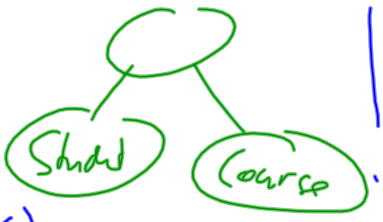
Q & A Session

- Classes versus object properties / data properties
-
  - When to establish what?
  - rules in Protege 4.x
  - new def. of rules → statement based
- consistency of rules can't be checked without the reasoner

Rules

Protégé 3.x



Student(?S) ∧ Course(?C)

→ StudyCourse(?S, ?C)

Protégé 4.x

· Courses(?C) are_tough,

min 1 Room(?R)

?

Statement:

If student joins MasterCourse

than he is a MasterStudent

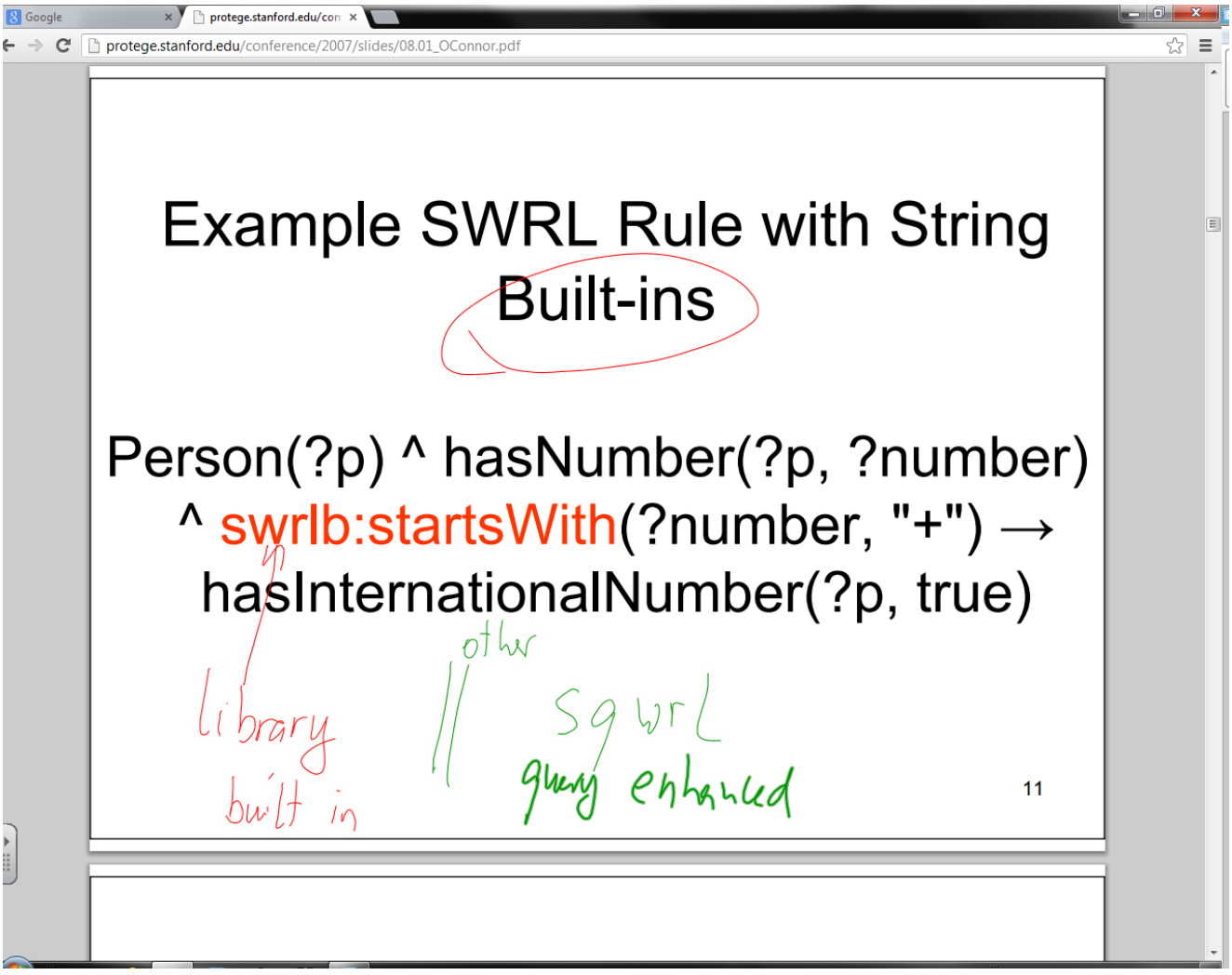
Student(?S), MasterCourse(?MC),

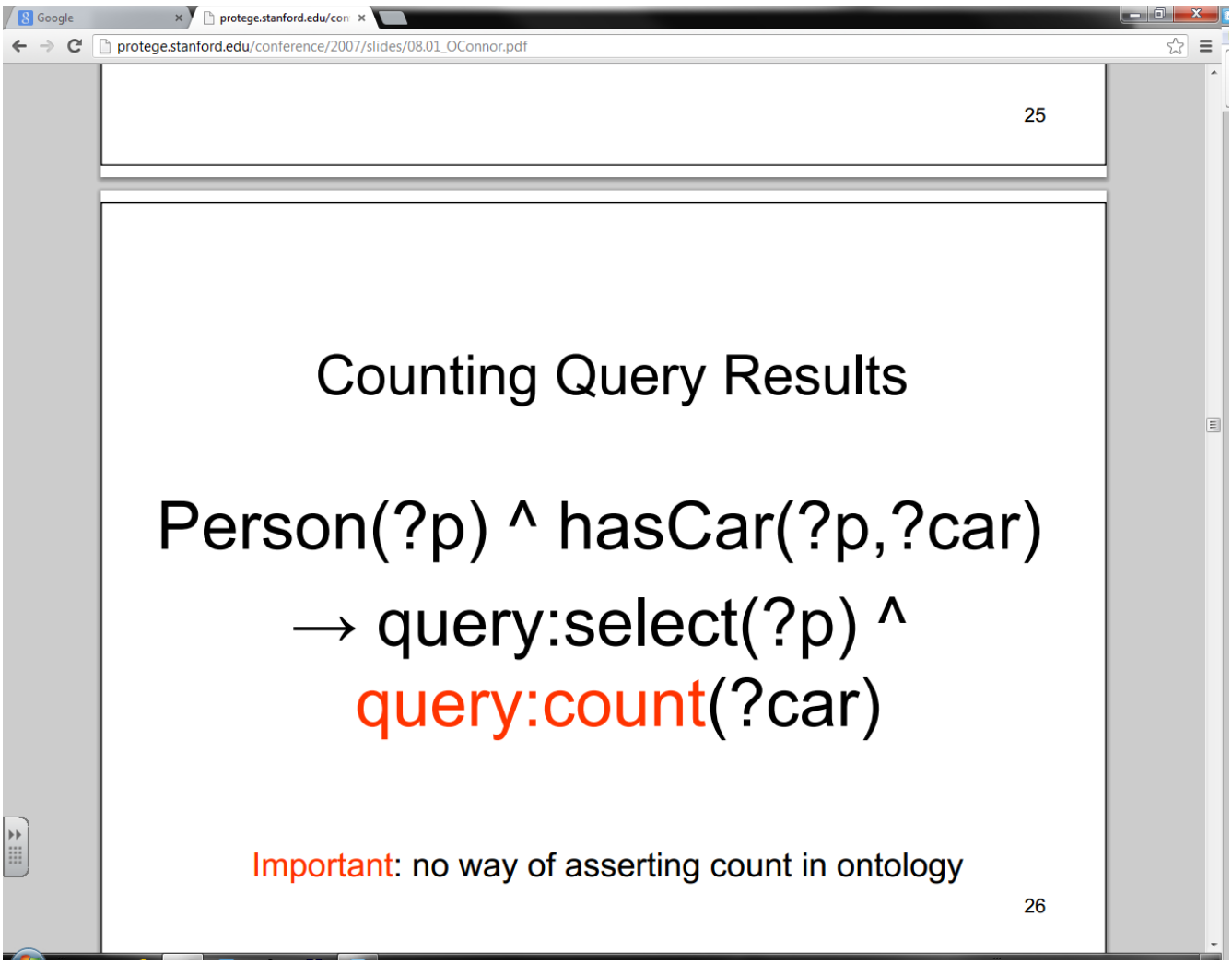
hasCourse(?S, ?MC) → MasterStudent(?S)

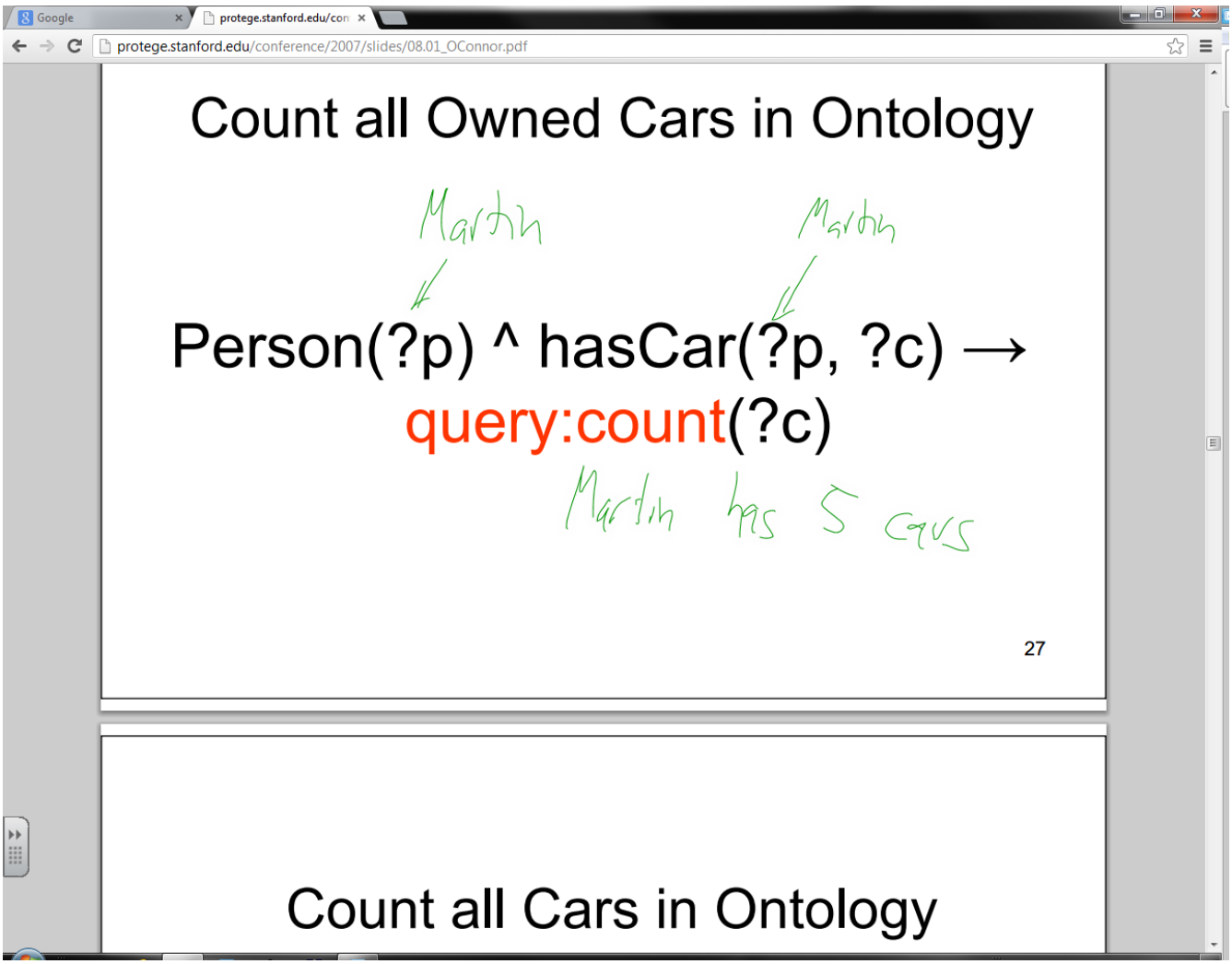
SWRL, SQWRL

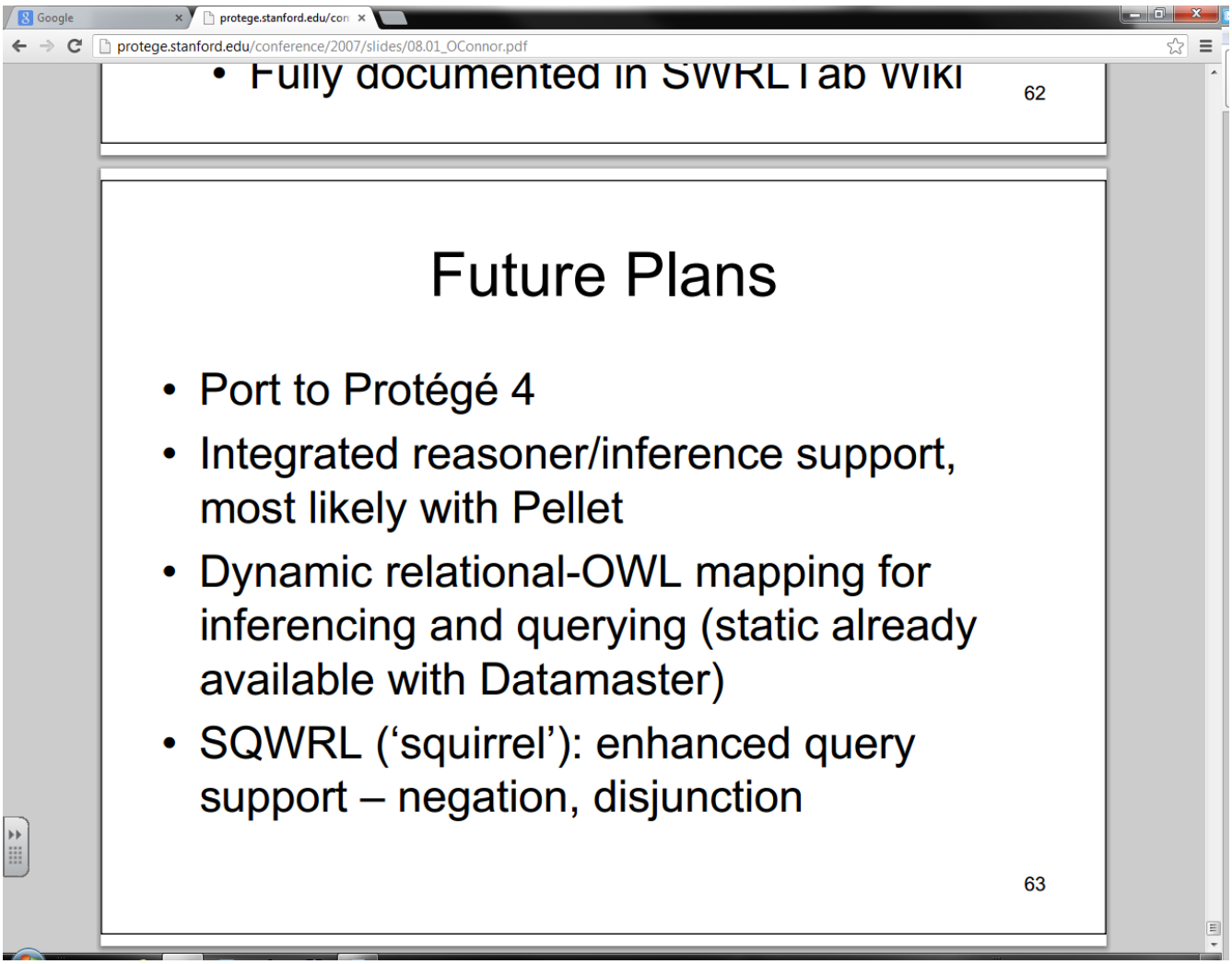
SWRL, ~~SQWRL~~

http://protege.stanford.edu/conference/2007/slides/08.01_OConnor.pdf

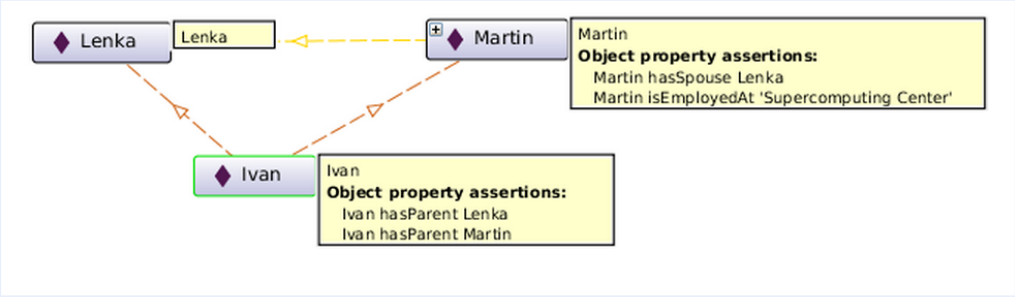



Google
protege.stanford.edu/con
protege.stanford.edu/conference/2007/slides/08.01_OConnor.pdf
25
Counting Query Results
Person(?p) ^ hasCar(?p,?car)
→ query:select(?p) ^
query:count(?car)
Important: no way of asserting count in ontology
26


protege.stanford.edu/conference/2007/slides/08.01_OConnor.pdf
Count all Owned Cars in Ontology
Martin
Martin
Person(?p) ^ hasCar(?p, ?c) →
query:count(?c)
Martin has 5 cars
27
Count all Cars in Ontology


protege.stanford.edu/conference/2007/slides/08.01_OConnor.pdf
• Fully documented in SWRLTab Wiki
62
Future Plans
• Port to Protégé 4
• Integrated reasoner/inference support, most likely with Pellet
• Dynamic relational-OWL mapping for inferencing and querying (static already available with Datamaster)
• SQWRL ('squirrel'): enhanced query support – negation, disjunction
63

Let's reuse our example of Ivan, the son of Martin and Lenka. The symmetric property *hasSpouse* connects Martin and Lenka.




```
SymmetricObject
ObjectPropertyA
ObjectPropertyA
ObjectPropertyA
```

We can add a SWRL rule saying that an individual X from the Person class, which has parents Y and Z such that Y has spouse Z, belongs to a new class *ChildOfMarriedParents*. Such rule is best described in the Protege syntax:

```
Person(?x), hasParent(?x, ?y), hasParent(?x, ?z), hasSpouse(?y, ?z) -> ChildOfMarriedParents(?x)
```

It can be described in functional syntax too:

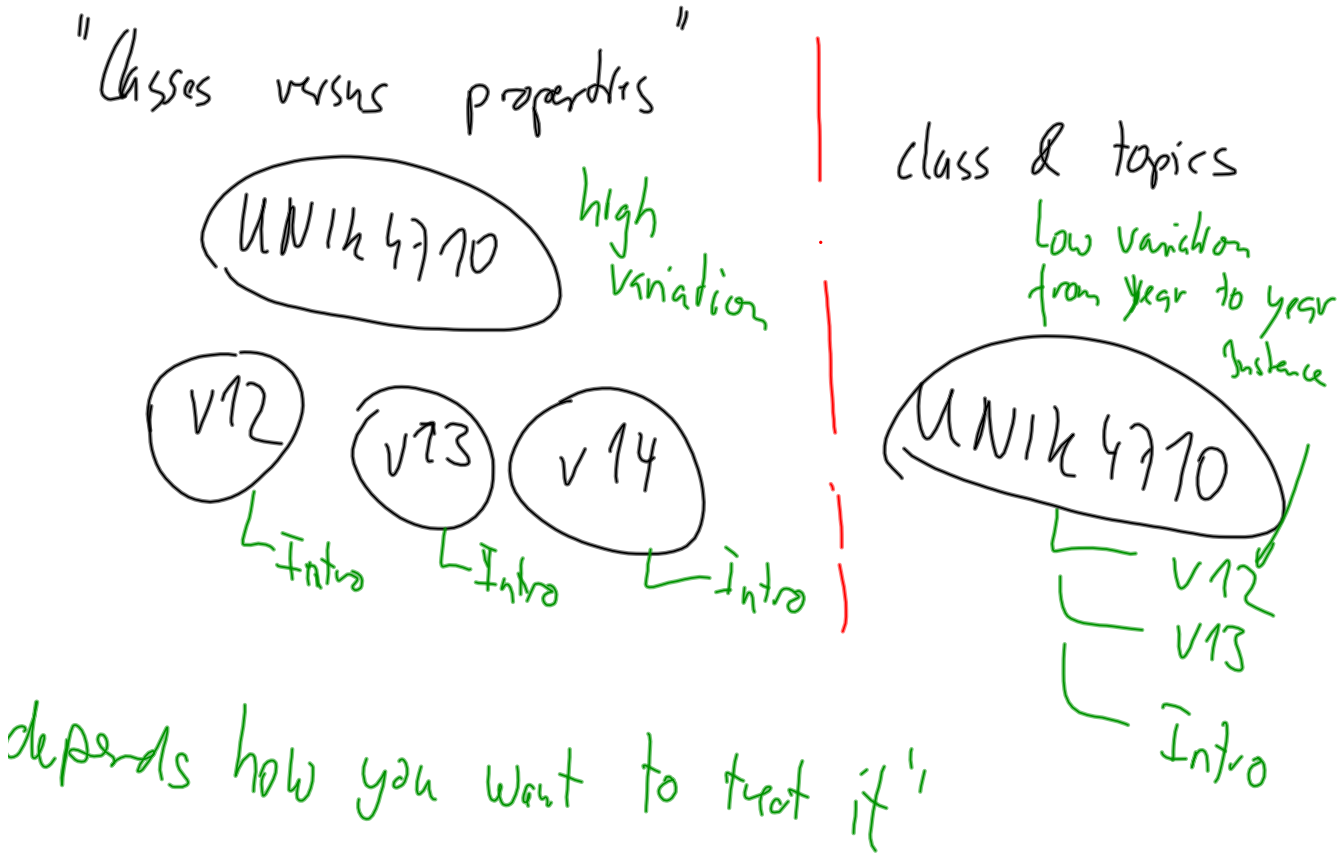
```
Prefix(var:=<urn:swrl#>)

Declaration( Class( :ChildOfMarriedParents ) )
SubClassOf( :ChildOfMarriedParents :Person )

DLSafeRule(
    Body(
        ClassAtom( :Person Variable(var:x))
        ObjectPropertyAtom( :hasParent Variable(var:x) Variable(var:y) )
        ObjectPropertyAtom( :hasParent Variable(var:x) Variable(var:z) )
        ObjectPropertyAtom( :hasSpouse Variable(var:y) Variable(var:z) )
    )
    Head(
        ClassAtom( :ChildOfMarriedParents Variable(var:x) )
```

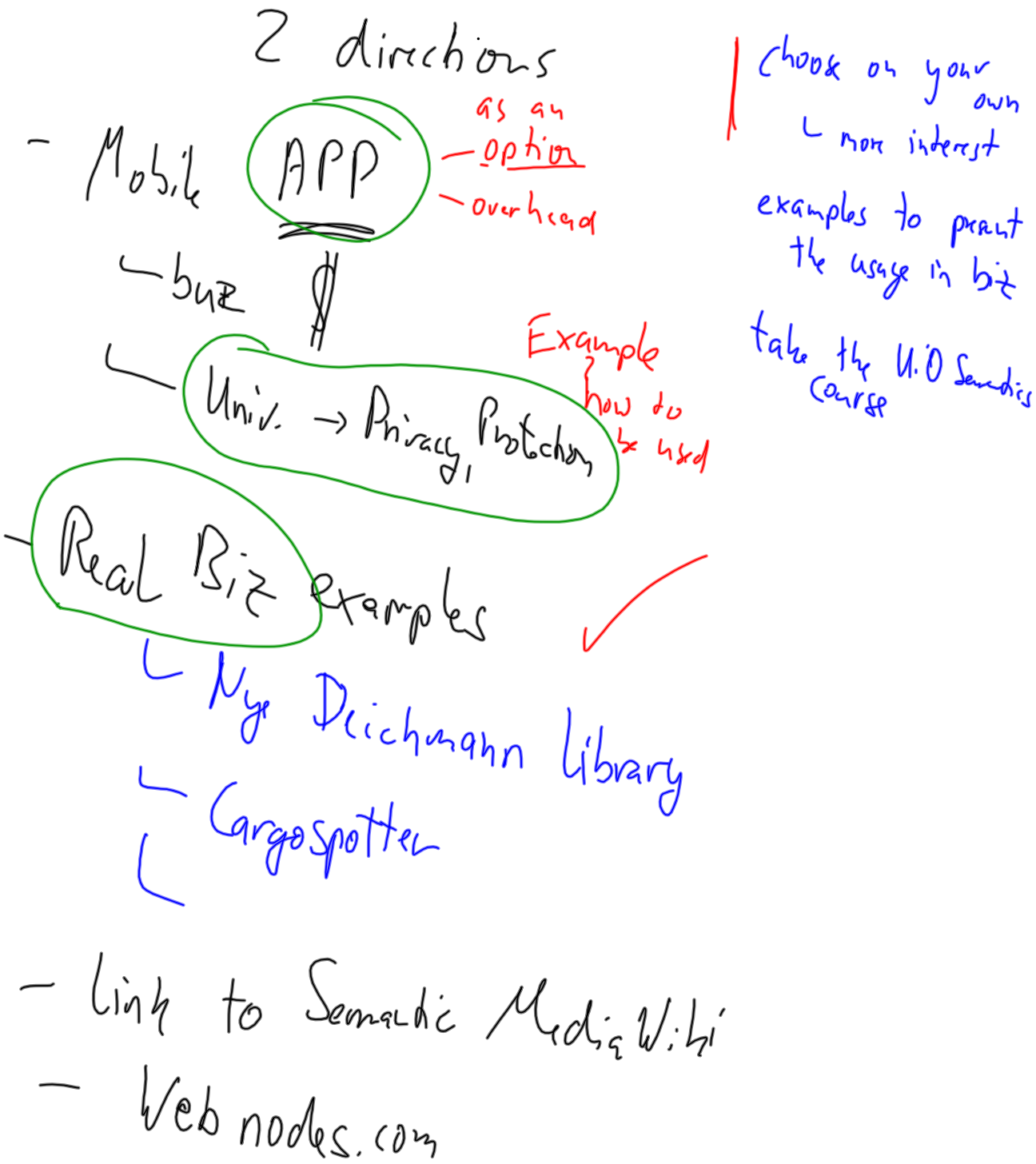
for teaching in Protege 4.2

- SPARQL tab, works by

OWL → rdf → listings

SPARQL


"Classes versus properties"
UNIK4770
high variation
V12
V13
V14
Intro
Intro
Intro
class & topics
Low variation from year to year
Instance
UNIK4770
V12
V13
Intro
depends how you want to treat it"


2 directions
- Mobile
APP
as an
- option
- overhead
- buz
Univ. → Privacy, Protection
Example
how do
be used
Real Biz examples
Ny Deichmann library
Cargospotter
choose on your own
more interest
examples to prove
the usage in biz
take the U.O Semantics
course
- link to Semantic MediaWiki
- Webnodes.com