 2)
- ids:RCard_Lower2
- ids:RCard_Upper1 (2 / 2)
- ids:RCard_Upper2
- ids:RUL_LowerBound (0 / 3)
- ids:RUL_PropertyRange (0 / 3)
- ids:RUL_UpperBound (0 / 3)
- ids:SP_Subclass
- ids:SP_Superclass
- ids:SPcc_Subclass

For Class: ids:RUL_PropertyRange

- Current PropertyRange 4 - 20 mA
- Mass PropertyRange 4.7 - 4.7 Kilogram
- Temperature PropertyRange -40 - 85 °C

For Individual: Temperature PropertyRange -40 - 85 °C

Property	Value
part2:hasId	PDB2347
protege:inferred...	ids:CIPc_PropertyRange
protege:inferred...	ids:RUL_PropertyRange
rdfs:label	Temperature PropertyRange -40 - 85 °C

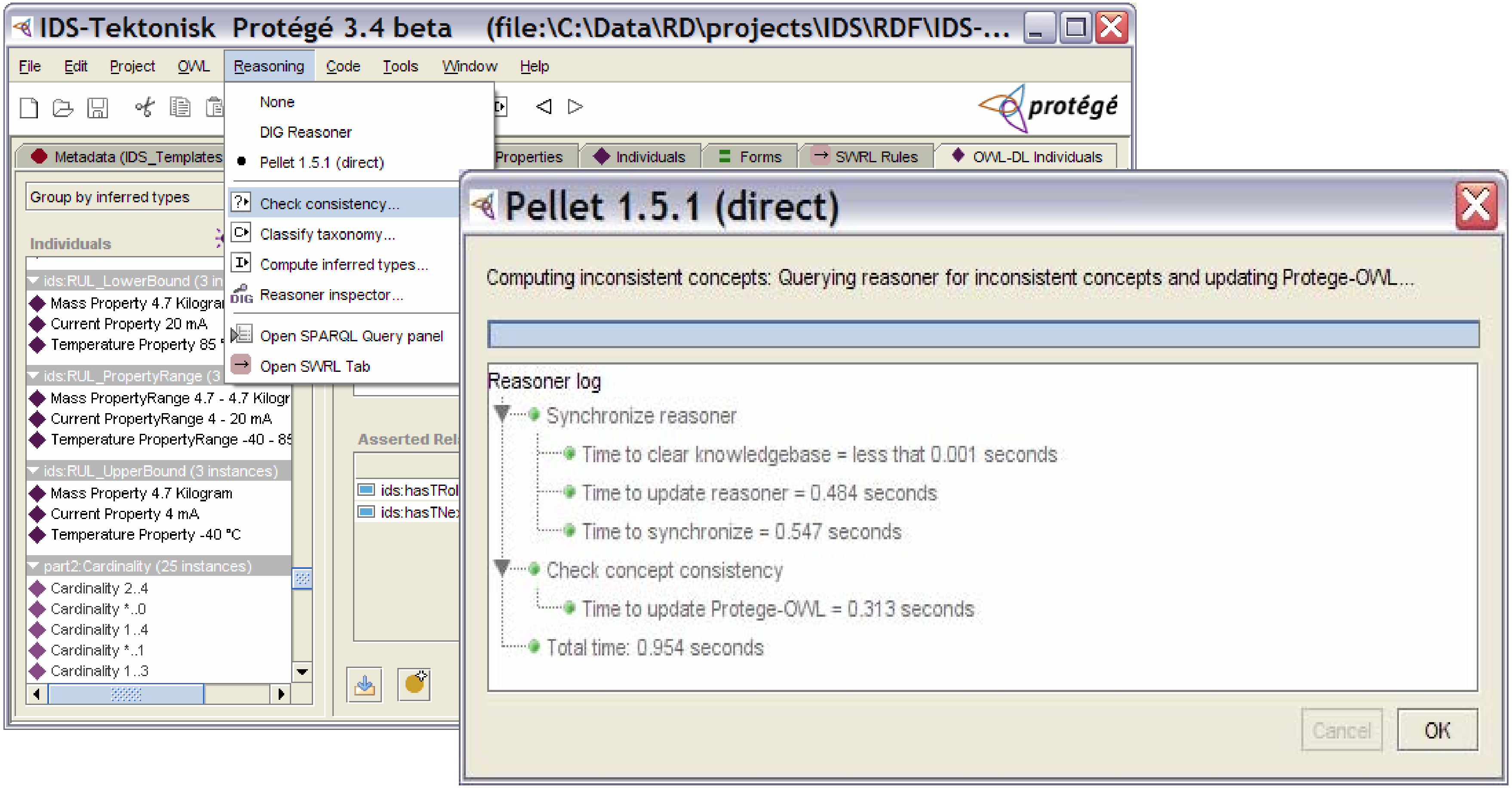
part2:hasId: PDB2347

part2:hasRecordCre

part2:hasRecordCop

part2:hasRecordLog

A consistency check

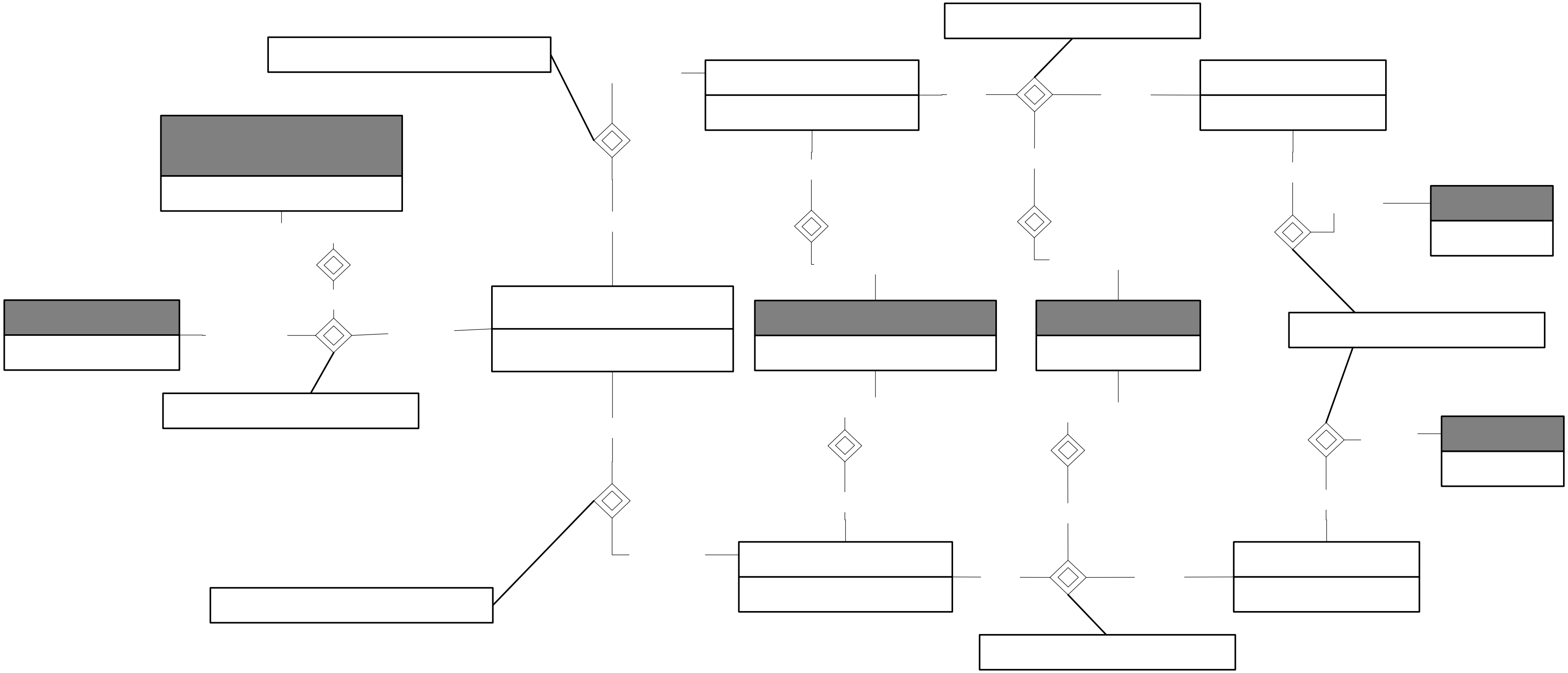


Computing inconsistent concepts: Querying reasoner for inconsistent concepts and updating Protege-OWL...

```
Reasoner log
▼ Synchronize reasoner
    Time to clear knowledgebase = less that 0.001 seconds
    Time to update reasoner = 0.484 seconds
    Time to synchronize = 0.547 seconds
▼ Check concept consistency
    Time to update Protege-OWL = 0.313 seconds
  Total time: 0.954 seconds
```

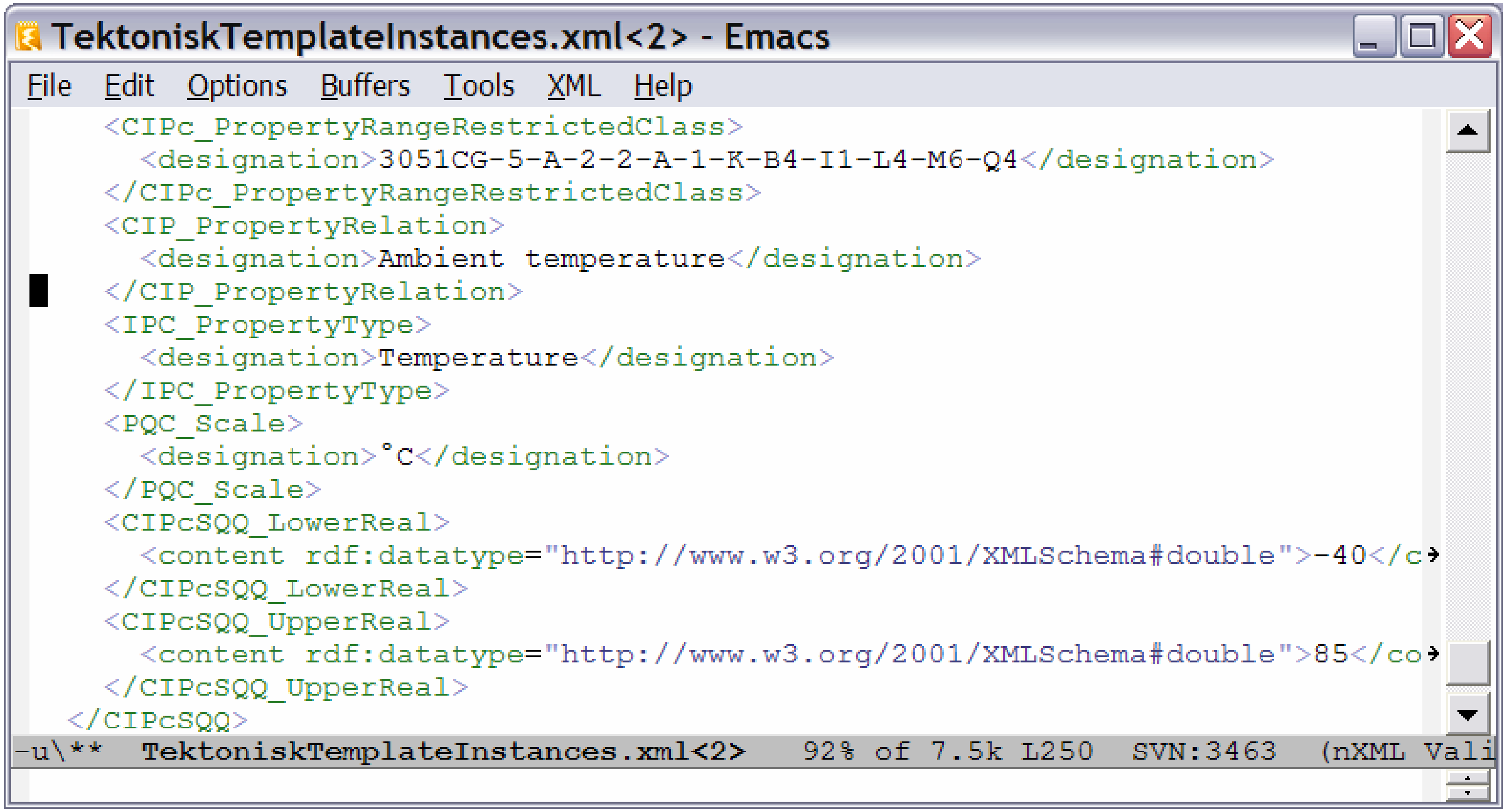
**i.e., experimental verification*

Example: Temperature range, once more



Upper Bound Of Pr

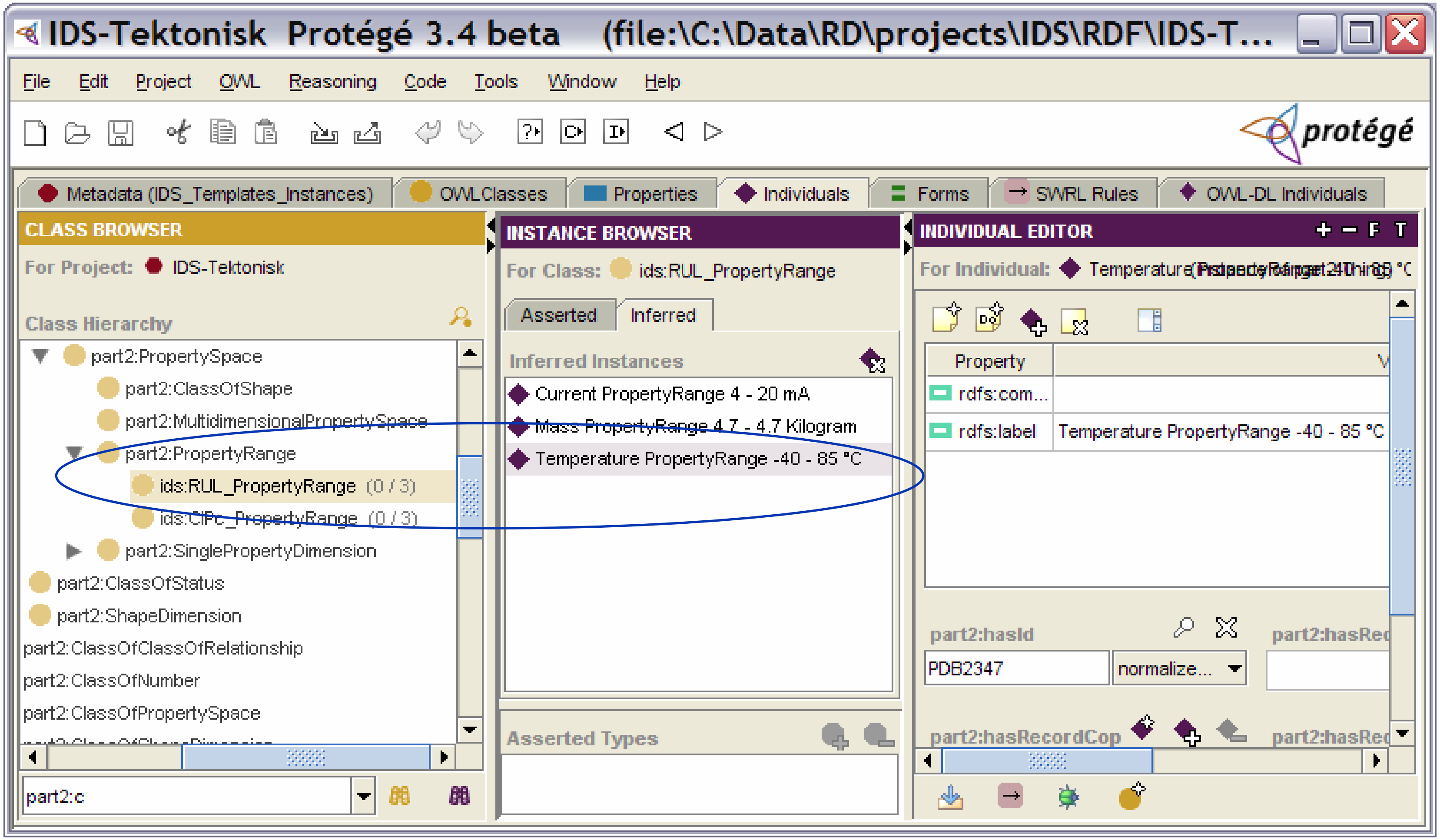
Mapped into an XML template



TektoniskTemplateInstances.xml<2> - Emacs
File Edit Options Buffers Tools XML Help

```
    <CIPc_PropertyRangeRestrictedClass>
      <designation>3051CG-5-A-2-2-A-1-K-B4-I1-L4-M6-Q4</designation>
    </CIPc_PropertyRangeRestrictedClass>
    <CIP_PropertyRelation>
      <designation>Ambient temperature</designation>
    </CIP_PropertyRelation>
    <IPC_PropertyType>
      <designation>Temperature</designation>
    </IPC_PropertyType>
    <PQC_Scale>
      <designation>°C</designation>
    </PQC_Scale>
    <CIPcSQQ_LowerReal>
      <content rdf:datatype="http://www.w3.org/2001/XMLSchema#double">-40</c
    </CIPcSQQ_LowerReal>
    <CIPcSQQ_UpperReal>
      <content rdf:datatype="http://www.w3.org/2001/XMLSchema#double">85</co
    </CIPcSQQ_UpperReal>
  </CIPcSQQ>
-u\**  TektoniskTemplateInstances.xml<2>   92% of 7.5k L250   SVN:3463   (nXML Vali
```

The range as an ISO 15926 Property Range



For Project: IDS-Tektonisk

Class Hierarchy

- part2:PropertySpace
  - part2:ClassOfShape
  - part2:MultidimensionalPropertySpace
  - part2:PropertyRange
    - ids:RUL_PropertyRange (0 / 3)
    - ids:ClPc_PropertyRange (0 / 3)
  - part2:SinglePropertyDimension
- part2:ClassOfStatus
- part2:ShapeDimension
- part2:ClassOfClassOfRelationship
- part2:ClassOfNumber
- part2:ClassOfPropertySpace

part2:c

For Class: ids:RUL_PropertyRange

- Current PropertyRange 4 - 20 mA
- Mass PropertyRange 4.7 - 4.7 Kilogram
- Temperature PropertyRange -40 - 85 °C

For Individual: Temperature PropertyRange -40 - 85 °C

Property	
rdfs:com...	
rdfs:label	Temperature PropertyRange -40 - 85 °C

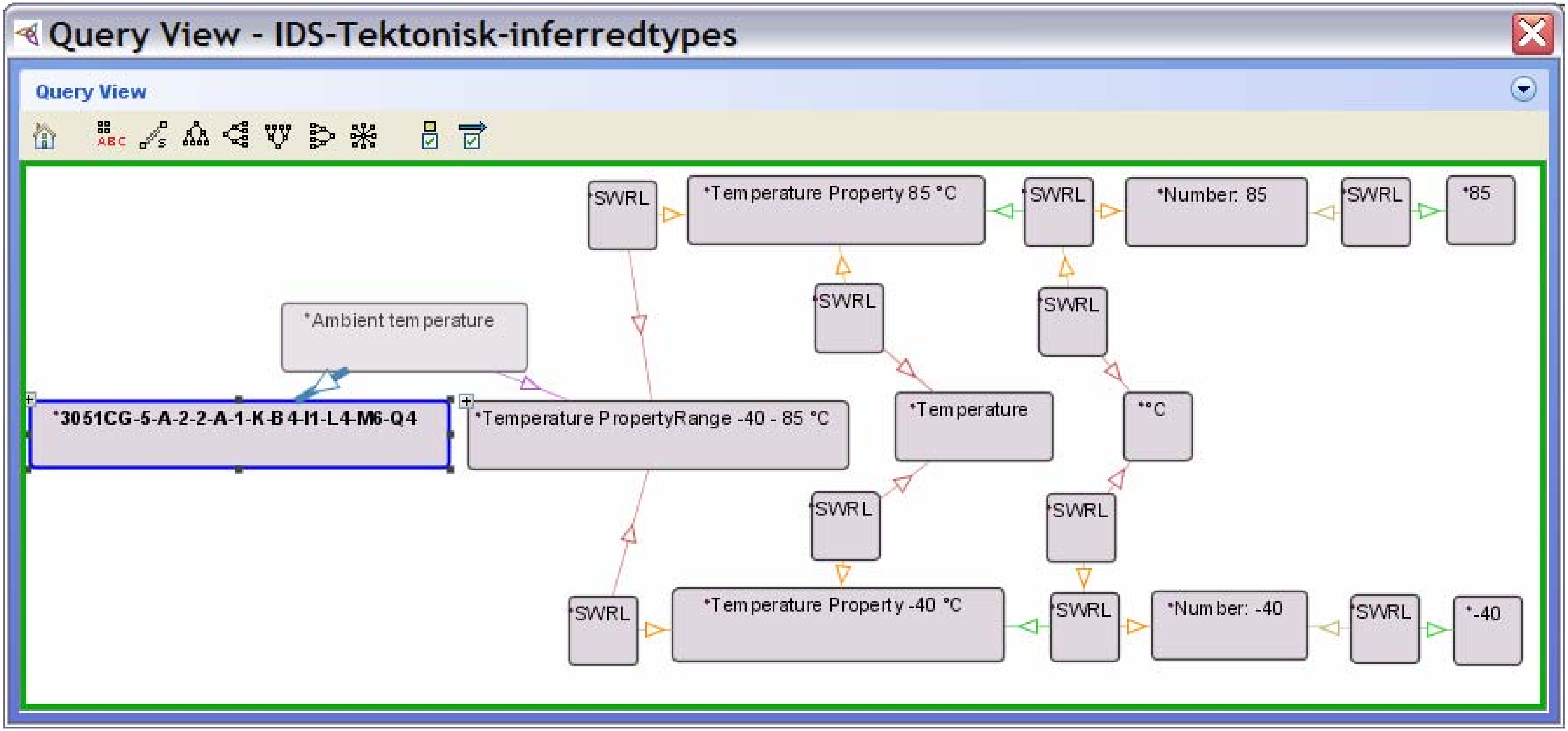
part2:hasId: PDB2347 normalize...

part2:hasRec

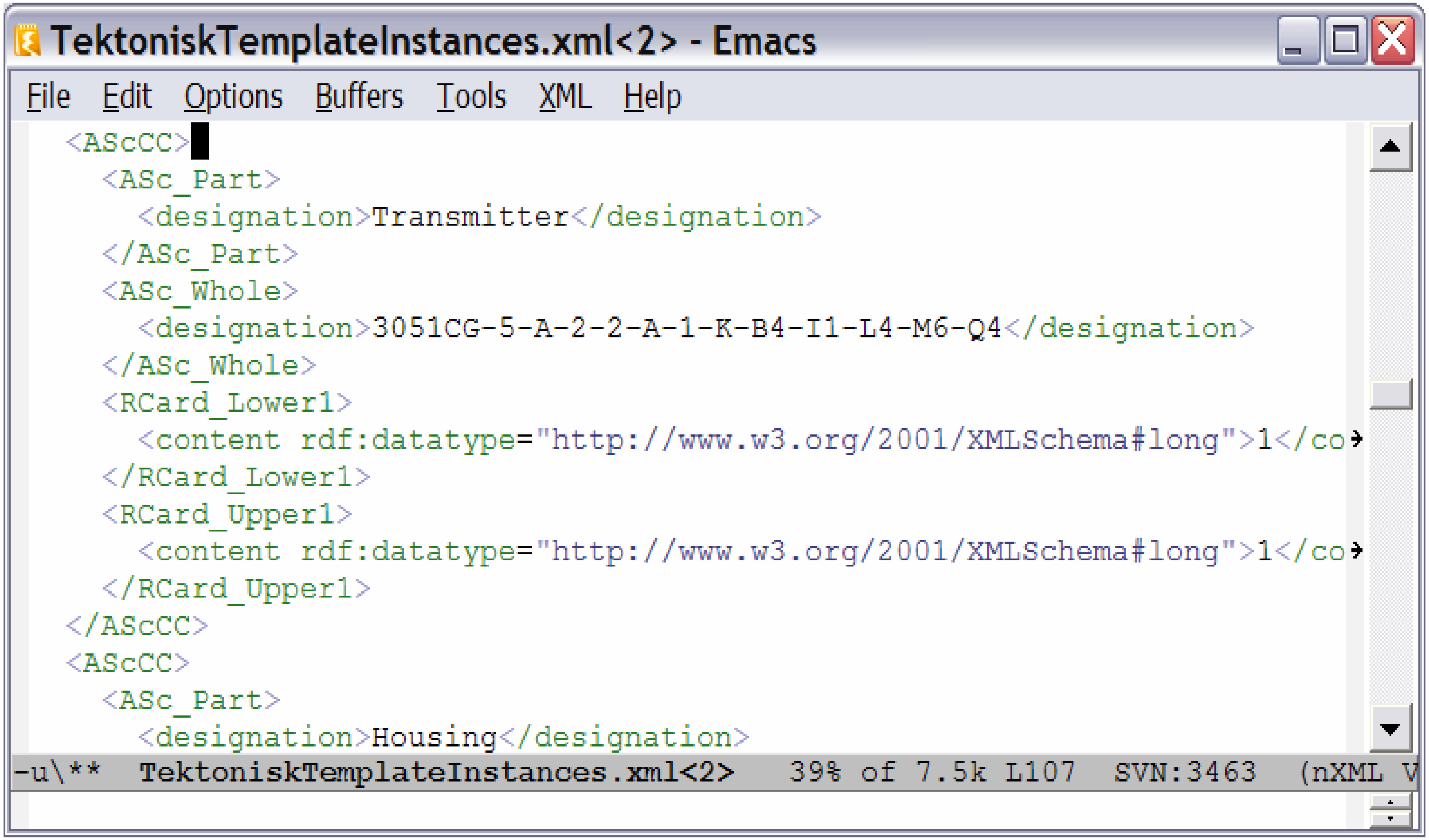
part2:hasRecordCop

part2:hasRec

Visualization in Protégé

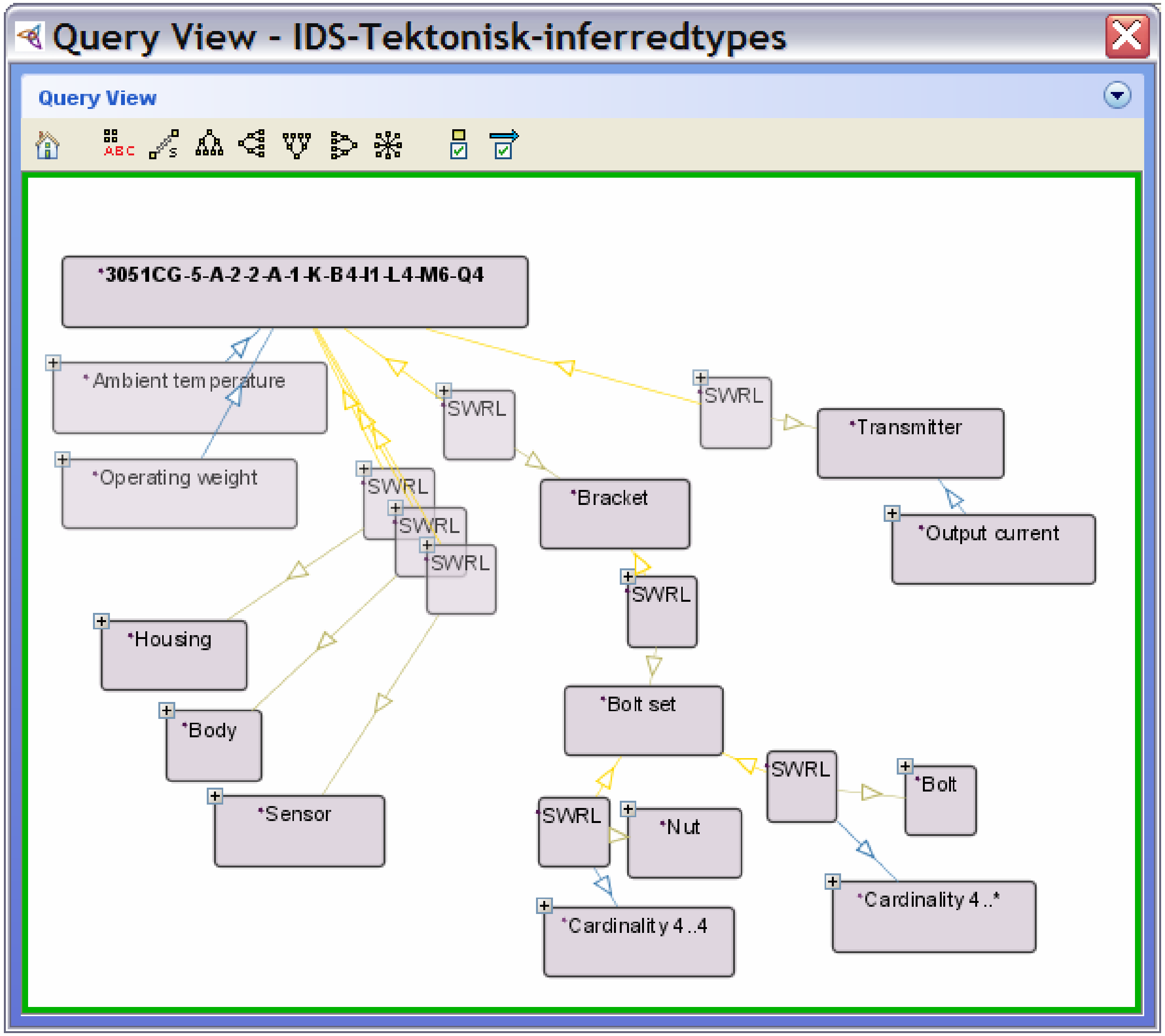

Query View - IDS-Tektonisk-inferredtypes
Query View
ABC
*SWRL
*Temperature Property 85 °C
*SWRL
*Number: 85
*SWRL
*85
*SWRL
*SWRL
*Ambient temperature
*3051CG-5-A-2-2-A-1-K-B4-I1-L4-M6-Q4
*Temperature PropertyRange -40 - 85 °C
*Temperature
*°C
*SWRL
*SWRL
*SWRL
*Temperature Property -40 °C
*SWRL
*Number: -40
*SWRL
*-40

Breakdown using *Parts* template

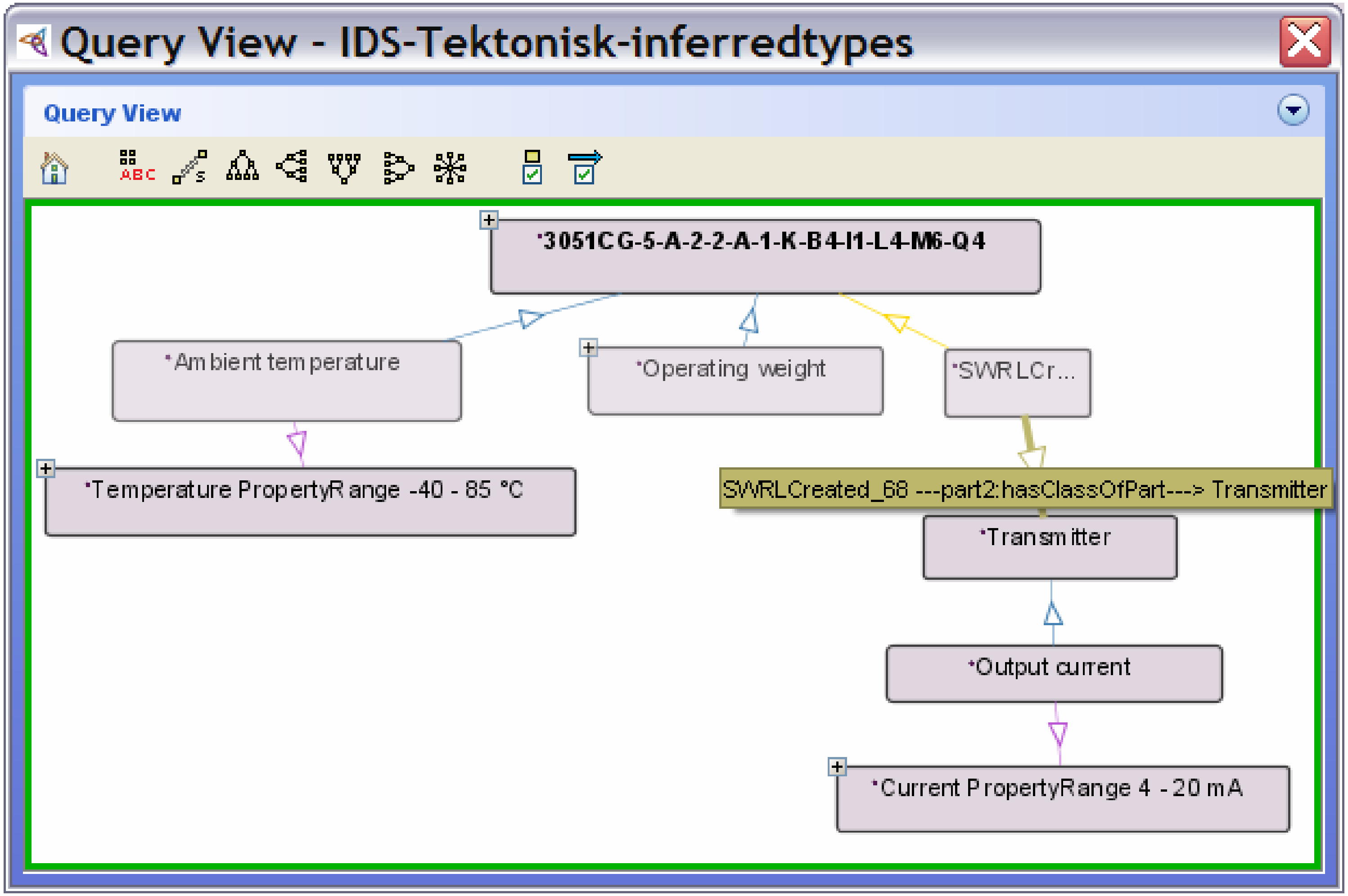

TektoniskTemplateInstances.xml<2> - Emacs
File Edit Options Buffers Tools XML Help

```
  <AScCC>
    <ASc_Part>
      <designation>Transmitter</designation>
    </ASc_Part>
    <ASc_Whole>
      <designation>3051CG-5-A-2-2-A-1-K-B4-I1-L4-M6-Q4</designation>
    </ASc_Whole>
    <RCard_Lower1>
      <content rdf:datatype="http://www.w3.org/2001/XMLSchema#long">1</co
    </RCard_Lower1>
    <RCard_Upper1>
      <content rdf:datatype="http://www.w3.org/2001/XMLSchema#long">1</co
    </RCard_Upper1>
  </AScCC>
  <AScCC>
    <ASc_Part>
      <designation>Housing</designation>
-u\**  TektoniskTemplateInstances.xml<2>   39% of 7.5k L107   SVN:3463   (nXML V
```

Breakdown visualized in Protégé

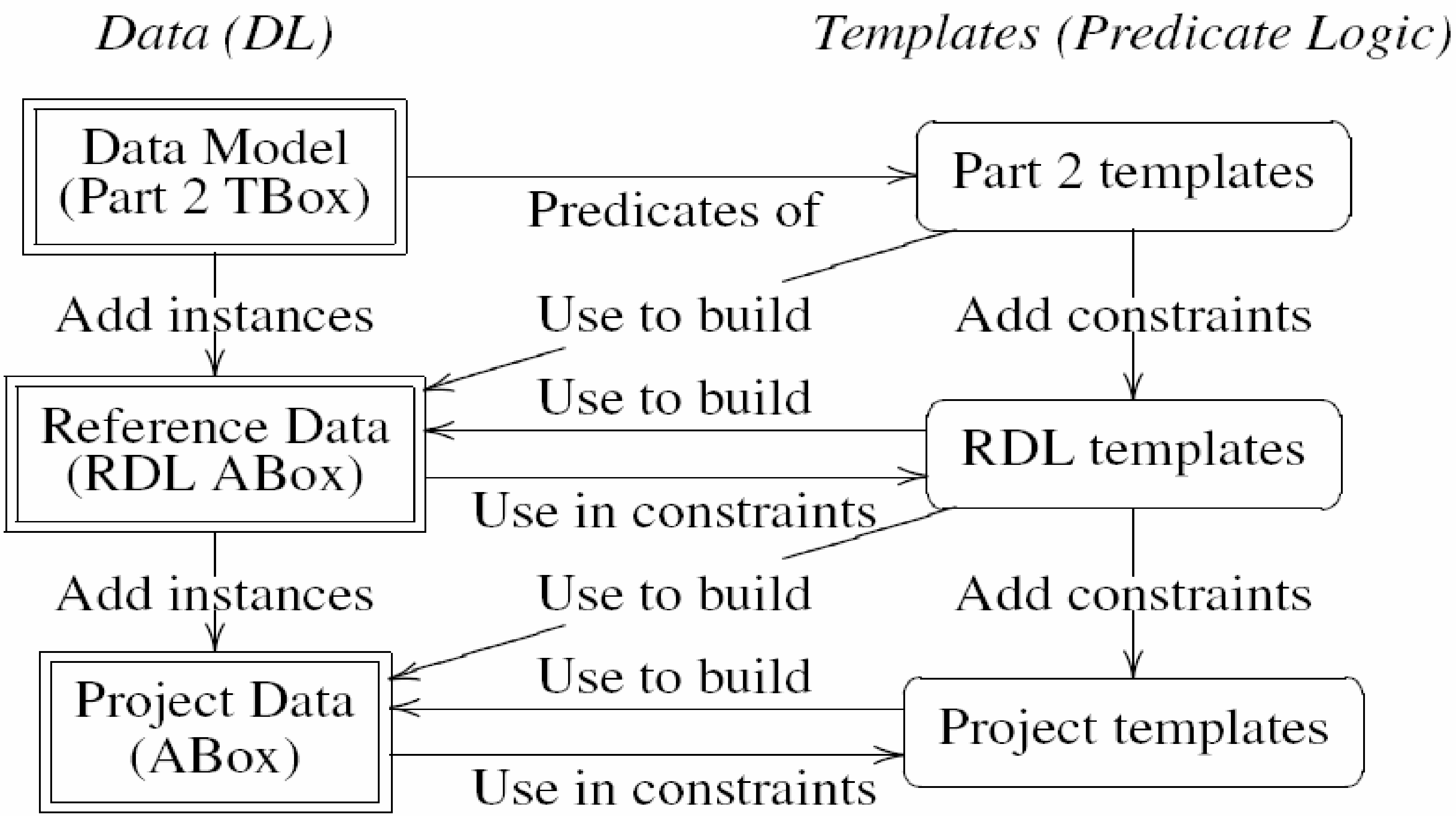


Ranges of the whole and of parts


Query View - IDS-Tektonisk-inferredtypes
Query View
ABC
3051CG-5-A-2-2-A-1-K-B4-I1-L4-M6-Q4
Ambient temperature
Operating weight
SWRLCr...
Temperature PropertyRange -40 - 85 °C
SWRLCreated_68 ---part2:hasClassOfPart---> Transmitter
Transmitter
Output current
Current PropertyRange 4 - 20 mA

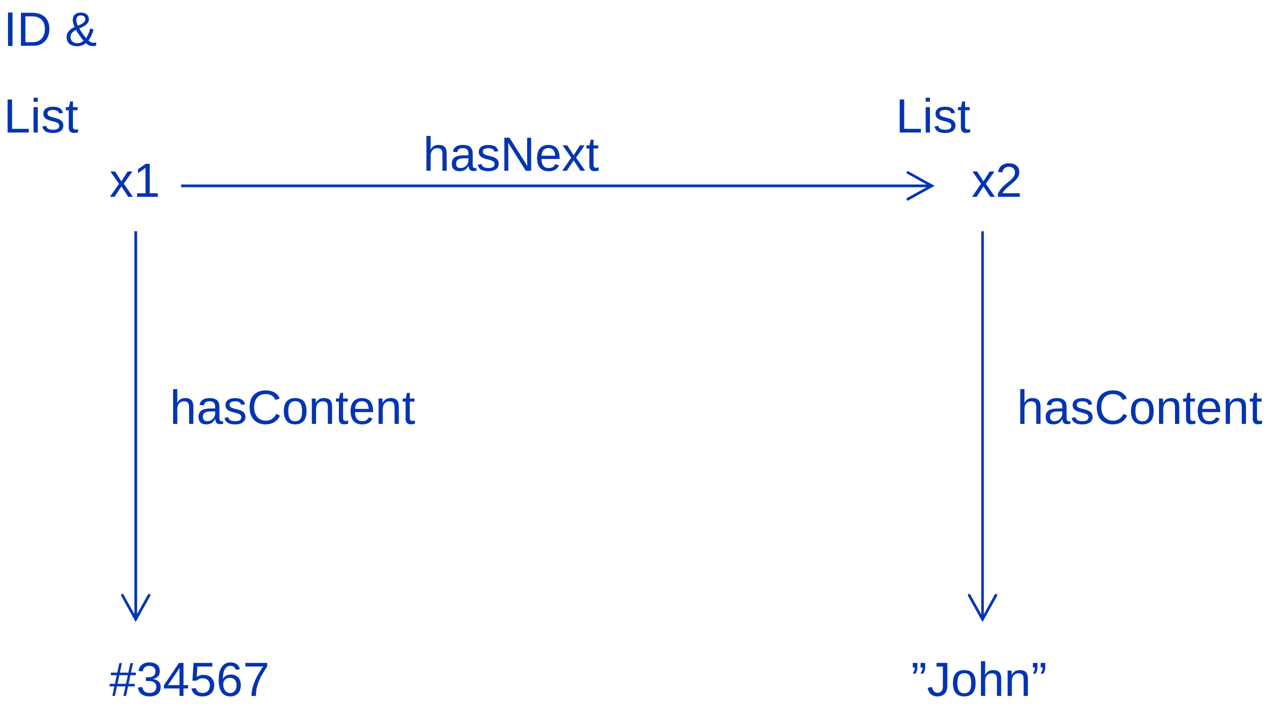
So many screenshots. Any moral to this?

ISO 15926 languages: SW perspective



The template statement is uninterpreted

- The signature is just a list of types
- Sample statement *ID(#34567, ”John”):*



ID &

List

x1

hasNext

List

x2

hasContent

hasContent

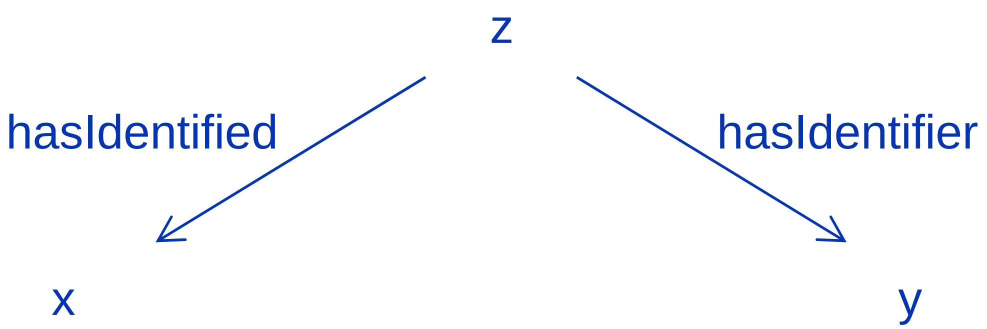
#34567

”John”

Rules encode interpretation patterns

- If ID(x, y),
  - there is some z s.t. z is a ClassOfIdentification relation, with
    - x in the hasIdentified,
    - y in the hasIdentifier role

Class of identification



z

hasIdentified

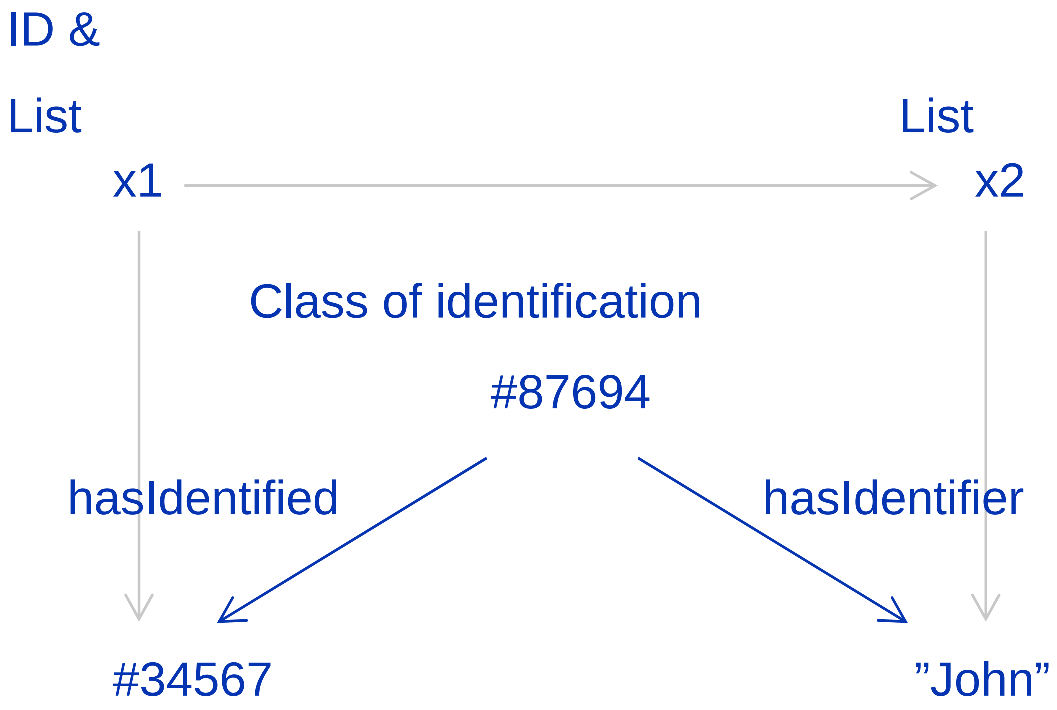
hasIdentifier

x

y

An interpreted statement

- The rule provides a structure that has interpretation in terms of Part 2



ID &

List

List

x1

x2

Class of identification

#87694

hasIdentified

hasIdentifier

#34567

”John”

Sample rules

- coidTriple(x, y, z) iff
  - ClassOfIdentification(x) & hasIdentified(x, y) & hasIdentifier(x, z)
- ID(x, y) iff
  - $\exists$z(coidTriple(z, x, y)
- IDC(x, y, z) iff
  - ID(x, y) & $\forall$u(coidTriple(u, x, y) $\rightarrow$ u $\in$ z
- etc.

- Probably, all this can be expressed in SWRL.*
- *Regular* logic?

**RIF!*

An extended ontology

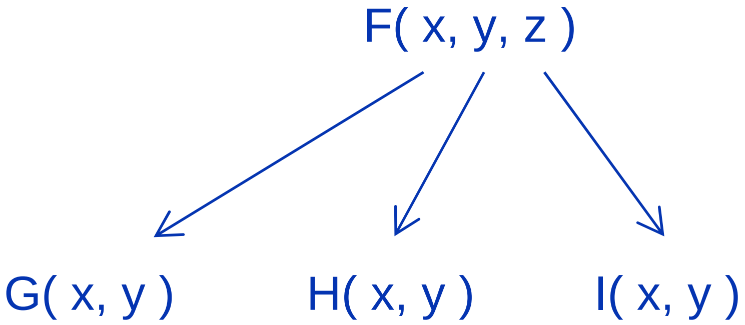
- Develop templates to cover the expressive needs
- This provides a theory of industrial notions
- This theory is also reference data
- Templates extend the RDL

- We need appropriate tools and theory to work with this ontology.

Extended ontology, new work to be done

Consistency checking for templates

- If F has a rule like the following,



F(x, y, z)

G(x, y) H(x, y) I(x, y)

- but other rules (or Part 2 itself) tell us,

G(x, y) iff *not* I(x, y)

- then F is unsatisfiable. A reasoner should be able to discover this.

Consistency checking for templates

- Experimental checking

  Make statements in the template language,

  Execute rules,

  Run reasoner to check consistency of the result

- Systematic checking

  Is the template language consistent?

  etc.

Reasoner tasks

- Satisfiability of templates
- Equivalence between templates
- Subsumption hierarchies (more/less general templates)
- Simplification (of signatures, rules)
- Explanation (why was my statement rejected?)

- Challenge: To provide these not just for Part 2, but for templates as well.