Applying Ontology Design Patterns in bio-ontologies

Mikel Egaña (eganaarm@cs.man.ac.uk), Alan Rector, Robert Stevens

BioHealth Informatics Group, School of Computer Science, University of Manchester, UK

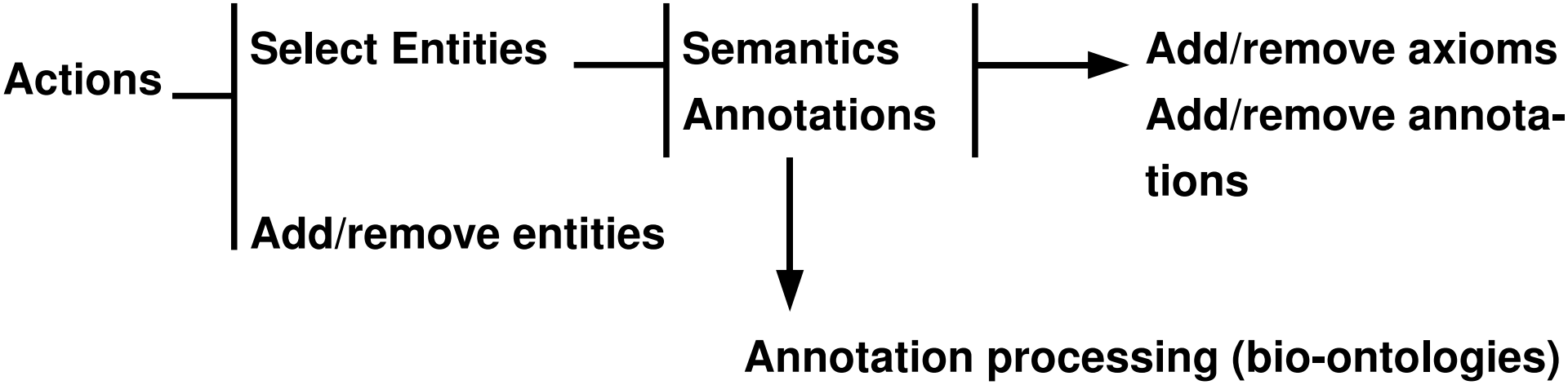
Erick Antezana

Department of Plant Systems Biology, VIB, Gent, Belgium

Department of Molecular Genetics, Gent University, Belgium

ONTOLOGY PREPROCESSOR LANGUAGE (OPPL)

High level scripting language for OWL.



Asserted/inferred mode (Pellet, FaCT++, any DIG reasoner).

Currently two versions:

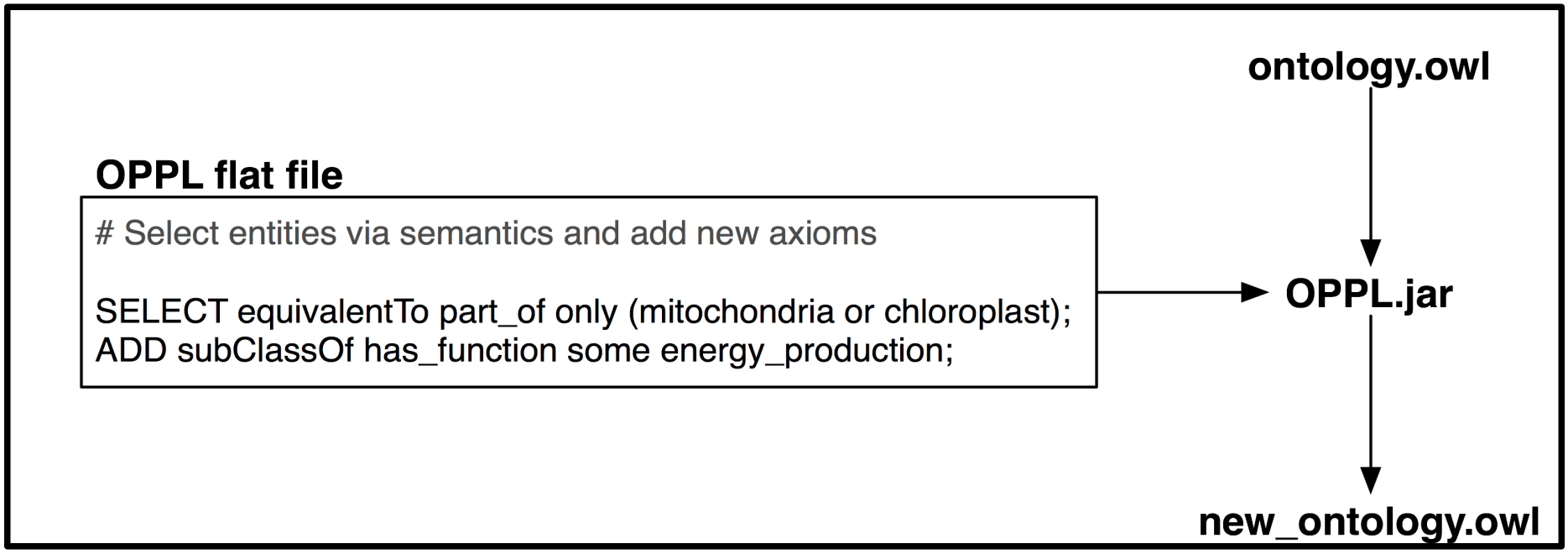
- **OPPL 1 (http://oppl.sf.net/)**
- **OPPL 2 (http://www.cs.man.ac.uk/~iannonel/oppl/)**

ONTOLOGY PREPROCESSOR LANGUAGE (OPPL)

OPPL syntax (Manchester OWL Syntax + OPPL keywords)

```
SELECT equivalentTo part_of only (mitochondria or chloroplast);
ADD subClassOf has_function some energy_production;
```

OPPL software (java)



ONTOLOGY PREPROCESSOR LANGUAGE (OPPL)

Store and share complex modelling for consistent application:

- **by different ontologists**
- **at different stages**
- **in different parts of the ontology (via queries)**

Documented and explicit modelling: trace modelling.

Try complex modelling easily, then decide: prototypes.

Ontology cleansing/enrichment.

Ontology cleansing/enrichment in pipelines (e.g. CCO http://www.cellcycleontology.org/).

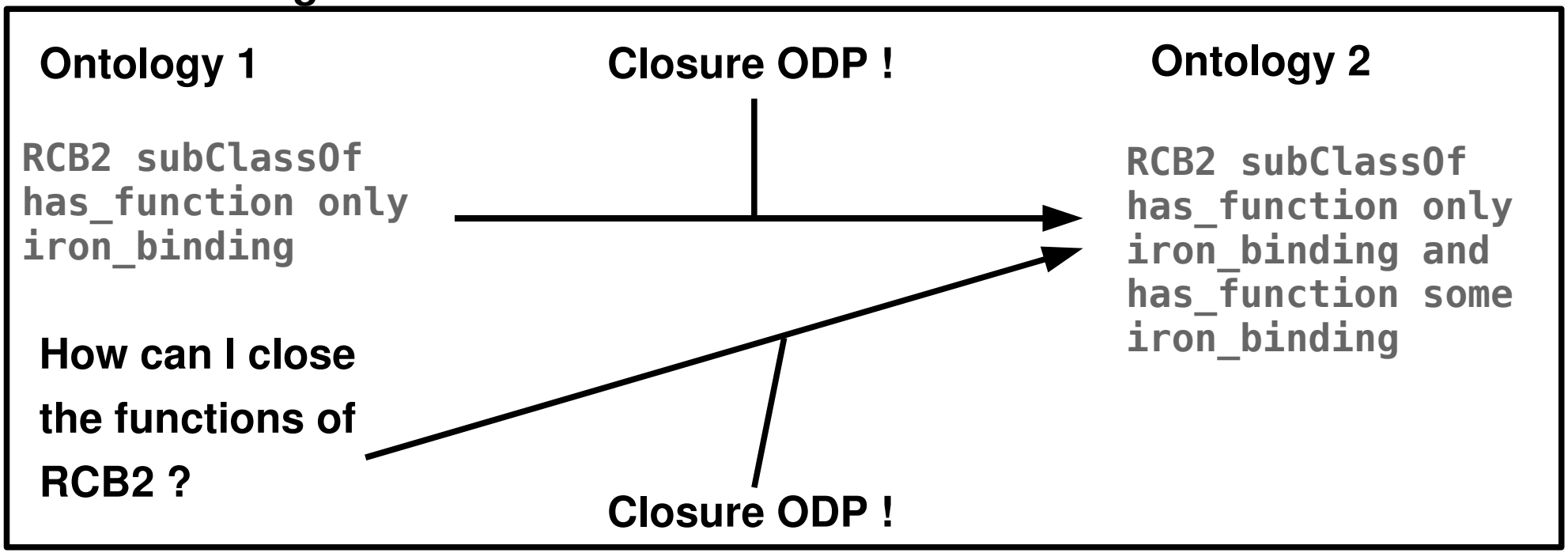
Automated modification/querying of big ontologies.

OPPL FOR ONTOLOGY DESIGN PATTERNS (ODPs)

Ontology Design Patterns (ODPs): encapsulate complex semantics, easier modelling.

e.g. Closure ODP: `prop only filler and prop some filler`

Bio-ontologies: lean axiomisation



OPPL FOR ONTOLOGY DESIGN PATTERNS (ODPs)

OPPL: store (flat files) and apply ODPs in OWL ontologies.

ODPs for modifiers	**Entity-Quality ODP (E-Q ODP)** **Entity-Property-Quality ODP (E-P-Q ODP)** **Entity-Feature-Value ODP (E-F-V ODP)**

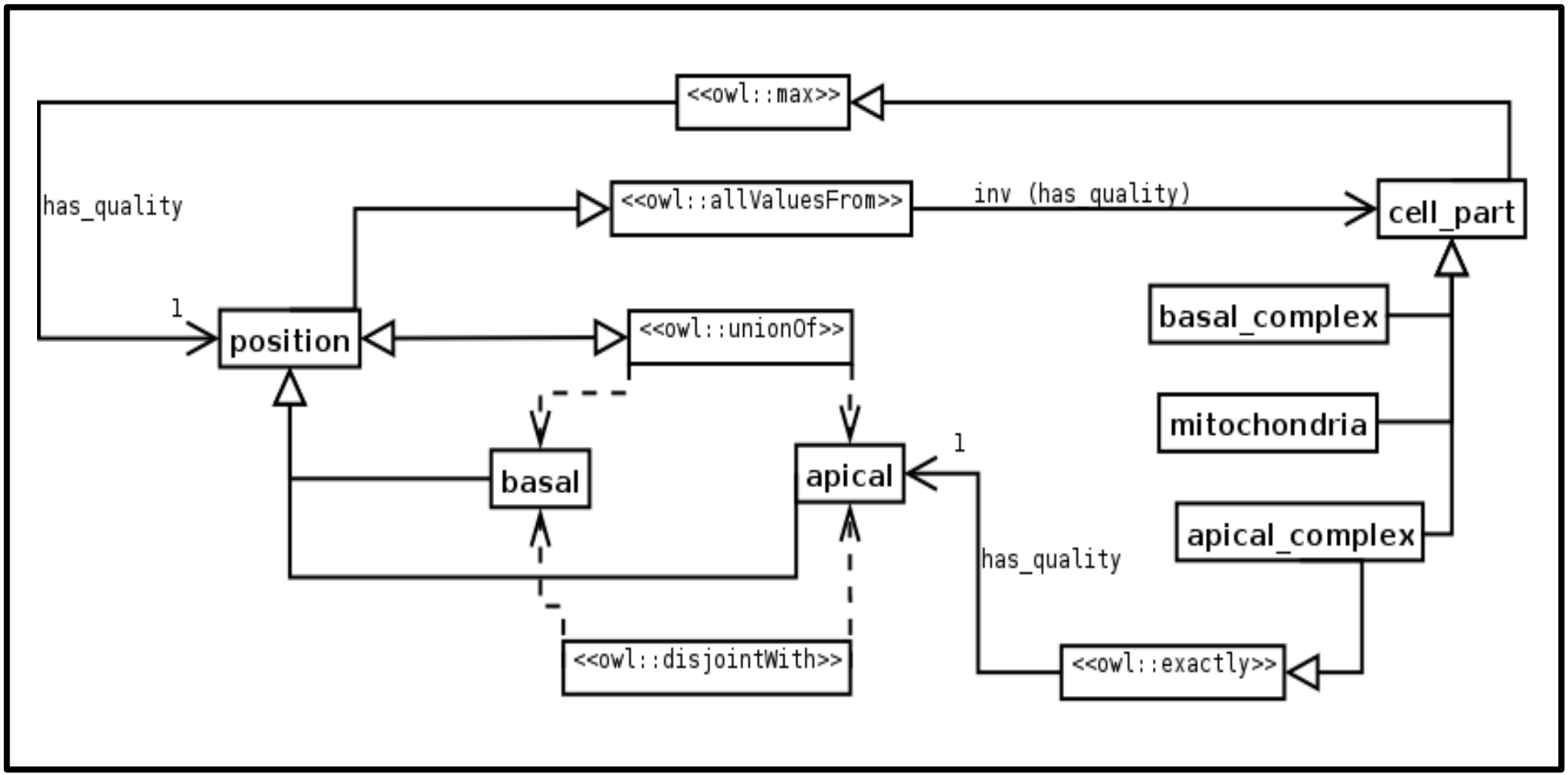
Try E-Q, E-P-Q, E-F-V, and then decide.

E-Q ODP in the Gene Ontology (GO): position of cell parts (e.g. the position of "apical complex" is the apical side of the cell).

Apply E-Q in GO via annotation query and processing with OPPL: 24/20,000.

Local vs global ODPs.

Entity-Quality ODP (E-Q ODP)



E-Q ODP applied in GO (OWL version) via OPPL script (flat file)

```
# Quality values

ADD Class: modifier;
ADD ObjectProperty: has_quality;
ADD Class: position;ADD subClassOf modifier;REMOVE subClassOf Thing;
ADD Class: apical;ADD subClassOf position;REMOVE subClassOf Thing;
ADD Class: basal;ADD subClassOf position;ADD disjointWith apical;
SELECT Class: position;ADD equivalentTo apical or basal;

# Constrain the quality values to the entities (CCO_C0001882 = cell part)

SELECT Class: position;ADD subClassOf inv (has_quality) only CCO_C0001882;

# Not having a position is legal

SELECT Class: CCO_C0001882;ADD subClassOf has_quality max 1 position;

# Add position values to actual cell parts

SELECT label "(basallapical) (.+?)";ADD subClassOf has_quality exactly 1 <1>;
```

ONTOLOGY PREPROCESSOR LANGUAGE 2

Developed by Luigi Iannone (BioHealth Informatics Group, University of Manchester).

Axiom centric, not entity centric: closer to OWL semantics.

Protégé plugin (autocomplete, ...).

Variables (e.g. Closure ODP)

```
?x:CLASS, ?z:CLASS SELECT ?x SubClassOf has_function only ?z
BEGIN ADD ?x SubClassOf has_function some ?z END;
```

Decidability: variables only to be bound by named entities, not expressions (Class, ObjectProperty, DataProperty, Individual, Constant).

CONCLUSION

OPPL: easy “programmatic” manipulation of OWL ontologies.

ODPs: semantic encapsulation; ease modelling.

OPPL for efficiently and consistently applying ODPs.

ODPs succesfully applied in the CCO with OPPL:

Mikel Egaña Aranguren, Erick Antezana, Martin Kuiper, Robert Stevens. Ontology Design Patterns for bio-ontologies: a case study on the Cell Cycle Ontology. BMC bioinformatics 2008, 9(Suppl 5):S1. http://www.biomedcentral.com/1471-2105/9/S5/S1

ACKNOWLEDGEMENTS

OPPL 1, OPPL 2:

Manchester OWL Syntax.

OWL API (http://owlapi.sf.net).

Funding:

Mikel Egaña: University of Manchester and EPSRC.

Erick Antezana: European Science Foundation (ESF), activity Frontiers of Functional Genomics.