Ω-RIDL

A Language and Interpreter for Ontological Commitments

Overview

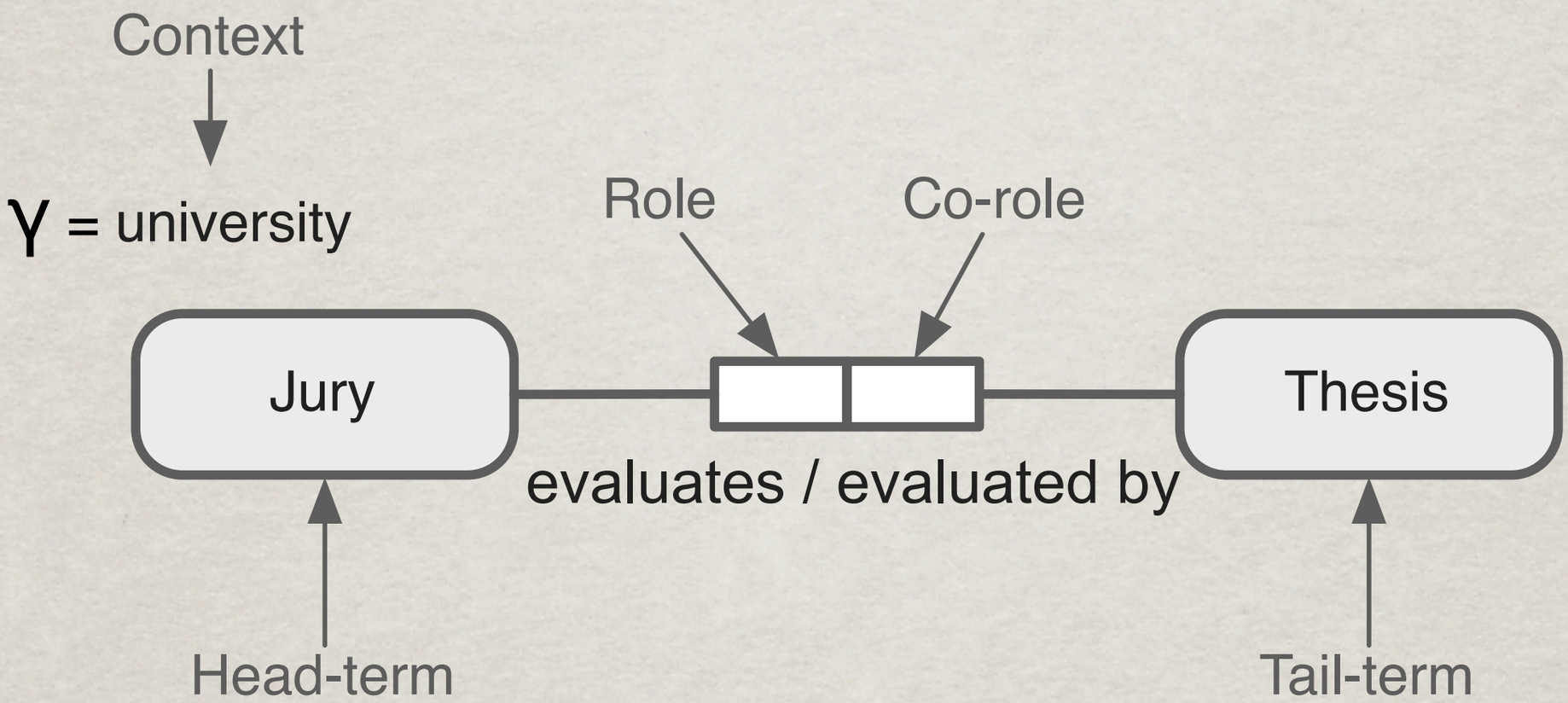
- DOGMA Foundations
- DOGMA Studio and T-Lex
- **Ω**-RIDL Ontological Commitment Language
- Conclusions and Future Work

DOGMA

Foundations

Developing Ontology Grounded
Methods and Applications

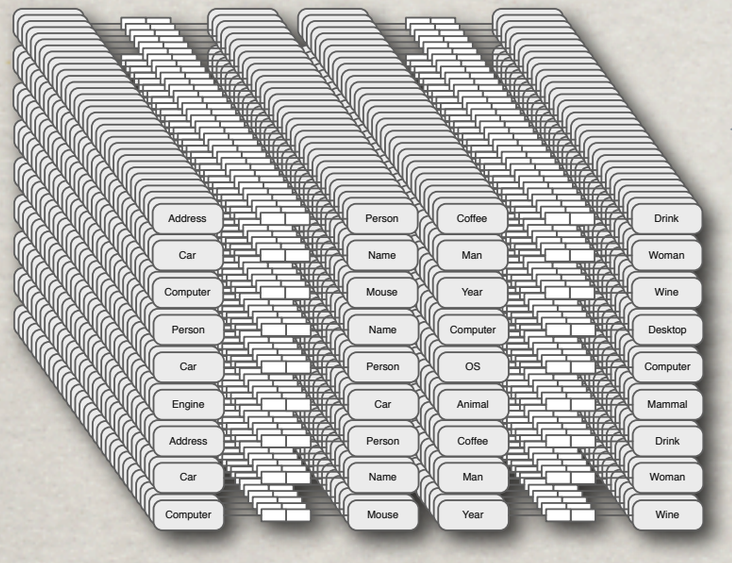
Lexon



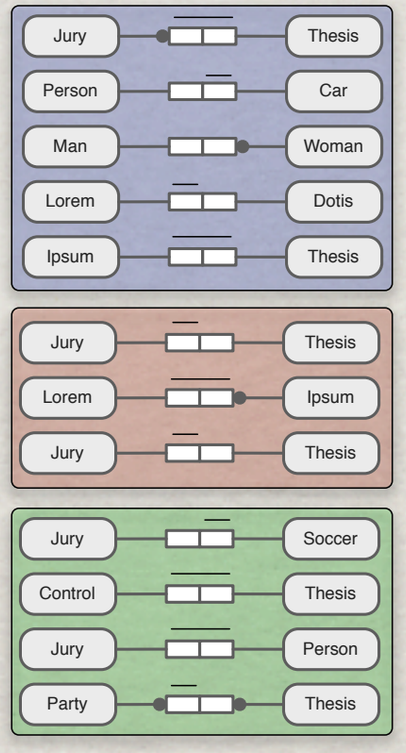
- Plausible elementary binary fact type
- (context, term) refers to a unique concept

DOGMA Layers

Lexon Base



Commitment Layer



Applications

"Double articulation principle"

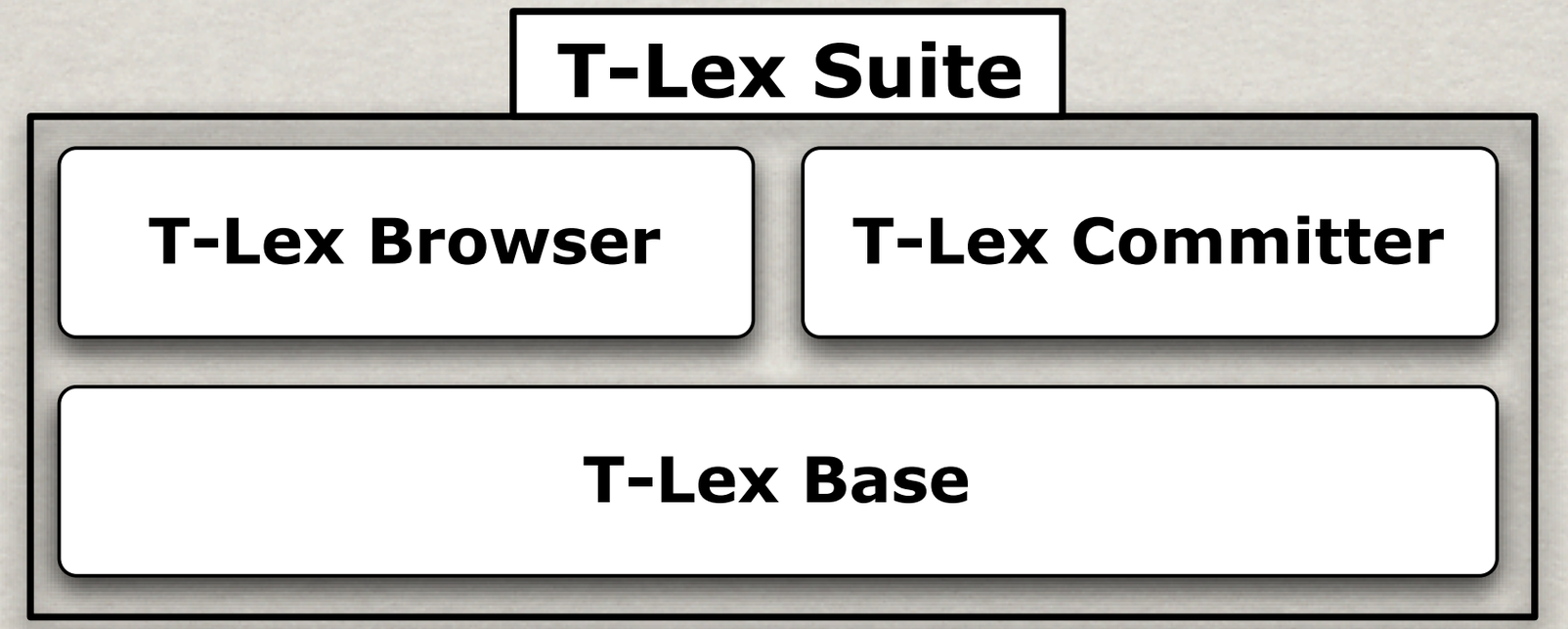
Commitment Layer

Commitments define an interpretation of (a subset of) the Lexon Base by:

- selecting lexons
- semantically constraining the use of the selected lexons by imposing constraints

DOGMA Studio & T-Lex

DOGMA STUDIO



Resource – T-Lex Commitment Editor – Eclipse Platform

File Edit View Navigate Search Project Run Window Help

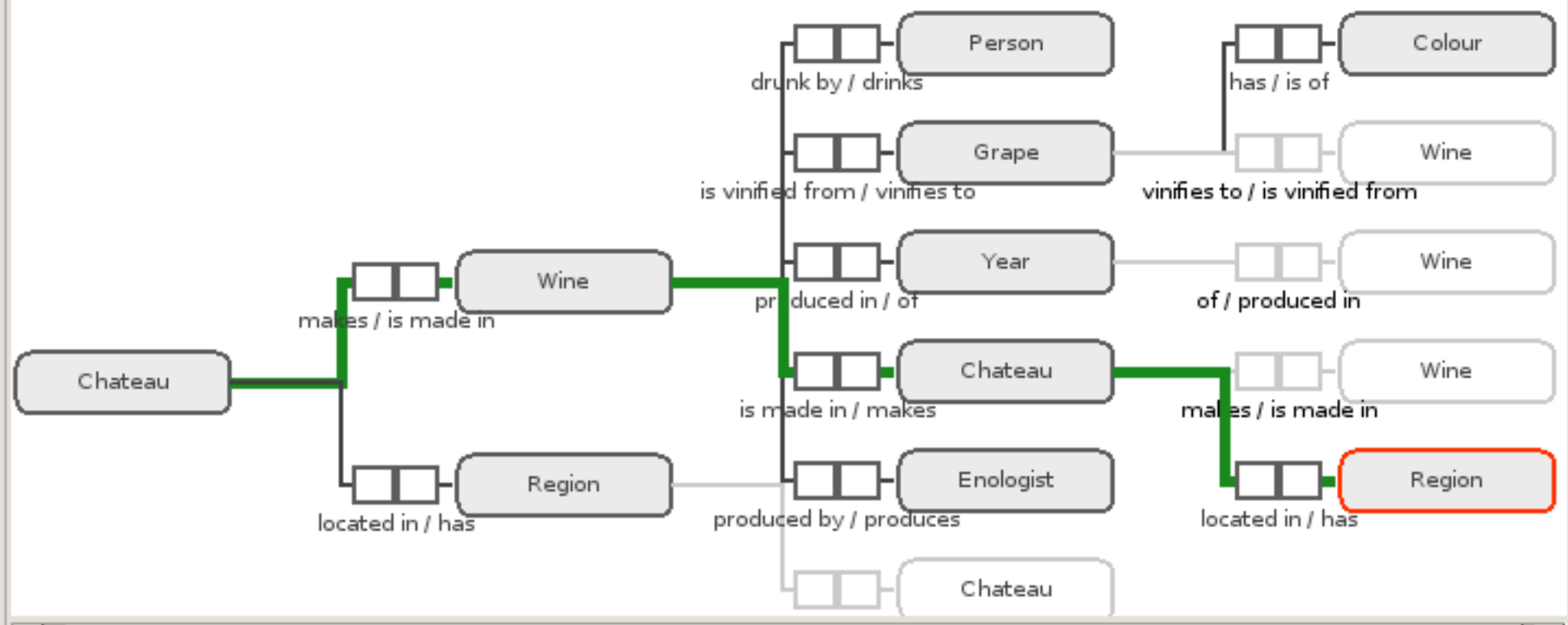
100% Resource

Navigator | LexonBase Explorer

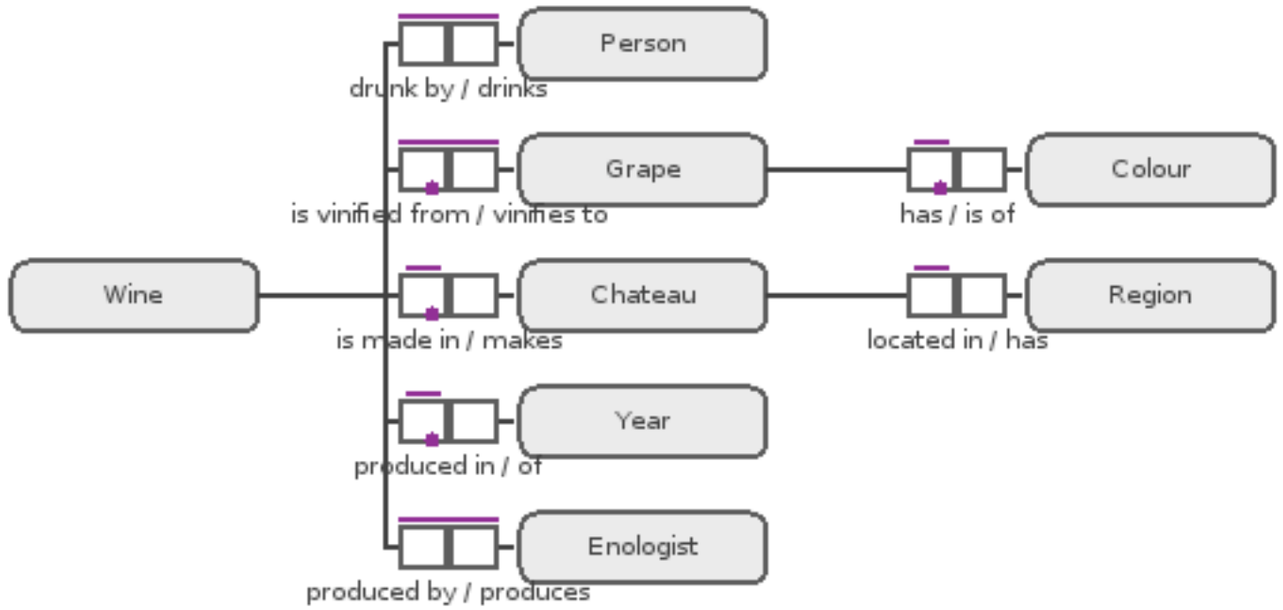
- WalloonPublEmpl_FOREM_TMPL_UCO_Jol
- WalloonPublEmpl_FOREM_TMPL_UCO_Jol
- WalloonPublEmpl_FOREM_TMPL_UCO_Jol
- WalloonPublEmpl_FOREM_TMPL_UCO_Jol
- WalloonPublEmpl_FOREM_TMPL_UCO_Jol
- WalloonPublEmpl_FOREM_TMPL_UCO_Jol
- WalloonPublEmpl_FOREM_TMPL_UCO_Jol
- weather
- Wikipedia_Wine
- Wine Encyclopedia
  - Chateau
  - Colour
  - Enologist
  - Grape
  - Person
  - Region
  - Wine
  - Year
- Wine Ontology (08/2006)

Outline

T-Lex Lexon Base Browser



*T-Lex Commitment Editor



Ω-RIDL

Ω-RIDL

- Language for:
  - defining ontological commitments
  - conceptual querying
- Based on RIDL

Ω-RIDL

Ontological commitments consist of three parts:

- Contextual declaration
- Lexical interpretation
- Semantic rules

An Example

```
define commitment
   in context WINEBOOK
   with subsumption ISA / SUBSUMES
   lexical interpretations
      map WINES.WINE_NAME
      on WINE HAS NAME
   semantic constraints
      each WINE HAS exactly one NAME
end
```

Conclusions & Future Work

Conclusion & Future Work

- **Ω**-RIDL will define ontological commitments and support conceptual querying
- T-Lex tool will need to be extended to support export to **Ω**-RIDL
- Build an **Ω**-RIDL interpreter