Towards scientific principles of ontology development

Author: Mariano Fernández López

Outline

1. **Ontology and ontologies in Philosophy**
2. **Ontologies in Computer Science**
3. **Why ontology development is not a science**
4. **My contribution**

Ontology and ontologies in Philosophy

1. Ontology and ontologies in Philosophy
2. Ontologies in Computer Science
3. Why ontology development is not a science
4. My contribution

What is Ontology?

The field of reference of ontology consists of *everything there is*. Hence the field of reference of ontology is universal: we may call it “the totality of being“. But not all facts with respect to this field form part of the *subject-matter* of the ontology. It aims at cognizing those facts that are fundamentals.

Meixner U (1997) Axiomatic Formal Ontology. Kluwer Academic Publishers, Dordrecht, The Netherlands

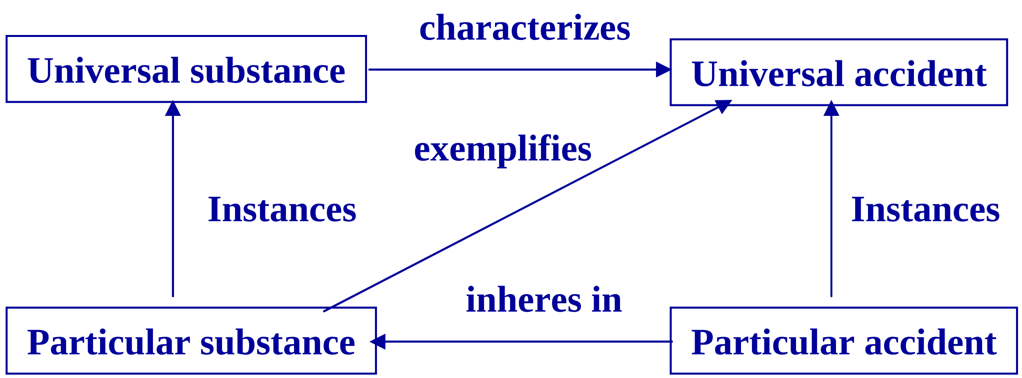
Some of the questions that the Ontology studies are:

- **Do physical objects have a common identity criterion?**
- **What does ‘essential property’ mean?**
- **What types of dependence between entities are there?**
- **Why can we say that a human body is a whole?**
- **What types of entity are there in reality?**

Ontologies in Philosophy

A classification that includes the types of entity that there are in reality is known as 'ontology'. Therefore, we have to distinguish between Ontology as a philosophical field, and 'ontology' as a result of such a field.

Guarino N (1998) Formal Ontology in Information Systems. In: Guarino N (ed) 1st International Conference on Formal Ontology in Information Systems (FOIS'98). Trento, Italy. IOS Press, Amsterdam, pp 3–15



Lowe EJ (2005) *The Four-Category ontology*. Oxford University Press, United Kingdom

Ontologies in Computer Science

1. Ontology and ontologies in Philosophy
2. **Ontologies in Computer Science**
3. Why ontology development is not a science
4. My contribution

Ontologies in Computer Science

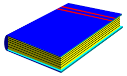
In Computer Science, an ontology is formalization with the following characteristics:

1. **It represents a part of the realitiy.**

2. **It is elaborated with the purpose of being accepted by people foreign to whom have developed such a formalization.**

3. **It is written in such a way that the computer can reason with its axioms and definitions.**

Definition provided by the author to the Real Academia de las Ciencias. It is based on:



•Gómez-Pérez A, Fernández-López M, Corcho O (2003) Ontological Engineering. Springer-Verlag, Londres

•Guarino N, Giaretta P (1995) Ontologies and Knowledge Bases: Towards a Terminological Clarification. In: Mars N (ed) Towards Very Large Knowledge Bases: Knowledge Building and Knowledge Sharing (KBKS'95). University of Twente, Enschede, The Netherlands. IOS Press, Amsterdam, The Netherlands, pp 25–32

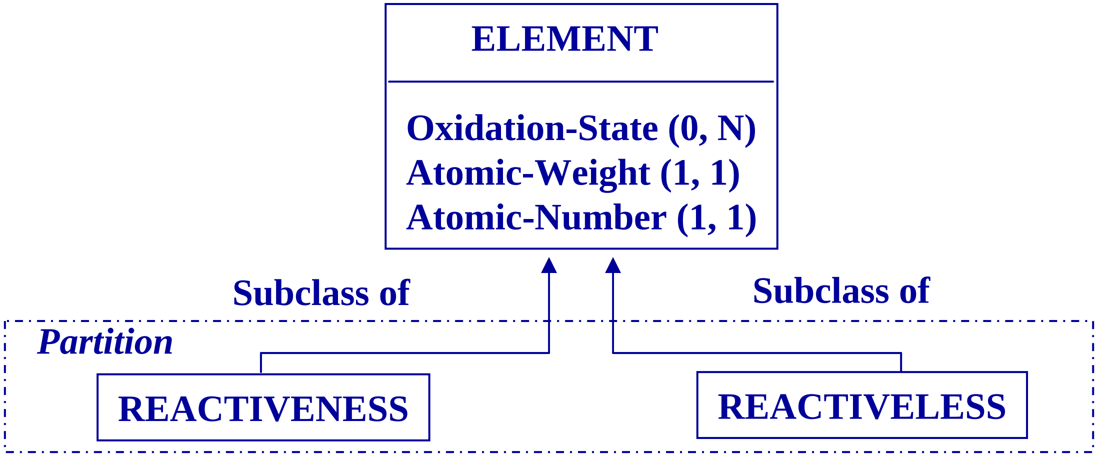
•Gruber TR (1993) A translation approach to portable ontology specification. Knowledge Acquisition 5(2):199–220

•Neches R, Fikes RE, Finin T, Gruber TR, Senator T, Swartout WR (1991) Enabling technology for knowledge sharing. AI Magazine 12(3):36–56

•Smith B (2004) Ontology. In: Floridi L (ed) Philosophy of computing and information. Blackwell Publishing, Malden, Massachusetts

•Studer R, Benjamins VR, Fensel D (1998) Knowledge Engineering: Principles and Methods. IEEE Transactions on Data and Knowledge Engineering 25(1-2):161–197

Example of ontology conceptualization (excerpt)



Fernández-López M (1996) CHEMICALS: ontología de elementos químicos. Final Project. Universidad Politécnica de Madrid, Spain

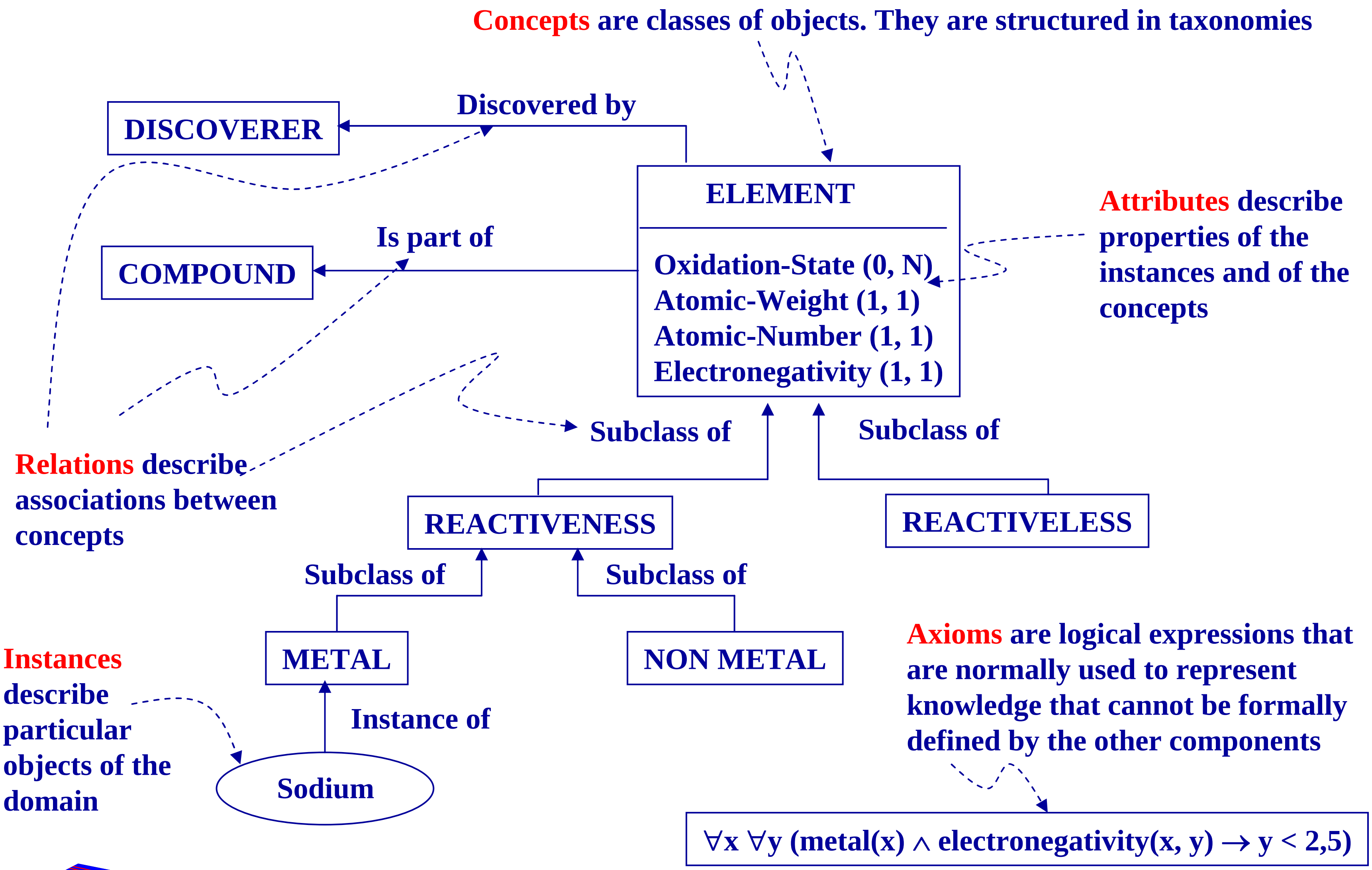
Example of ontology implementation in Prolog (excerpt)

```
class(element).
class(metal).
class(non_metal).
class(reactiveless).
class(reactiveness).
subclass_of(reactiveness,element).
superclass_of(element,reactiveness).
subclass_of(reactiveless,element).
superclass_of(element,reactiveless).
subclass_of(metal,reactiveness).
superclass_of(reactiveness,metal).
subclass_of(non_metal,reactiveness).
superclass_of(reactiveness,non_metal).
template_slot_of(atomic_number,element).
facet_of(doc,atomic_number,element).
facet_of(maxcardinality,atomic_number,element).
facet_of(mincardinality,atomic_number,element).
facet_of(valuetype,atomic_number,element).
facet_of(minvalue,atomic_number,element).
value_of_facet_of('It is the number of protons that an element
has',doc,atomic_number,element).
value_of_facet_of(1,maxcardinality,atomic_number,element).
value_of_facet_of(1,mincardinality,atomic_number,element).
value_of_facet_of(cardinal,valuetype,atomic_number,element).
value_of_facet_of(1,minvalue,atomic_number,element).
less_than(Y, 25) :- metal(X), electronegativity(X, Y).
```

Example of ontology implementation in OWL (excerpt)

```
<rdf:RDF xml:base="http://www.owl-ontologies.com/unnamed.owl">

<owl:Ontology rdf:about="">
<owl:imports rdf:resource="http://www.daml.org/rules/proposal/swrlb.owl"/>
<owl:imports rdf:resource="http://www.daml.org/rules/proposal/swrl.owl"/>
</owl:Ontology>

<owl:Class rdf:ID="reactiveness">

<rdfs:subClassOf>
<owl:Class rdf:ID="element"/>
</rdfs:subClassOf>
</owl:Class>

<owl:Class rdf:ID="reactiveless">
<rdfs:subClassOf rdf:resource="#element"/>
</owl:Class>
</rdf:RDF>
```

Components of an ontology



Gruber, T. *A translation Approach to portable ontology specifications.* **Knowledge Acquisition**. Vol. 5. 1993. 199-220.

Why ontology development is not a science

1. **Ontology and ontologies in Philosophy**
2. **Ontologies in Computer Science**
3. **Why ontology development is not a science**
4. **My contribution**

Why ontology development is not a science

A science must have a series of definitions and a series of principles. The rest of the assertions must be inferred form these definitions and principles.

OPEN PROBLEM:

THERE IS NOT A SET OF LAWS THAT RULE HOW TO DEVELOP ONTOLOGIES.

My contribution

1. Ontology and ontologies in Philosophy

2. Ontologies in Computer Science

3. Why ontology development is not a science

4. **My contribution**

My contribution

My purpose is to extract a set of laws that rule the design of domain ontologies (on particular portions of the reality).

These laws should refer ontology components: concepts, attributes, relations, etc.

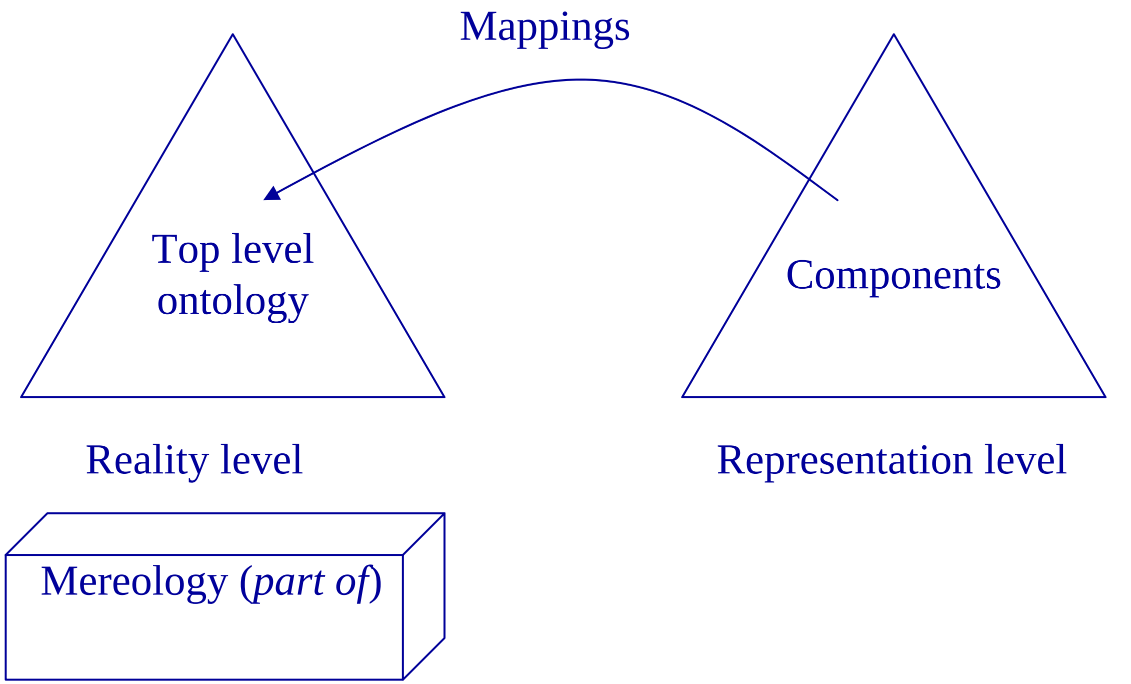
Therefore, the first step is to define such components in a precise way.

My contribution

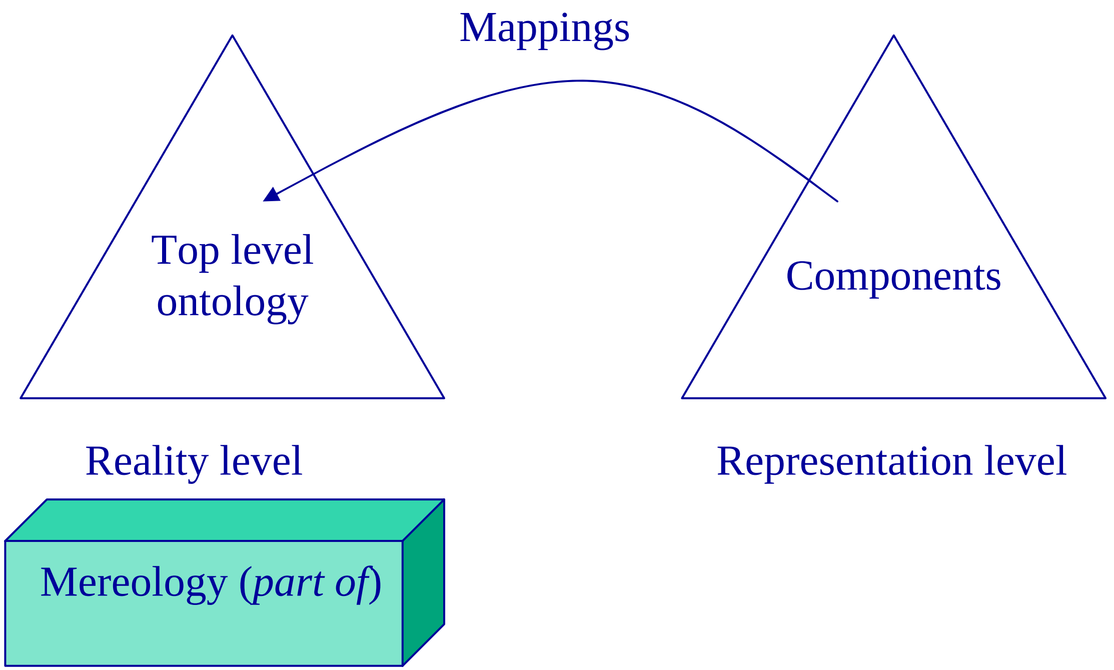
•**Software Engineering (SE) components do not have the same meaning as ontology components (e.g. identity attributes must be provided for concepts in SE; meanwhile this is not necessary in ontologies).**

•**OWL components (*concepts*, *datatype properties*, *object properties*, etc.) are interpreted in set theory. This is not very useful for ontology designer.**

My contribution



Mereology (I)



Meixner U (1997) *Axiomatic Formal Ontology*. Kluwer Academic Publishers, Dordrecht, The Netherlands

Simons P (1983) *Lesniewski's logic and its relation to classical and free logics*. In: Dorn G, Weingartner P (eds) Foundations of logic and linguistic. Problems and their solutions: a selection of contributed papers from the VIIth International congress of logic, methodology, and philosophy of science, Salzburg, Austria, 1983. Edited by Plenum Press, New York, 1985, pp 369-402. Reprinted in: Simons P (1992) Philosophy and logic in Central Europe from Bolzano to Tarski. Selected essays, Kluwer, Dordrecht, The Netherlands, pp 271-293

Mereology (II)

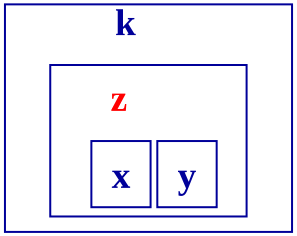
•**Part of** is a partial order (reflexive, antisymmetric and transitive).

$$\forall x\ PO(x, x) \quad (A1)$$
$$\forall x\ \forall y\ (PO(x, y) \wedge PO(y, x) \rightarrow x = y) \quad (A2)$$
$$\forall x\ \forall y\ \forall z\ (PO(x, y) \wedge PO(y, z) \rightarrow PO(x, z)) \quad (A3)$$

•**Proper part**: *x* is part of *y* and *x* is different to *y*.

$$PP(x, y) := PO(x, y) \wedge x \neq y \quad (D1)$$

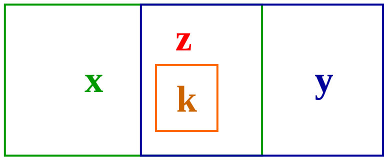
•The **conjunction** of *x* and *y* is the smallest whole of which *x* and *y* are parts.



$$x \cap y := \iota z(PO(x, z) \wedge PO(y, z) \wedge \forall k(PO(x, k) \wedge PO(y, k) \rightarrow PO(z, k))) \quad (D2)$$

Mereology (III)

- **The disjunction of *x* and *y* is the biggest part of *x* and *y*.**



$$x \cup y := \iota z(PO(z, x) \wedge PO(z, y) \wedge \forall k(PO(k, x) \wedge PO(k, y) \rightarrow PO(k, z))) \quad (D3)$$

- ***X* is minimality if and only *x* if is part of every *y*.**

$$Mi(x) := \forall y\ PO(x, y) \quad (D4)$$

Mereology (IV)

•**The complementary of *x* is that element whose only common part with *x* is a minimality.**

$$N(x) := \iota z\, (\forall y\, (PO(y, x) \wedge PO(y, z) \rightarrow Mi(y)) \wedge \forall k\, (\forall y\, (PO(y, x) \wedge PO(y, k) \rightarrow Mi(y)) \rightarrow PO(k, z))) \quad \text{(D5)}$$

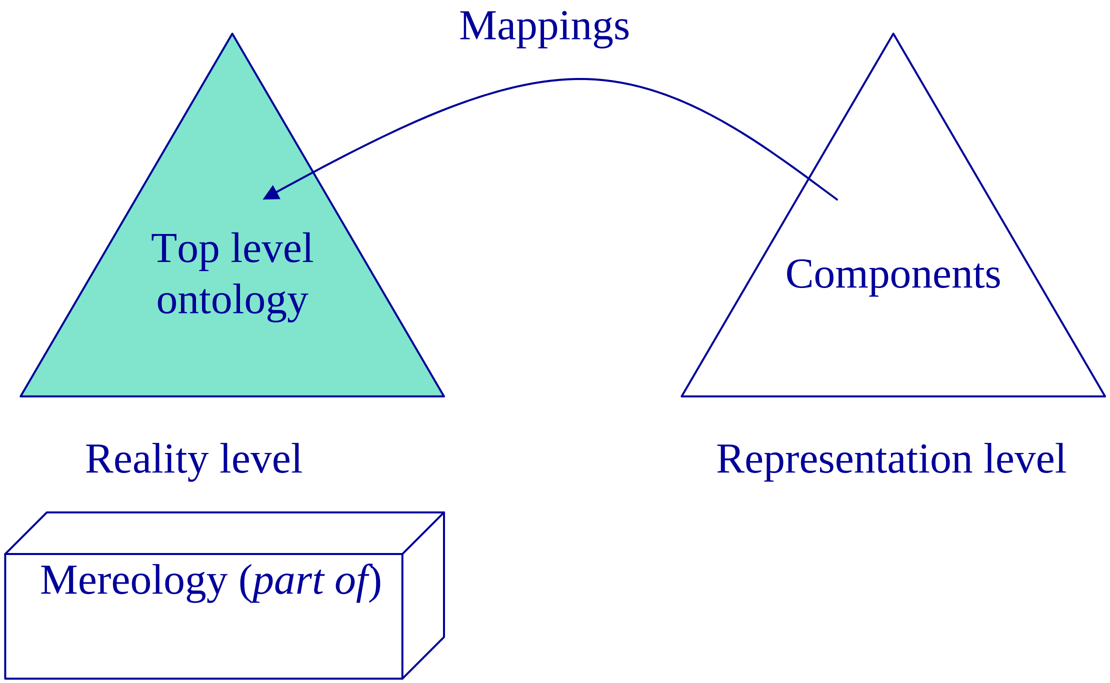
•***X* is a maximality if and only if every *y* is part of *x* or of its complementary.**

$$Mx(x) := \forall y\, (PO(y, x) \vee PO(N(y), x)) \quad \text{(D6)}$$

•***X* is consistent if and only if y such that y is part of x and its complementary does not exist.**

$$K(x) =def \neg \exists\, y(PO(y, x) \wedge PO(N(y), x)) \quad \text{(D7)}$$

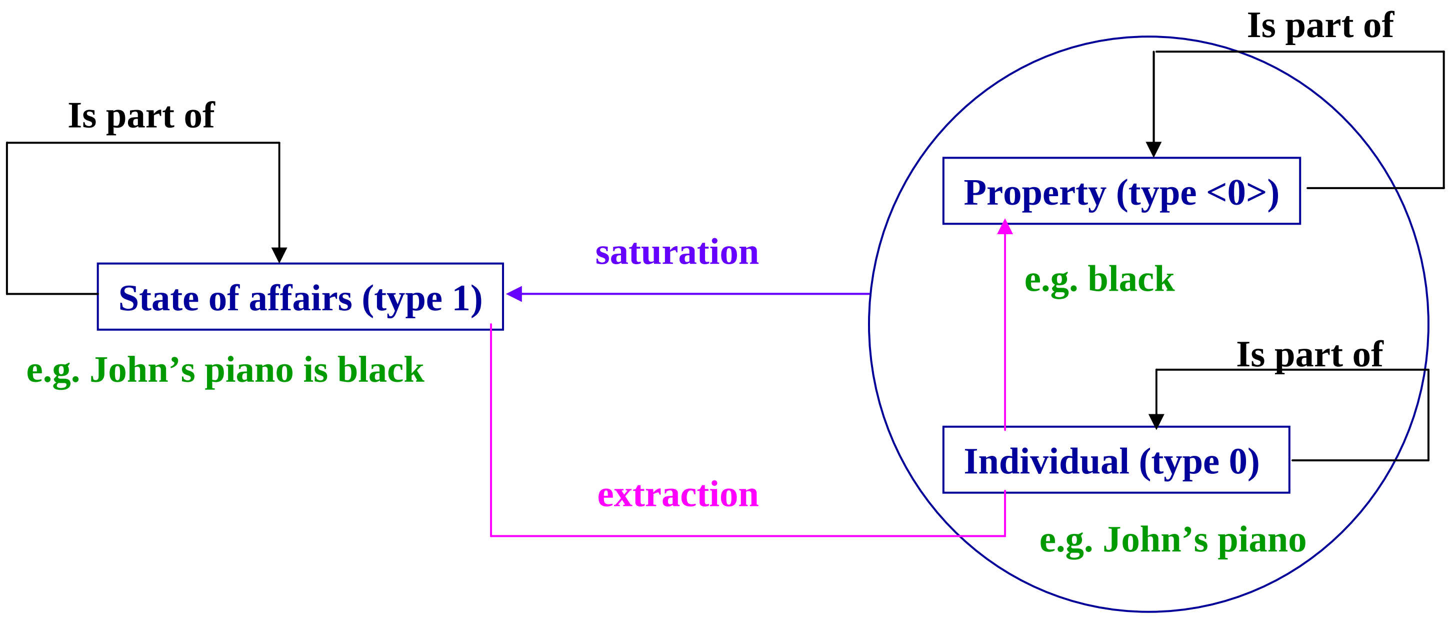
Top level ontology (I)



Meixner U (1997) *Axiomatic Formal Ontology*. Kluwer Academic Publishers, Dordrecht, The Netherlands

Simons P (1983) *Lesniewski's logic and its relation to classical and free logics*. In: Dorn G, Weingartner P (eds) Foundations of logic and linguistic. Problems and their solutions: a selection of contributed papers from the VIIth International congress of logic, methodology, and philosophy of science, Salzburg, Austria, 1983. Edited by Plenum Press, New York, 1985, pp 369-402. Reprinted in: Simons P (1992) Philosophy and logic in Central Europe from Bolzano to Tarski. Selected essays, Kluwer, Dordrecht, The Netherlands, pp 271-293

Top level ontology (II)



$\forall x\ (type1(x) \rightarrow \neg\ type0(x))$ (A4)

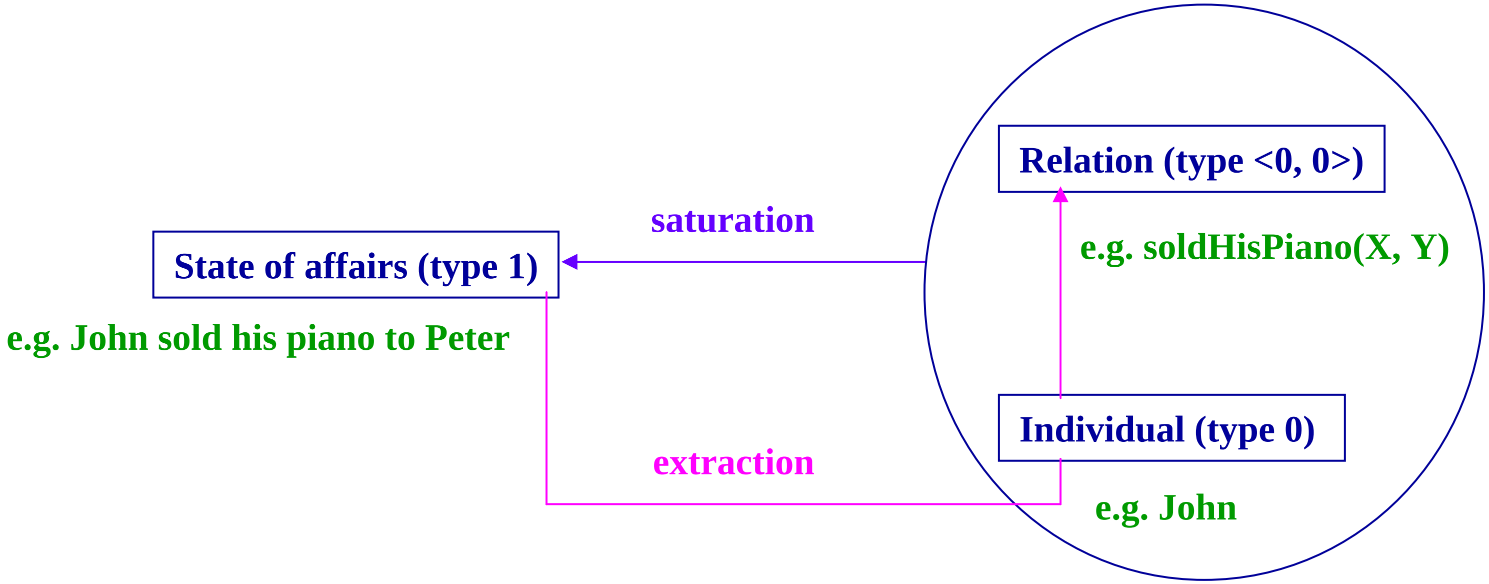
$\forall x\ (type1(x) \rightarrow \neg\ type{<}0{>}(x))$ (A5)

$\forall x\ (type0(x) \rightarrow \neg\ type{<}0{>}(x))$ (A6)

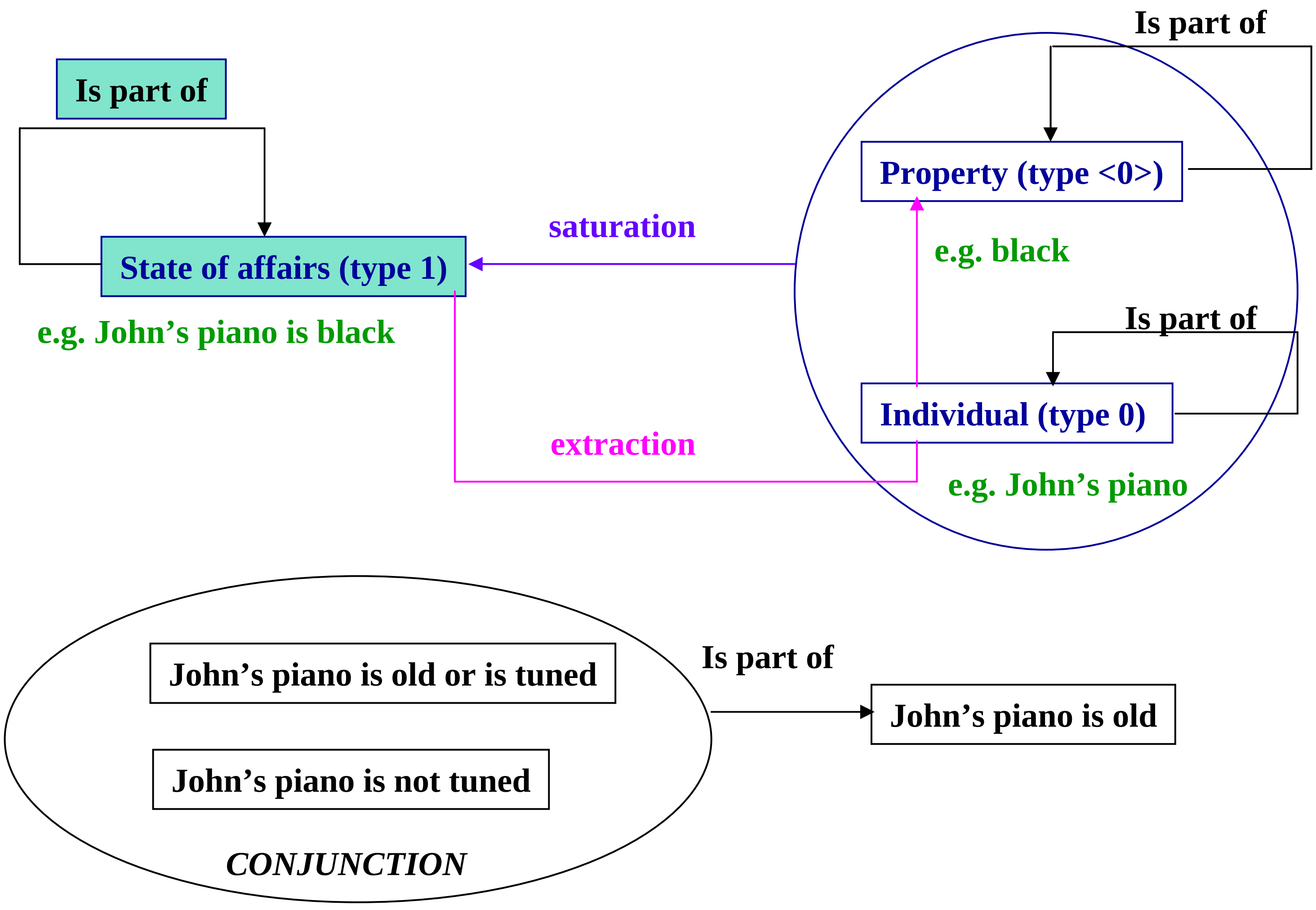
$\forall x\ \forall y\ (type{<}0{>}(x) \wedge type0(y) \rightarrow type1(Sat(x, y)))$ (A14)

$\forall t\ (type0(t) \wedge type1(f(t)) \rightarrow type{<}0{>}(Ex(t, f(t))))$ (A15)

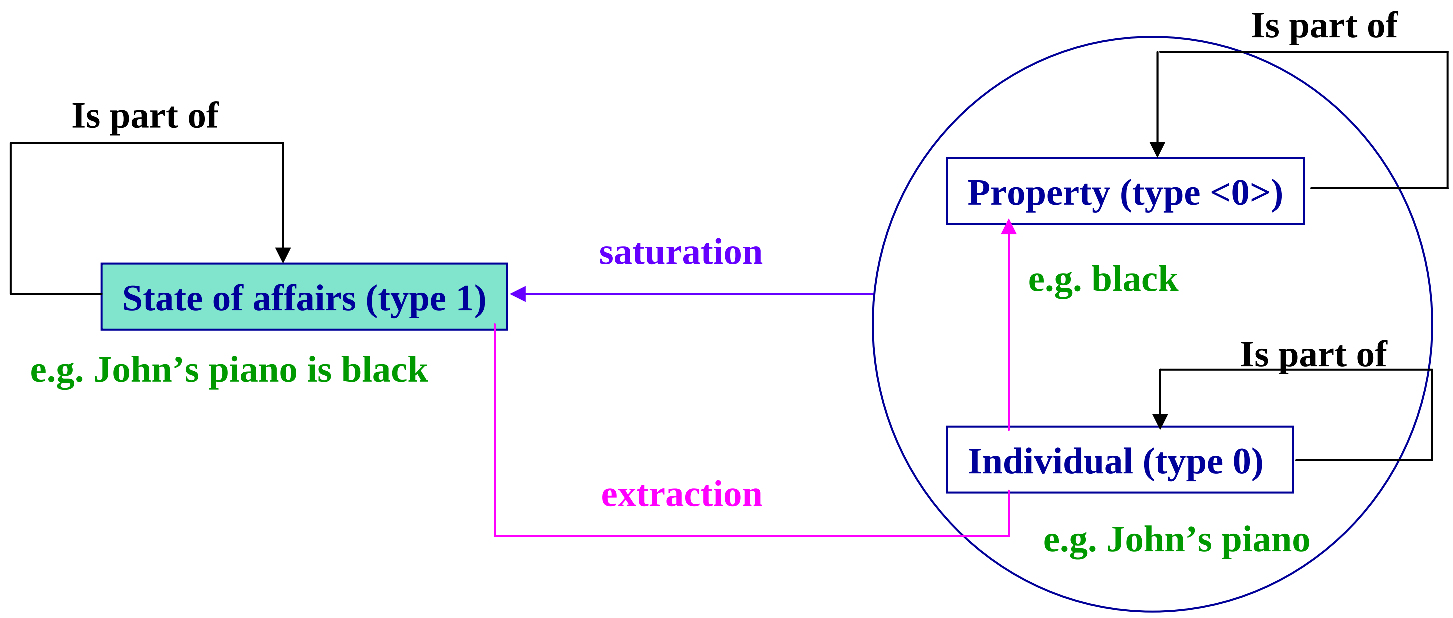
Top level ontology (III). Relations



Top level ontology (IV). States of affairs



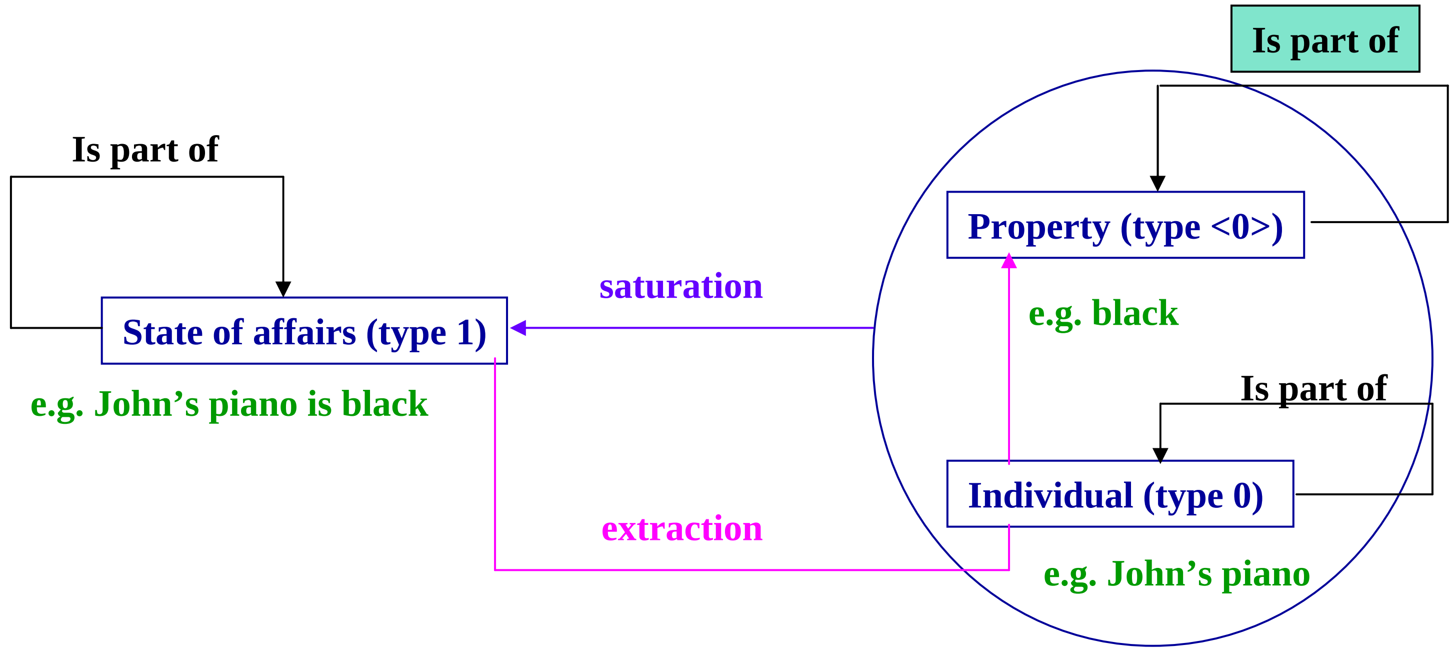
Top level ontology (V). States of affairs. The real world



The real world is a maximal and consistent state of affair

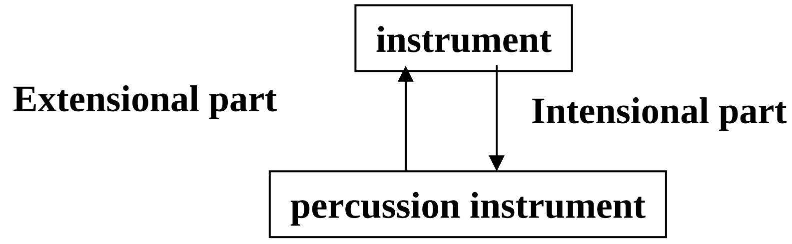
$KMx_1(x)$ =def $K_1(x) \wedge Mx_1(x)$ (D8)
$KMx_1(w)$ (A7)

Top level ontology (VI). Properties

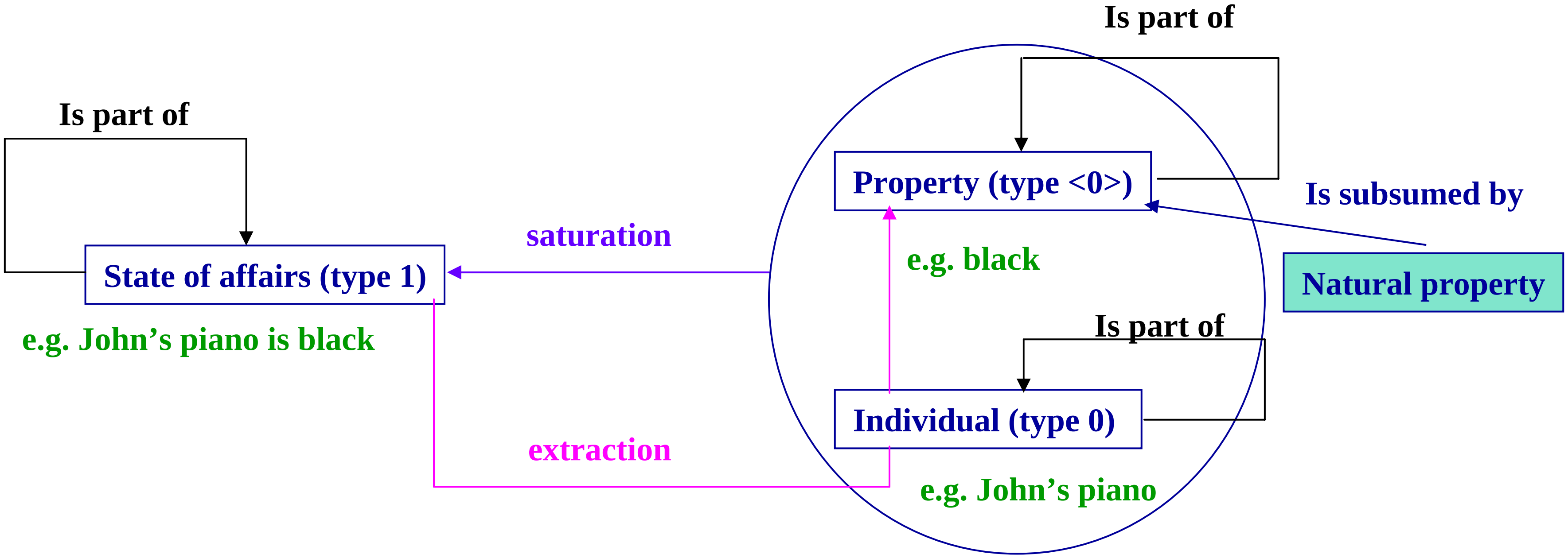


INVERSION PRINCIPLE

$\forall f \, \forall g \, (PO_{<0>i}(f, g) \leftrightarrow PO_{<0>e}(g, f))$ **(A9)**



Top level ontology (VII). Properties



A natural property is not a combination of natural properties

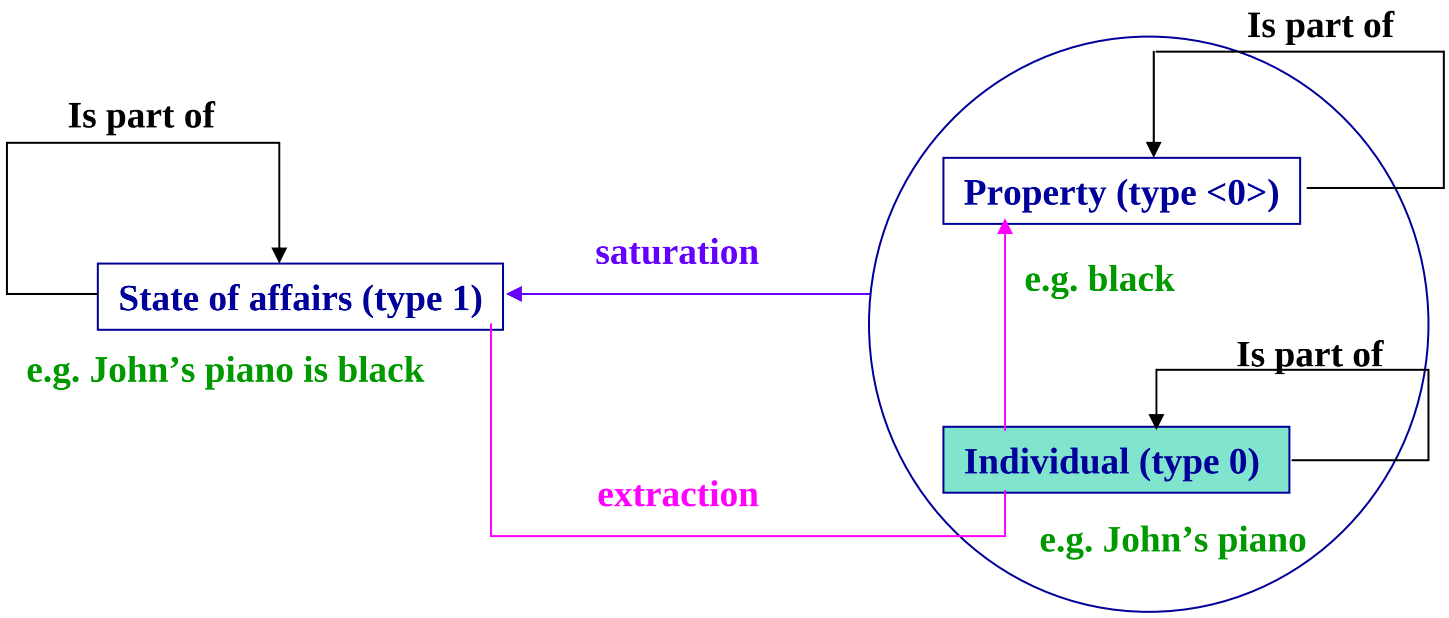
$\forall f \, (Pn(f) \rightarrow K_{<0>}(f))$ **(A10)**

$\forall x \, (Pn(f) \rightarrow \neg \, Pn(N_{<0>}i(f)))$ **(A11)**

$\forall f1 \, \forall f2 \ldots \forall fn \, (Pn(f1) \wedge Pn(f2) \wedge \ldots \wedge Pn(fn) \rightarrow$
$\neg \, Pn(f1 \cup_{<0>i} f2 \cup <0>i \ldots \cup_{<0>i} fn))$ **(A12)**

$\forall f1 \, \forall f2 \ldots \forall fn \, (Pn(f1) \wedge Pn(f2) \wedge \ldots \wedge Pn(fn) \rightarrow$
$\neg \, Pn(f1 \cap_{<0>i} f2 \cap_{<0>i} \ldots \cap_{<0>i} fn))$ **(A13)**

Top level ontology (VIII). Individuals

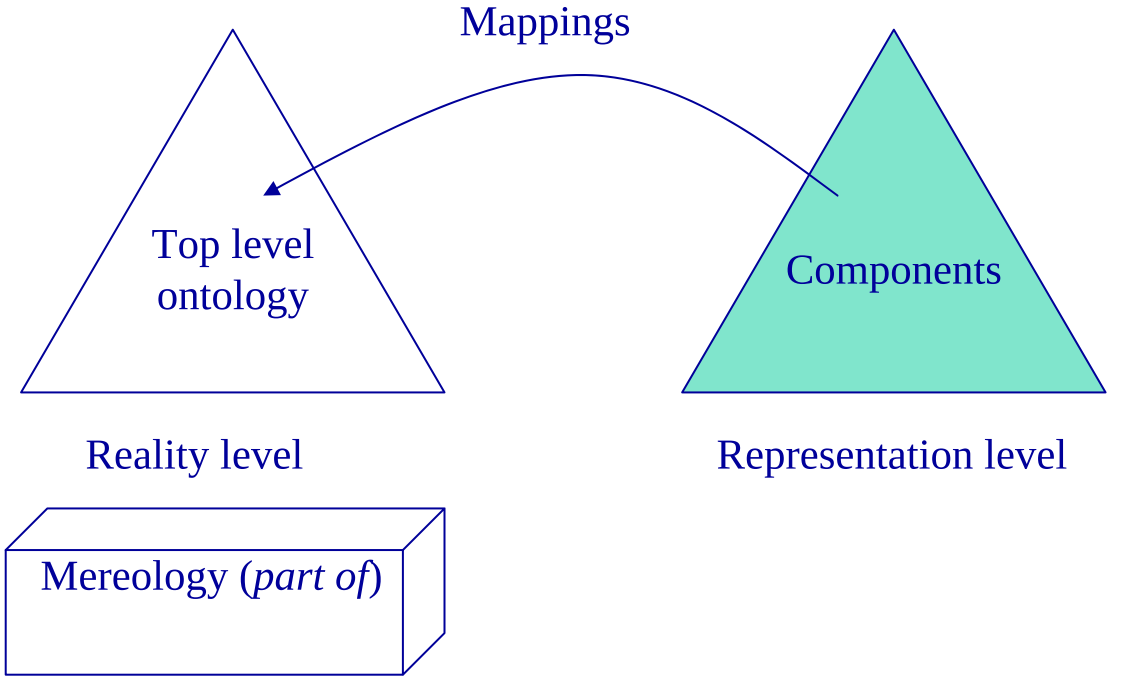


$\exists$x tipo0(x) **(A8)**

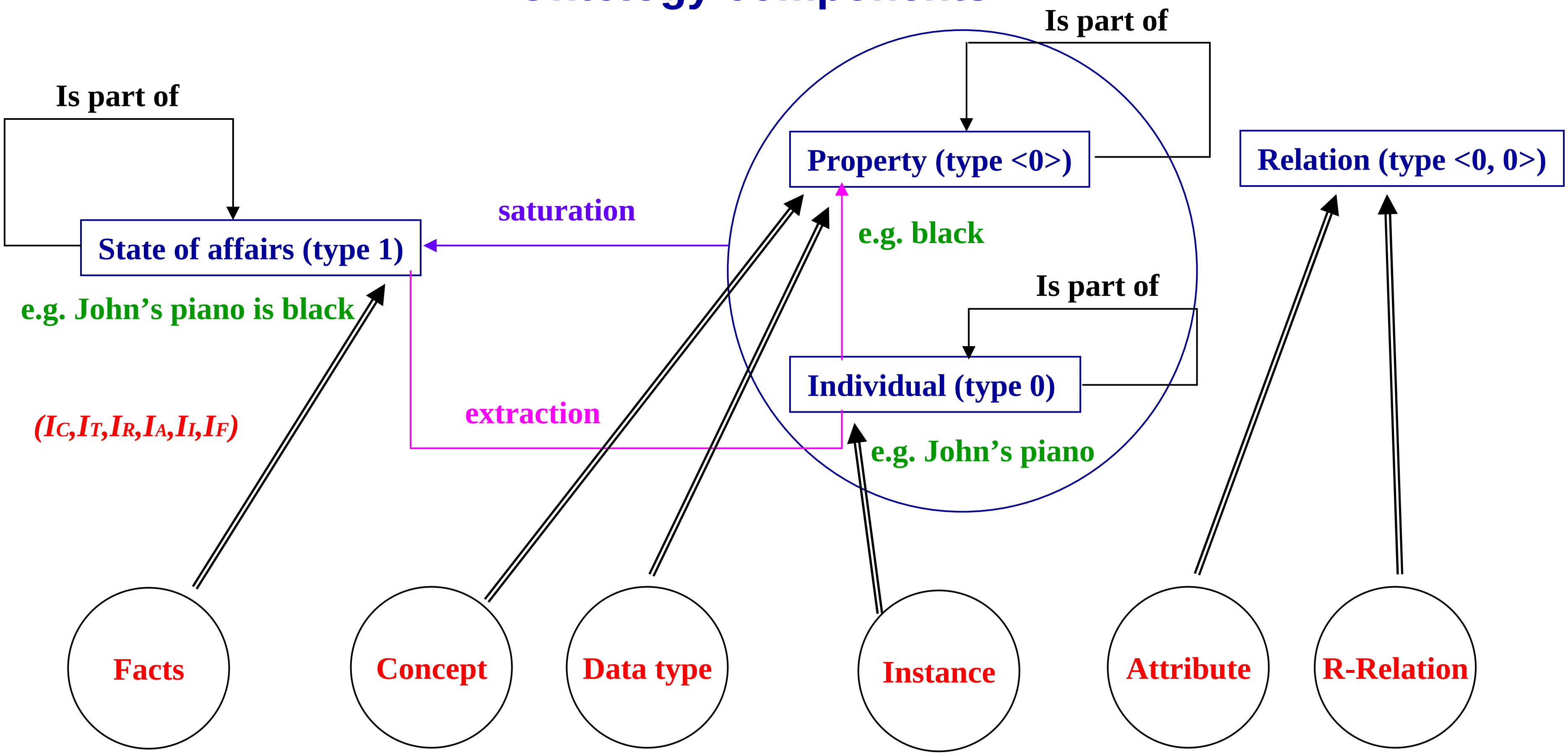
Every individual has a consistent and maximal property (John has the property 'being John')

$\forall$ t (tipo0(t) $\rightarrow$ $\exists$ f (K<0>i(f) $\wedge$ Mx<0>i(f) $\wedge$ Ej(t, f))) **(A16)**

Ontology components



Ontology components



$\forall x((x \in C) \rightarrow tipo<0>(I_C(x)))$ (A17)

$\forall x((x \in T) \rightarrow tipo<0>(I_T(x)))$ (A18)

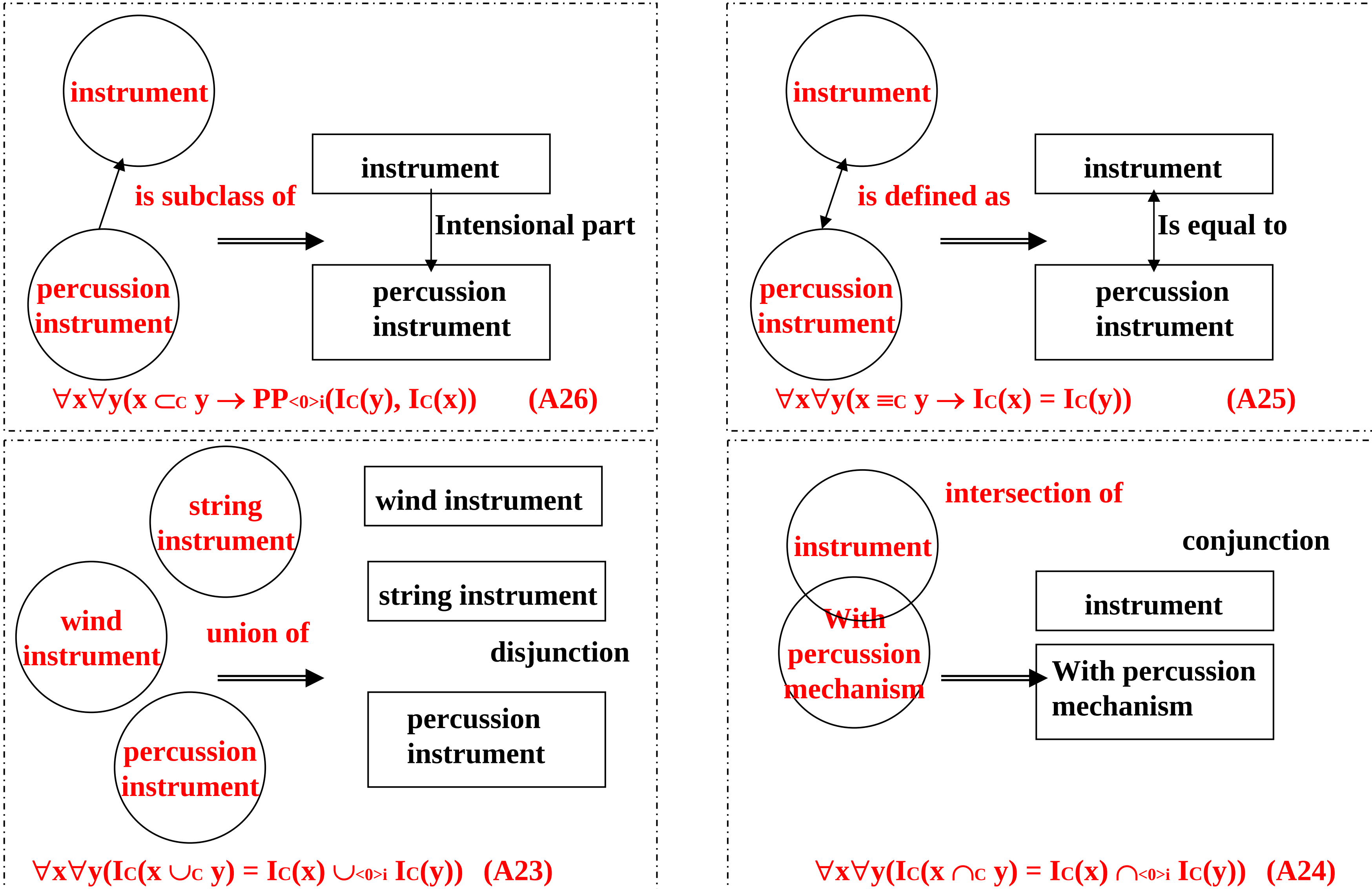
$\forall x((x \in R) \rightarrow tipo<0, 0>(I_R(x)))$ (A19)

$\forall x((x \in A) \rightarrow tipo<0, 0>(I_A(x)))$ (A20)

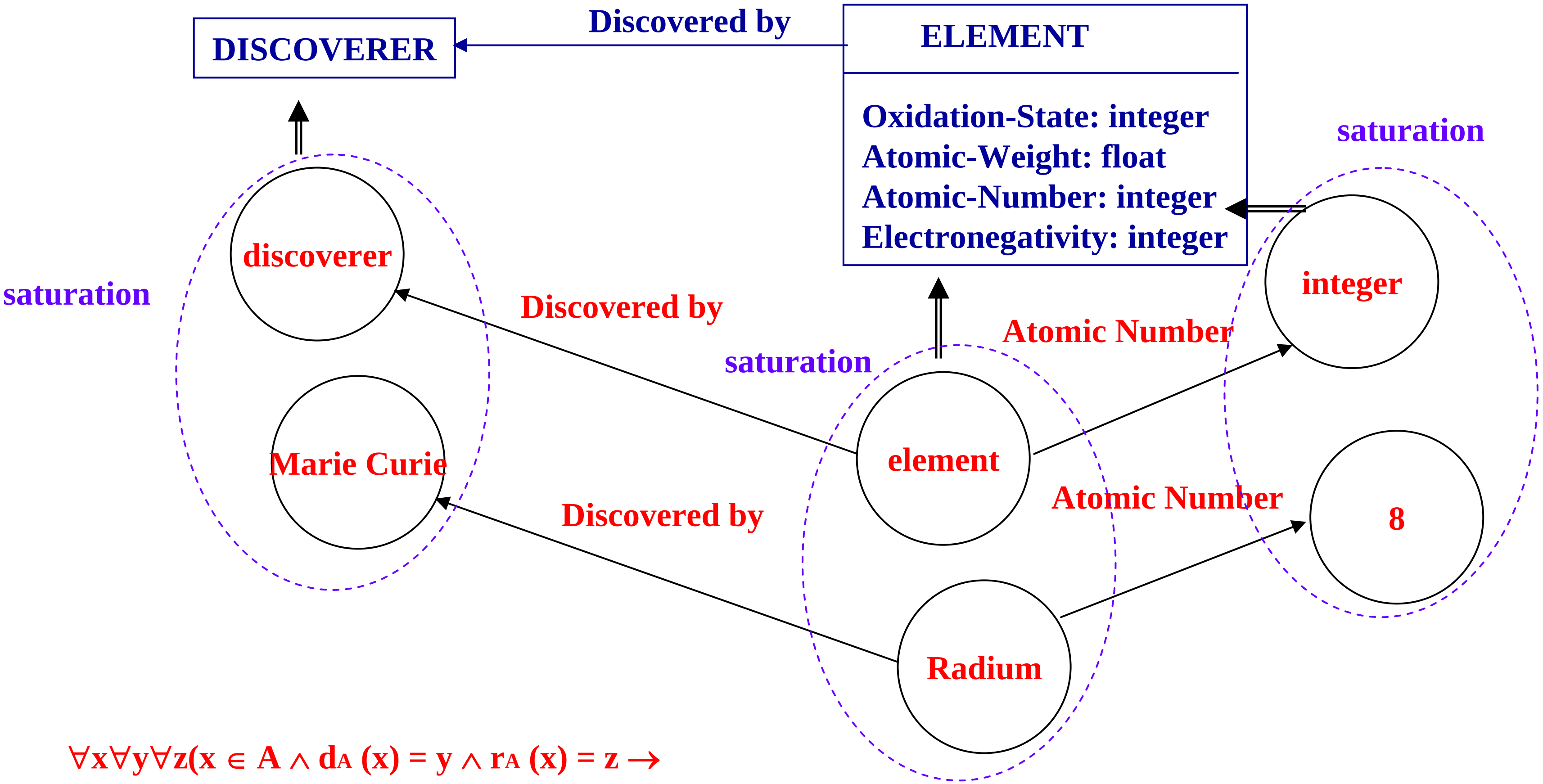
$\forall x((x \in I) \rightarrow tipo0(I_I(x)))$ (A21)

$\forall x((x \in F) \rightarrow tipo1(I_F(x)))$ (A22)

Ontology components



Ontology components



$$\forall x \forall y \forall z (x \in A \wedge d_A(x) = y \wedge r_A(x) = z \rightarrow$$
$$\exists x1\ \exists y1\ \exists z1\ \exists s1\ \exists t1\ \exists w1\ (tipo\langle 0, 0\rangle(x1) \wedge tipo\langle 0\rangle(y1) \wedge tipo\langle 0\rangle(z1) \wedge tipo1(s1) \wedge tipo0(y2) \wedge tipo0(z2) \wedge$$
$$I_A(x) = x1 \wedge I_C(y) = y1 \wedge I_T(z) = z1 \wedge Ej(y2, y1) \wedge Ej(z2, z1) \wedge s1 = Sat(x1, y2, z2))) \quad (A27)$$

$$\forall x \forall y \forall z (x \in A \wedge d_R(x) = y \wedge r_R(x) = z \rightarrow$$
$$\exists x1\ \exists y1\ \exists z1\ \exists s1\ \exists t1\ \exists w1\ (tipo\langle 0, 0\rangle(x1) \wedge tipo\langle 0\rangle(y1) \wedge tipo\langle 0\rangle(z1) \wedge tipo1(s1) \wedge tipo0(y2) \wedge tipo0(z2) \wedge$$
$$I_R(x) = x1 \wedge I_C(y) = y1 \wedge I_C(z) = z1 \wedge Ej(y2, y1) \wedge Ej(z2, z1) \wedge s1 = Sat(x1, y2, z2))) \quad (A28)$$

Conclusions

•Ontology components have been defined in a precise way. These definitions provide a precise characterization of ontology components.

•The basis for scientific principles have been established.

•The theory presented here includes the definition of the notion of ‘natural property’. It is helpful to identify concepts.

Towards scientific principles of ontology development

Author: Mariano Fernández López