Ontology and Context

I. Cafezeiro
E. H. Haeusler
A. Rademaker

April 10, 2008

Ontologies and Computer Science

- **Ontologies** describe real world things. Hierarchically organizing concepts and enriching this hierarchy with relationships among concepts.
- A real word entity to be represented is always related to a **context**. A semantically consistent body of information in which the entity **makes sense**.
- The need of contexts? In mobile **applications**, where the **environment** suffer dynamic re-configurations.

Proposal

- We propose an **algebra** for manipulate *Contextualized Ontologies* with a few basic **formal concepts** turn it **accessible**.
- We adopt: (i) an **homogeneous description** of entities and contexts and; (ii) maps that **consistently link** entities and contexts.
- **Flexibility** to: (i) combine entities or contexts in several ways and; (ii) changing and inheritance of context by an entity, and other useful operations.
- The formal approach: (i) **rigorous definition** of Contextualized Ontologies; (ii) **abstract enough** to make possible the replacement of ontologies by other **knowledge representation** technique.

Contextualized Entities

- **Entities** are described by three parts: the entity itself, a **context**, and a **link** between entity and its context.
- A triple (entity, link, context) represented by $e \longrightarrow c$, will be named **contextualized entity**.
- As both entity and context are **ontologies**, an entity can be the context of other entity.
- The context, gives general information about the entity or about the environment wherein the entity operates.
- Any context can be linked to a (meta)context.
- If the entity, or context, is represented by ontologies, we call **contextualized ontology**.

Constraints about Links

The **link** entity–context ensures the **coherence**. The context preserves the nature of the entity.

$$F : E \rightarrow C \text{ such that } F(f(e_1, e_2)) = F(f)[F(e_1), F(e_2)]$$

Constrains:

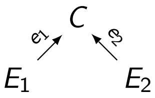
i any entity must have an **identity link**, that maps the entity to itself, and thus the entity may be viewed as a (non-informative) context of itself;

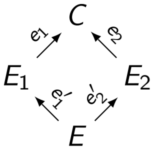
ii an entity is called **domain** of a link, while a context is called **codomain** of a link;

iii links can be composed in an **associative** way if the codomain of the first is the domain of the second. "$\circ$" denotes composition of links!

Integrating entities that share the same context

- A **semantic intersection** of contextualized entities.
- Is guided by the context (C) and results E that is also attached to that context.
- The original entities play the role of context to the produced entity. By **transitivity**, C is also a context for E.



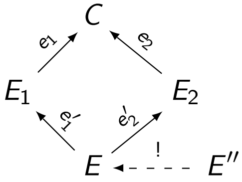


If E_1 and E_2 give different approaches about a subject C, then E express their agreement with respect to C.

Properties

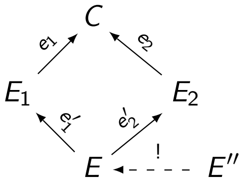
i The diagram is **commutative**. E_1, E_2 and E are coherent with respect of the context.

ii E is the **more complete** entity that makes the diagram commute. All components of E_1 and E_2 linked to the same element in C have a corresponding in E, and nothing more.



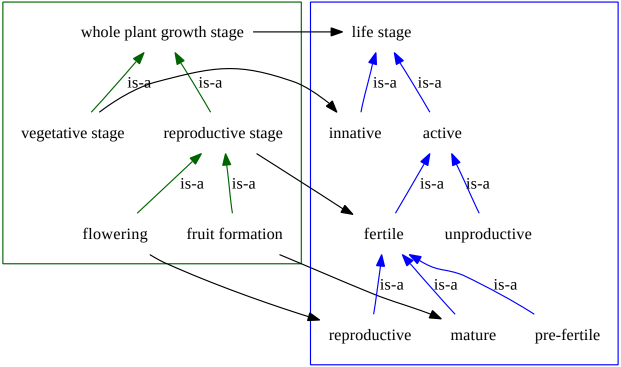
Definition: entity integration

Given two contextualized entities sharing the same context $e_1 : E_1 \longrightarrow C$ and $e_2 : E_2 \longrightarrow C$, the integration of E_1 and E_2 with respect to C is the contextualized entity $E \longrightarrow C$, such that, **(i)** There exists $e_1' : E \longrightarrow E_1$ and $e_2' : E \longrightarrow E_2$ such that $e_1 \circ e_1' = e_2 \circ e_2'$, and, **(ii)** For any other other entity E'', with links $e_1'' : E'' \longrightarrow E_1$ and $e_2'' : E'' \longrightarrow E_2$ there exists a unique link $! : E'' \longrightarrow E$ with $e_1' \circ ! = e_1''$ and $e_2' \circ ! = e_2''$



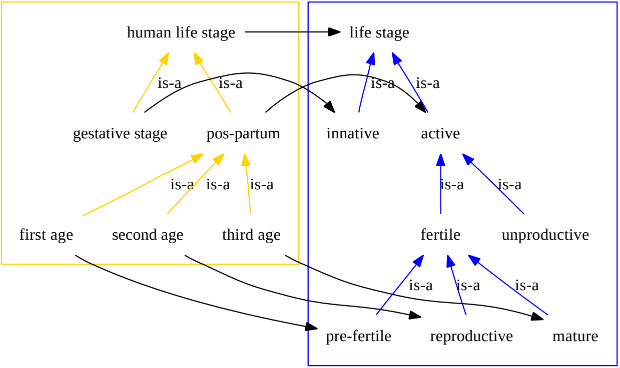
Example 1/3: E_1 and C

Whole Plant Growth Stage contextualized by Life Stage:



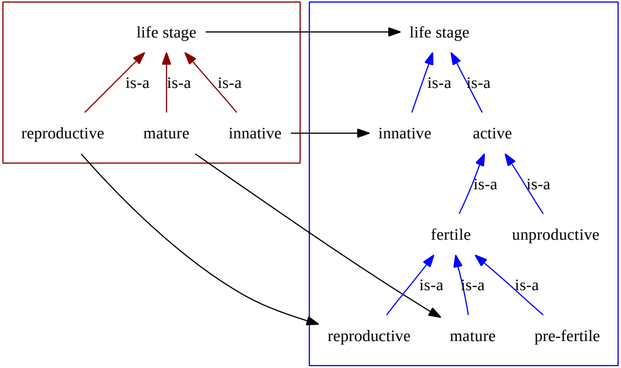
Example 2/3: E_2 and C

Human contextualized by Life Stages:



Example 3/3: E and C

The semantic intersection of E_1 and E_2 guided by C:



Example comments

- Both are contextualized by an ontology that describes life stages.
- The result embodies the semantic intersection of "plant" and "human".
- Both plant and human ontologies could also be viewed as context to the resulting entity.
- By (*i*), components of the entity E correspond to those of "plant" and "human" that are linked to the same component of "life stage".
- By (*ii*), *all* the components that satisfies (*i*) are present in E.

The algorithm

Algorithm. (Entity Integration)

Input: $e_1 : E_1 \to C$ and $e_2 : E_2 \to C$ Output: $E \to C$

Notation: x_i are variables for concepts of entities and y_i are variables for relations of entities. (C_E, R_E, H_E^C, rel_E) identify the components of an entity E. f_e is component f of a link e and g_e is component g of a link e. The symbol $\mapsto$ denotes the association by a function of the element at the left to the element at the right of the symbol $\mapsto$.

Initial conditions: C_E, R_E are empty sets and $f_{e'_1}, f_{e'_2}, g_{e'_1}, g_{e'_2}$ are empty functions.

For all $x_1 \in C_{E_1}$
  If there is $x_2 \in C_{E_2}$ with $f_{e_1}(x_1) = f_{e_2}(x_2)$
    $C_E := C_E \cup f_{e_1}(x_1)$
    $f_{e'_1} := f_{e'_1} \cup (f_{e_1}(x_1) \in C_E) \mapsto x_1$
    $f_{e'_2} := f_{e'_2} \cup (f_{e_2}(x_2) \in C_E) \mapsto x_2$
For all $y_1 \in R_{E_1}$
  If there is $y_2 \in R_{E_2}$ with $g_{e_1}(y_1) = g_{e_2}(y_2)$
    $R_E := R_E \cup g_{e_1}(y_1)$
    $g_{e'_1} := g_{e'_1} \cup (g_{e_1}(y_1) \in C_E) \mapsto y_1$
    $g_{e'_2} := g_{e'_2} \cup (g_{e_2}(y_2) \in C_E) \mapsto y_2$
return $(f_{e_1}, g_{e_1}) \circ (f_{e'_1}, g_{e'_1})$

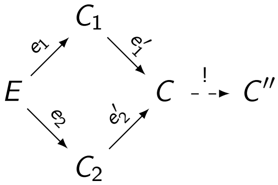
A summation (amalgamation) of contexts

- A single entity E can be viewed in different ways, C_1 and C_2.
- A new context as a result of combining and integrating given contexts.
- The resulting context must be **coherent** with respect to the corresponding entity.
- **All** components of the original contexts will be represent in C, resulting links have the original contexts as domain.

Properties

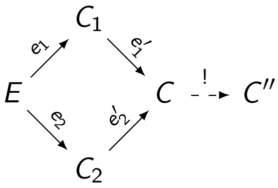
i The diagram commutes, so C is a coherent sum with respect to the entity E.

ii C is the *less informative* context that makes the diagram commute. All elements of C_1 and C_2 are represent in C, and nothing more.



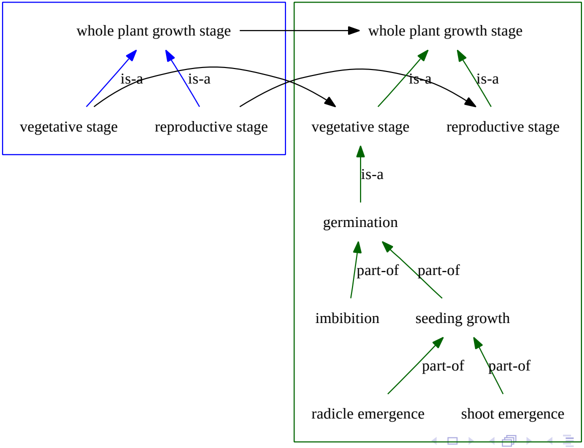
Definition: context integration

Given two contextualizations of the same entity $e_1 : E \longrightarrow C_1$ and $e_2 : E \longrightarrow C_2$, the context integration of C_1 and C_2 with respect to E is the contextualized entity $E \longrightarrow C$, such that, **(i)** There exists $e_1' : C_1 \longrightarrow C$ and $e_2' : C_2 \longrightarrow C$ such that $e_1' \circ e_1 = e_2' \circ e_2$, and, **(ii)** For any other other context C'', with maps $e_1'' : C_1 \longrightarrow C''$ and $e_2'' : C_2 \longrightarrow C''$ there exists a unique map $! : C \longrightarrow C''$ with $! \circ e_1' = e_1''$ and $! \circ e_2' = e_2''$.



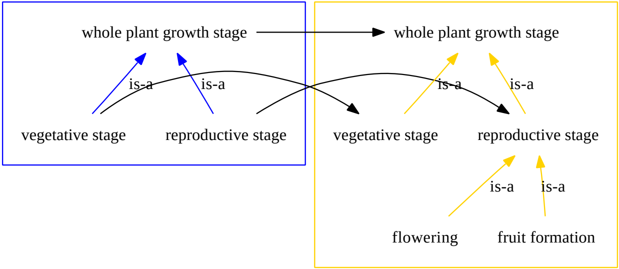
Example 1/3: E and C_1

Part of *Whole Plant Growth Stage* ontology:



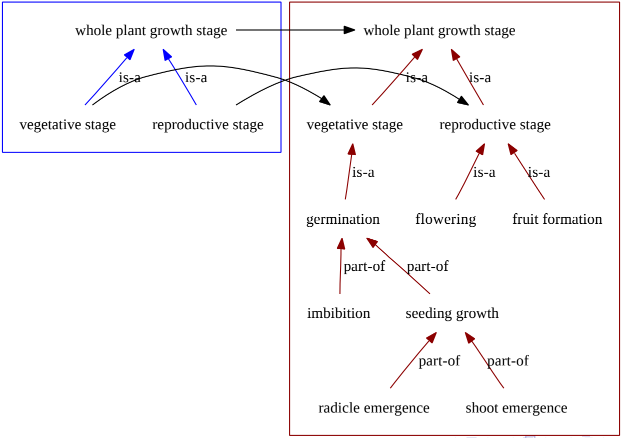
Example 2/3: E and C_2

Another part of *Whole Plant Growth Stage* ontology:



Example 3/3: E and C

Ammalgamation of entities of the ontologies:



Example comments

- A **plant growth stage** ontology is composed by two separate parts: vegetative stage and reproductive stage.
- These parts can be developed in separate and **glued** later to form the complete plant growth stage.
- In the glue process part of the ontology (the glue points) must be **contextualized** by the ontologies containing the new parts.

The integration of these contexts results the whole plant growth stage ontology.

The algorithm

Algorithm. (Context Integration)

Input: $e_1 : E \to C_1$ and $e_2 : E \to C_2$ Output: $E \to C$

Notation: similar of Entity Integration.

Initial conditions: C_E is the empty set and $f_{e'_1}, f_{e'_2}$ are empty functions.

(i) For all $x \in C_E$

$C_C := C_C \cup x$

$f_{e'_1} := f_{e'_1} \cup (f_{e_1}(x) \in C_{C_1}) \mapsto (x \in C_C)$

$f_{e'_2} := f_{e'_2} \cup (f_{e_2}(x) \in C_{C_2}) \mapsto (x \in C_C)$

(ii) For all $x \in C_{E_1}$ that is not in the image of f_{e_1}

$C_C := C_C \cup x$

$f_{e'_1} := f_{e'_1} \cup (x \in C_{C_1}) \mapsto (x \in C_C)$

(iii) For all $x \in C_{E_2}$ that is not in the image of f_{e_2}

$C_C := C_C \cup x$

$f_{e'_2} := f_{e'_2} \cup (x \in C_{C_2}) \mapsto (x \in C_C)$

return $f_{e'_1} \circ f_{e_1}$

Morphism between Contextualized Entities

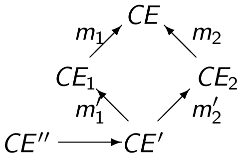
How to operate contextualized entities as a whole (entity and context)? The commutative square:

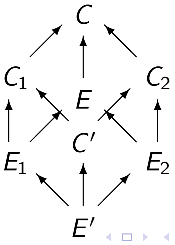
$$\begin{array}{ccc} C_1 & \xrightarrow{c} & C_2 \\ {\scriptstyle e_1}\uparrow & & \uparrow{\scriptstyle e_2} \\ E_1 & \xrightarrow{e'} & E_2 \end{array}$$

Given two contextualized entities $E_1 \xrightarrow{e_1} C_1$ and $E_2 \xrightarrow{e_2} C_2$, a pair contextualized entities $(C_1 \xrightarrow{c} C_2, E_1 \xrightarrow{e'} E_2)$ is a map from e_1 to e_2 if $E_1 \xrightarrow{c \circ e_1} C_2 = E_1 \xrightarrow{e_2 \circ e'} C_2$ is a contextualized entity.

The Relative Intersection

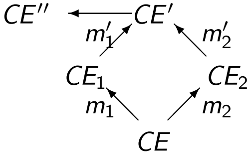
- **Commonalities** among entities with different contexts.
- **Coherent** intersection of two given contextualized entities with respect to a third one.
- CE' is the **more informative** contextualized entity, coherent with CE_1 and CE_2 with respect to CE.
- All the lateral squares of the right cube commute (by def). The bottom and top squares of the cube also commute.

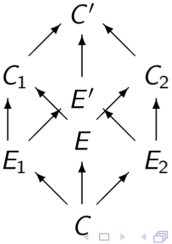




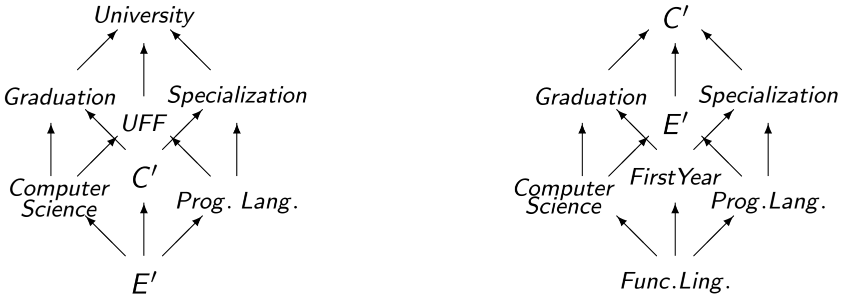
The Collapsing Union

- Acts in context and entity of CE_1 and CE_2. Results the union of them, possibly collapsing some components.
- Any concept in CE_1 or CE_2 is mapped a component in CE'. The concepts of CE are mapped to the same concept of CE' via links through CE_1 or CE_2.
- CE' is the less informative *Cont.Ent.* coherent with CE_1 and CE_2.





Examples


University
Graduation
Specialization
UFF
Computer Science
C'
Prog. Lang.
E'
C'
Graduation
Specialization
E'
FirstYear
Computer Science
Prog.Lang.
Func.Ling.

Category Theory: what is it? Why use it?

- The presented algebra is an application of Category Theory;
- "Thing" (**objects**) described abstractly by their interactions; Focus in the relationship (**morphisms**);
- Functors relates categories, co-existence of **heterogeneous** "things";
- Successfully used where interoperability is a crucial point;

A Category

A category $\mathcal{C}$ is a structure:

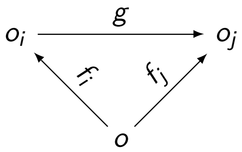
$$\mathcal{C} = (O, M, Dom, Cod, \circ, id)$$

O collection of objects; M morphisms $f : A \longrightarrow B$ where $A, B \in O$; $Dom, Cod : M \longrightarrow O$; $\circ$ associative operation of morphisms composition; id morphisms id_A for each object $A \in O$.

Diagrams and Cones

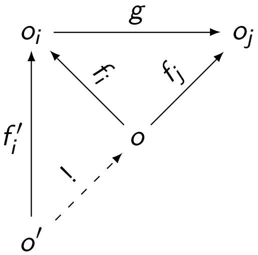
- A category can be pictured as graphs (diagrams) for reasoning.
- Some definitions can be easily obtained by just reversing "arrows".

Cone $\{f_i : o \rightarrow o_i\}$: for any $g : o_i \rightarrow o_j$ we have $g \circ f_i = f_j$



Limits

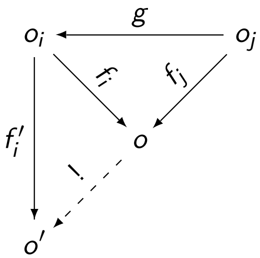
A limit for a diagram D with objects o_i is a cone $\{f_i : o \longrightarrow o_i\}$ such that for any other cone $\{f'_i : o' \longrightarrow o_i\}$, for D, there is a unique morphism $! : o' \longrightarrow o$ for which $f_i \circ ! = f'_i$ with $f'_i : o' \longrightarrow o_i$.



Special cases of D: two single objects is **product**; $o_1 \longrightarrow o \longleftarrow o_2$ is **pullback** (product guide by o).

Colimits: duality of limits

A colimit for a diagram D with objects o_i is a **cocone** $\{f_i : o_i \longrightarrow o\}$ such that for any other **cocone** $\{f_i' : o_i \longrightarrow o'\}$, for D, there is a unique morphism $! : o \longrightarrow o'$ for which $! \circ f_i = f_i'$ with $f_i' : o_i \longrightarrow o'$.



Special cases of D: two single objects is **coproduct**; $o_1 \longleftarrow o \longrightarrow o_2$ is **pushout** (sum of o_1 and o_2 possibly collapsing according to o).

An Ontology

An ontology $\mathcal{O}$ is a structure:

$$\mathcal{O} = (C, R, H^C, rel, A)$$

Concepts, relations, $H^C \subseteq C \times C$ hierarchy of concepts (taxonomic relation), $rel : R \rightarrow C \times C$ relates concepts non-taxonomically and Axioms.
$(x_1, x_0) \in H^c$ means x_1 is subconcept of x_0.

Ontology operations

Given two ontologies o_1 and o_2:

mapping total mapping between o_1 and o_2 which preserves hierarchy, conceptual relations and specify semantic overlap between them.

alignment is the task of establishing a collection of binary relations between vocabularies of o_1 and o_2. A pair of total functions (ontology mappings) from a intermediate o.

merging unification of o_1 and o_2 into a new one that embodies the semantic differences and collapses the semantic intersection.

matching finding commonalities between ontologies.

hiding erasing a concept/relation preserving hierarchy, conceptual relations and semantic relations.

Ontologies in a Categorical view

The category *Ont* of ontologies:

- objects are ontology structures;
- morphisms pairs of functions $(f, g) : O \longrightarrow O'$ where $O = (C, R, H^C, rel)$ and $O' = (C', R', H^{C'}, rel')$ and $f : C \longrightarrow C'$ and $g : R \longrightarrow R'$ such that:
    - i if $(c_1, c_2) \in H^C$ then $(f(c_1), f(c_2)) \in H^{c'}$, and
    - ii if $(c_1, c_2) \in rel(r)$ then $(f(c_1), f(c_2)) \in rel'(g(r))$.

(f, g) are links!

Ontologies in a Categorical view

- A contextualized entity is an object of $Ont^{\rightarrow}$ where objects are morphisms of Ont, morphisms are pairs of Ont morphisms (m, m');
- **Entity integration** is a **pullback** in Ont (matching);
- **Context integration** is a **pushout** in Ont (merging);
- Combined operations are performed in $Ont^{\rightarrow}$.
- Proofs that operations presented are well defined;

Conclusion

- Contexts are essential to clarify the meaning of entities;
- **Uniform representation** of entities and contexts and the **compositional definitions** of operations give us **abstraction**, **modularity** and **reuse**;
- The role of an object (entity or context) is given by the **net** of links **from** or **to** it;
- Expansive, specificity, explicit, separated and transparent (from Roman, Julien & Payton "Formal Treatment of Context Awareness", 2004);
- . . . **Interoperability** of heterogeneous descriptions (different kinds of categories).

Future works

Contextualizing web queries: a **query** is a morphism in *Ont* where *dom* ontology is the information to be search (O) and *cod* is the context (O_1).

$$
\begin{array}{l}
\textit{for all } O_2 \in \textit{search}(O) \\
\quad \textit{if } \iota : O \rightarrow O_2 \in \textit{Morphisms}(\textit{Ont}) \\
\quad\quad \textit{Results} = \textit{Results} \cup \textit{pushout}(O_1 \leftarrow O \hookrightarrow O_2) \\
\textit{return Results}
\end{array}
$$

Institutions: model-theoretical and syntactic mechanism from the operations on the structural level.