

OWL API

- Facts**
- code example
 - communicate with owl
 - thinks OWL

Criteria

- Evaluation
Recommendations**

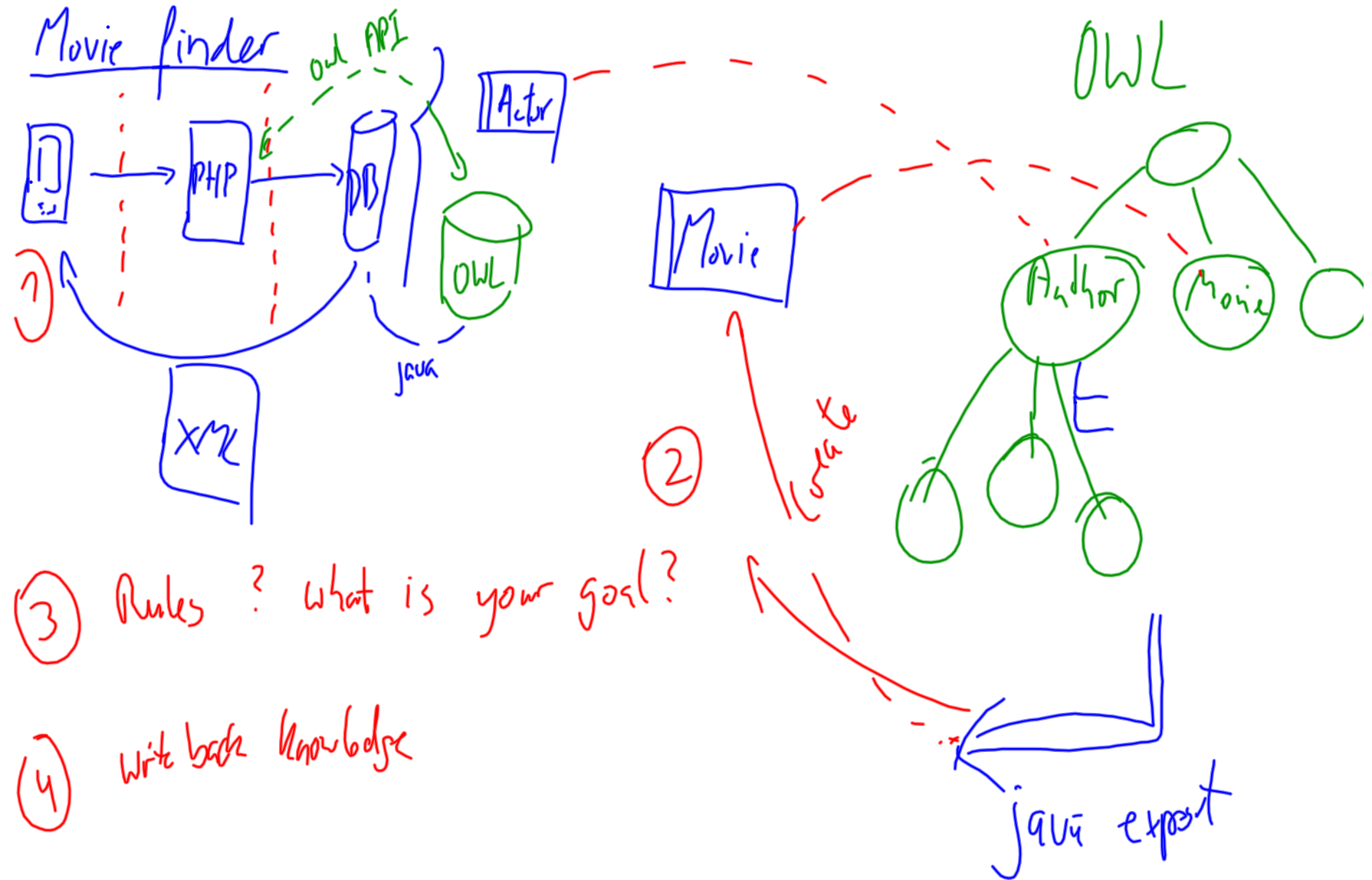
- more logic/business process oriented



Java export

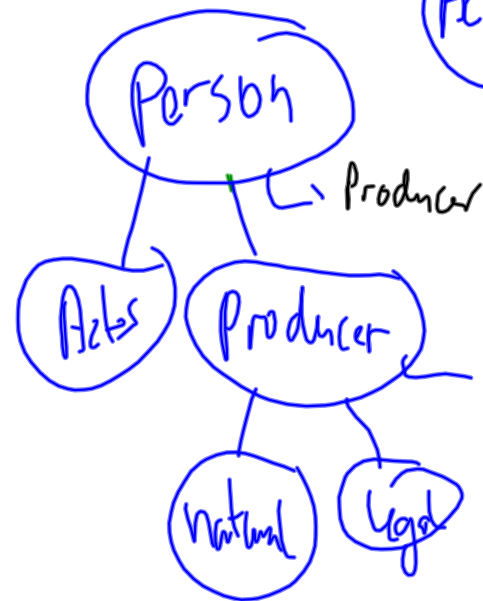
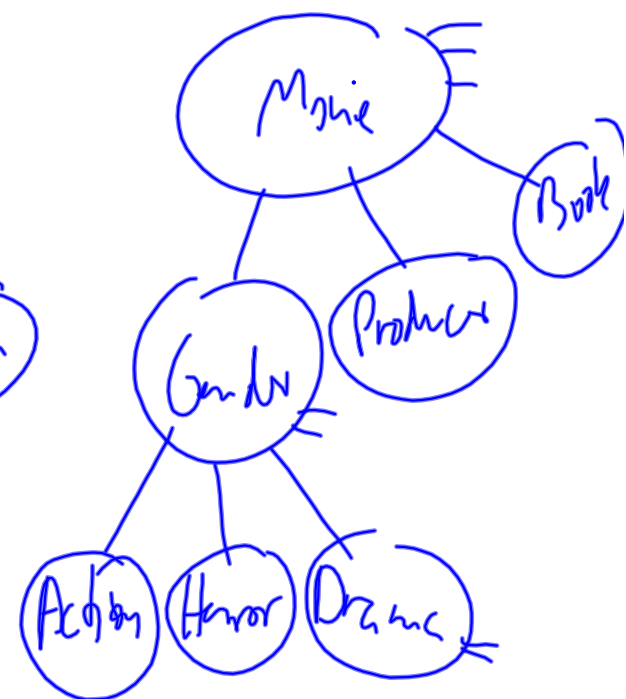
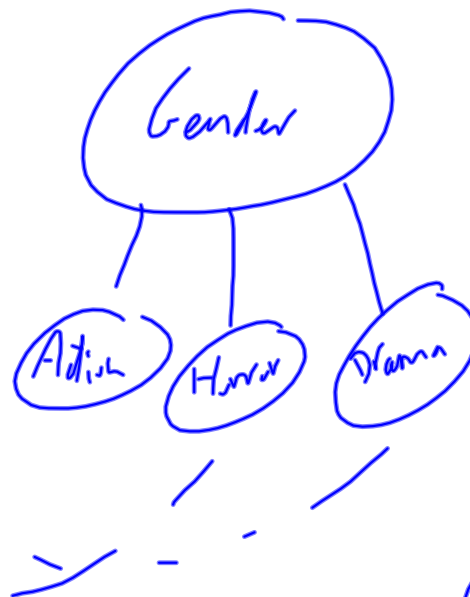
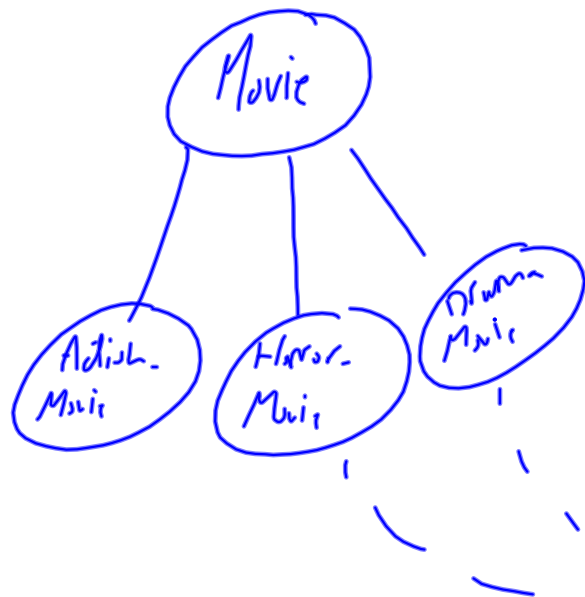
- easy export // user friendly
- creates structure for java prog
- allows to "think" java
- just classes and relations
- don't need to think OWL
- implicit knowledge
- programming, object oriented
- does not need OWL knowledge

= how to write back knowledge into .owl

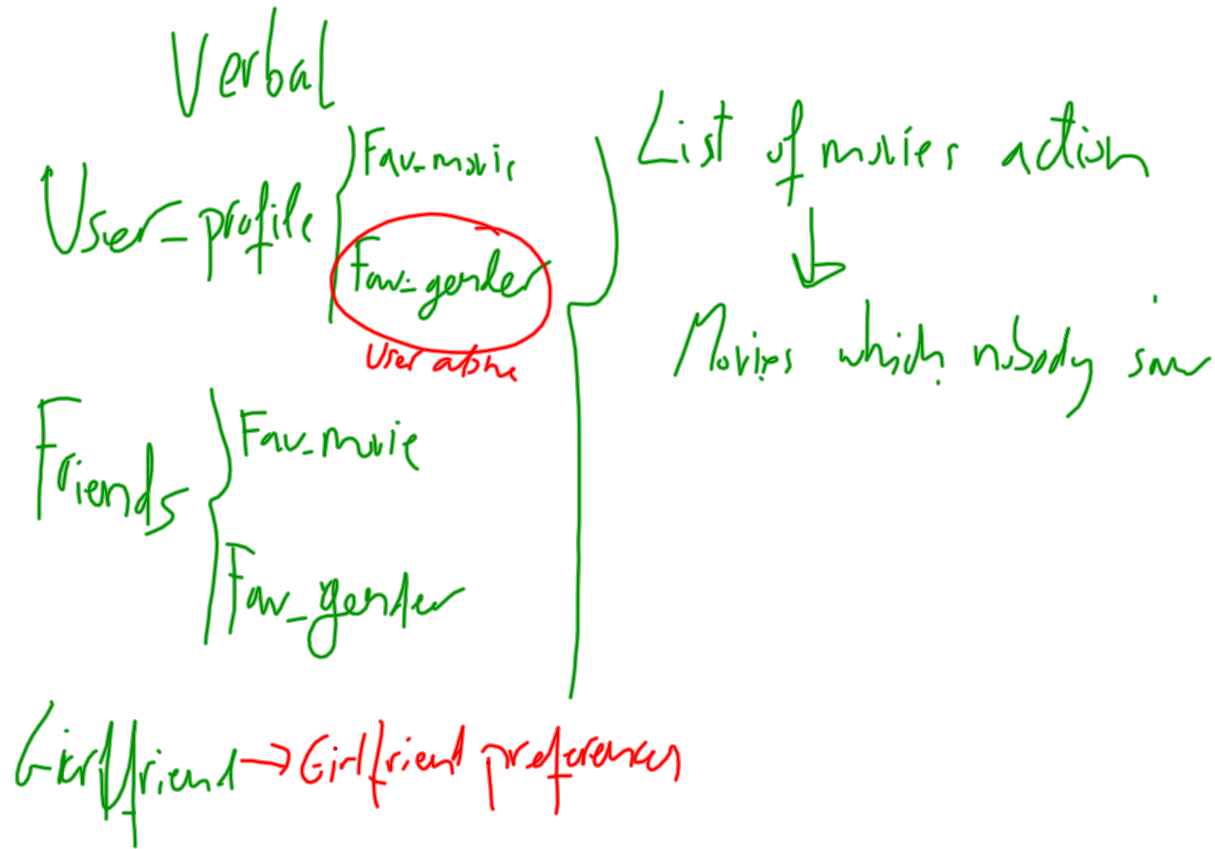


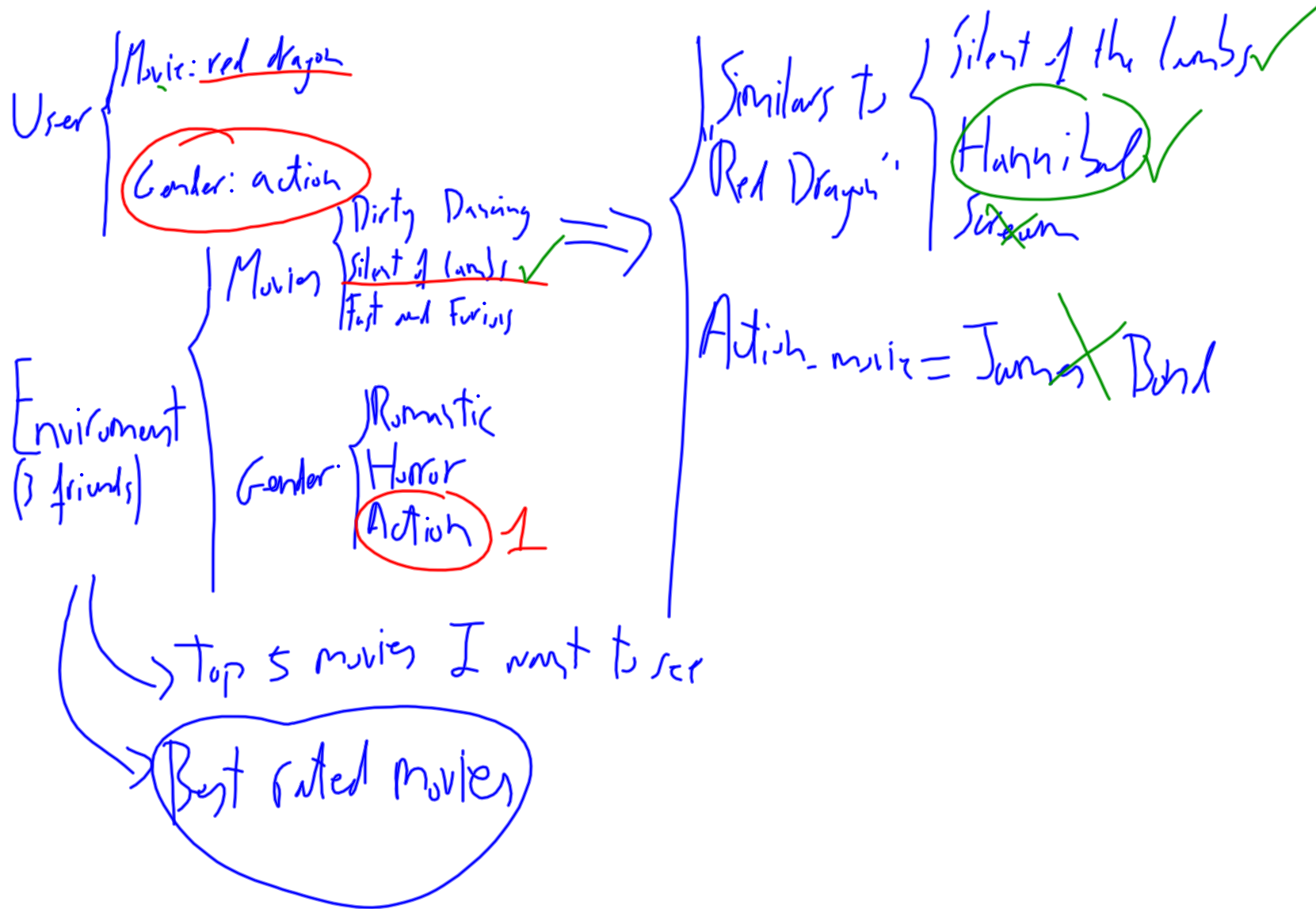
③ Rules? what is your goal?

④ write back knowledge



Rules: "business rules"





.net API frameworks

- DOTNETRDF



OWL → RDF export
supported by Protege 4.2

~~OWL 2 API .net~~
not yet
Robert Weiss

Alternative:

Use Java export from
Protege

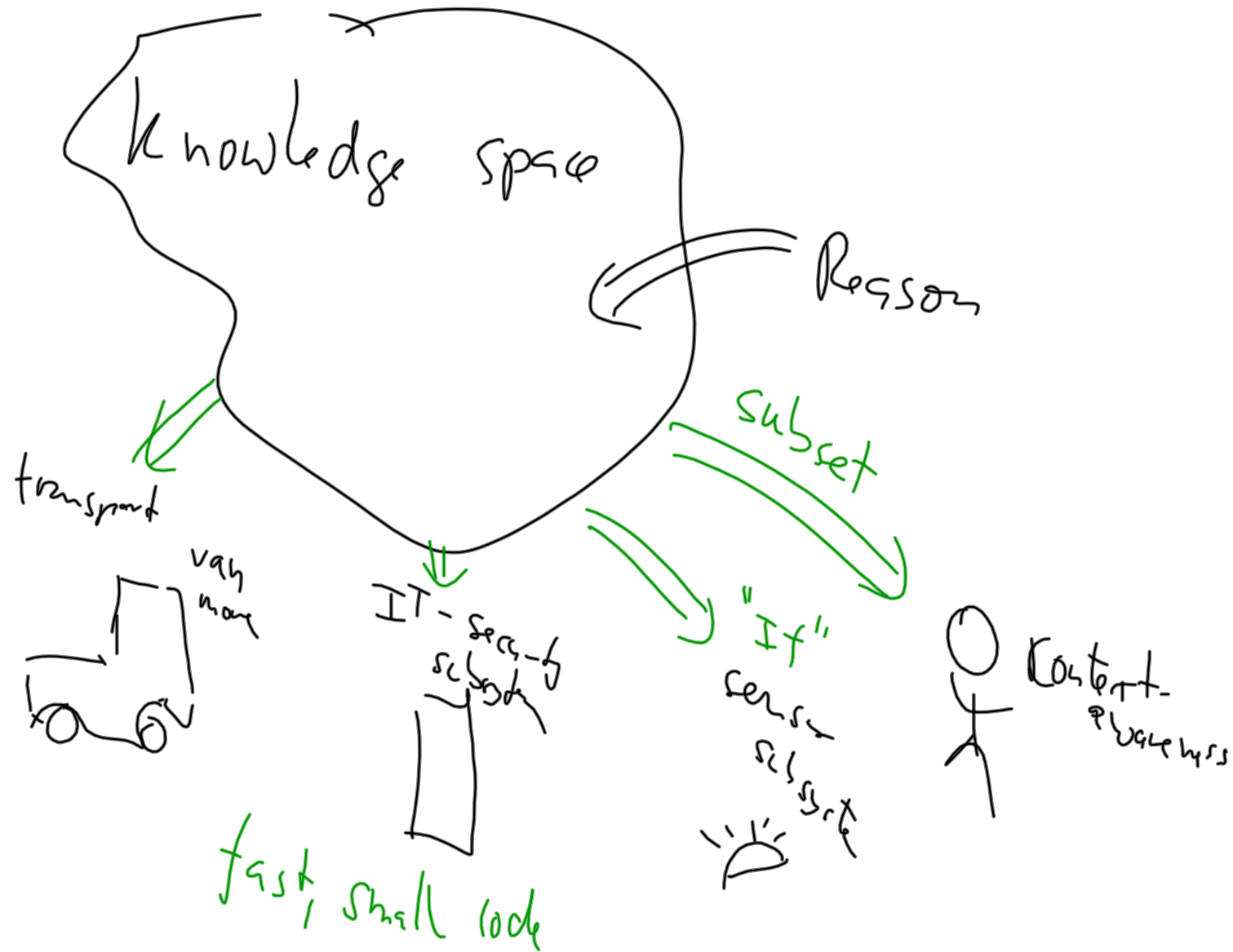
Logic representation

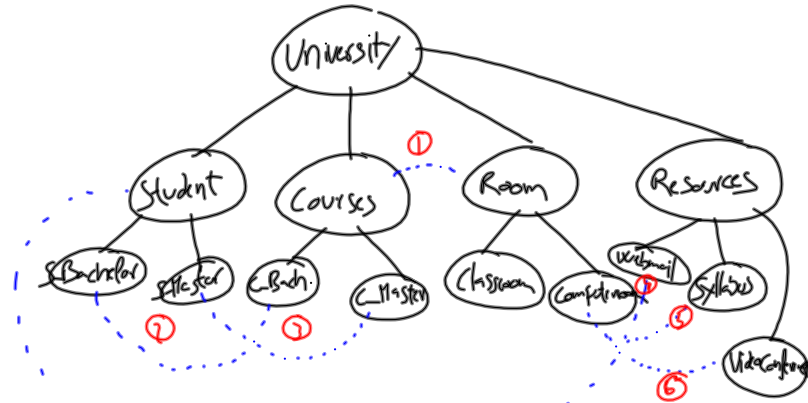


1.) Structured data "think logic"

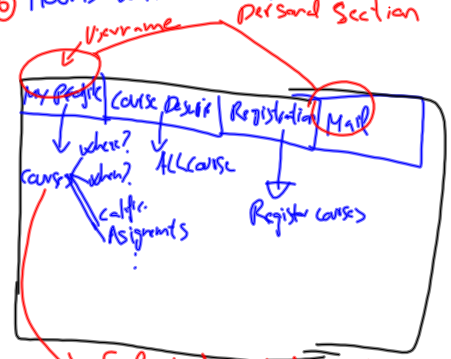
2.) Reasoning "consistency", "implicit knowledge"
SWRL, FacT++, Pellets, *demonstrat*

3.) applicability
representation through *K, D, P*





- ① which room the courses are taught
- ② Courses of Bach. Students
- ③ " " Master
- ④ Student webmail
- ⑤ " " Syllabus
- ⑥ Rooms with videconfernce resource



→ Select 'User' from OR 'Rite'