

Evolution of Ontologies and Types

Thierry Despeyroux

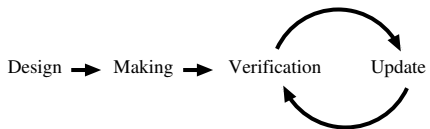
Inria, Paris-Rocquencourt

IADIS WWW/Internet 2008 Conference

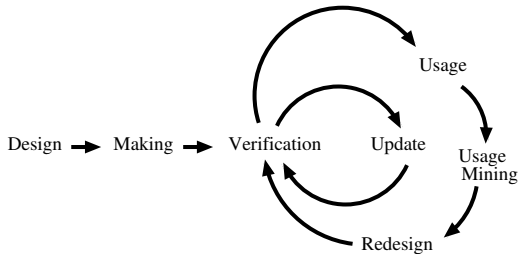
Au menu...

- 1 Information Systems Lifetime
- 2 Ontologies Evolution
- 3 Ontologies as Types
- 4 Conclusion

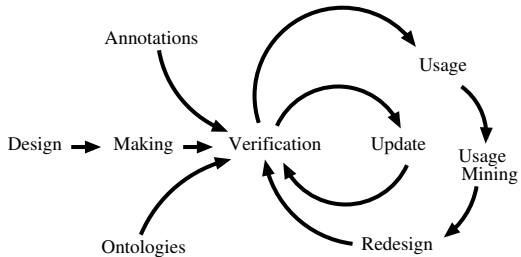
Basic concept



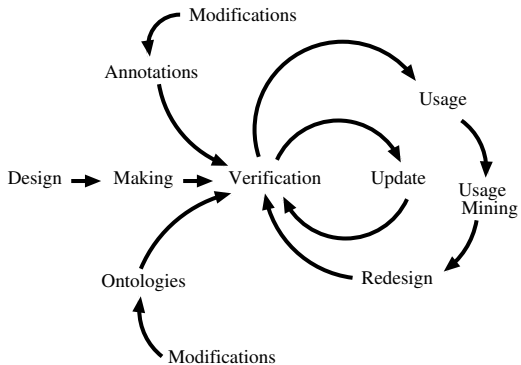
User centered redesign



Advanced conception



Complete evolution



Analogy between I.S. or Web sites and programs

- The lifetime is similar
- Verification is important
- Ontologies take the role of libraries
- Annotations take the role of data

Analogy between I.S. or Web sites and programs

- The lifetime is similar
- Verification is important
- Ontologies take the role of libraries
- Annotations take the role of data

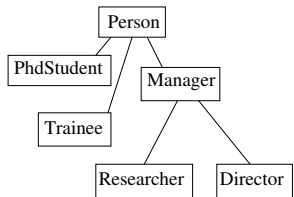
Analogy between I.S. or Web sites and programs

- The lifetime is similar
- Verification is important
- Ontologies take the role of libraries
- Annotations take the role of data

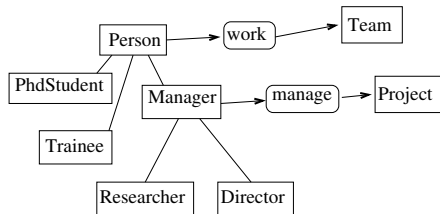
Analogy between I.S. or Web sites and programs

- The lifetime is similar
- Verification is important
- Ontologies take the role of libraries
- Annotations take the role of data

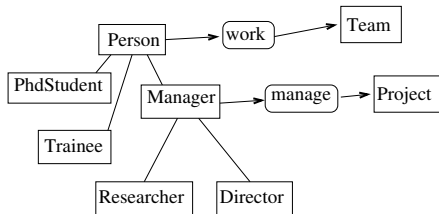
Example of ontology: concepts



Properties



Annotations



(r1 work v1) (r1 type Person)

(r2 work v2) (r2 type PhdStudent)

(r3 work v3) (r3 type Manager)

(r4 manage v4) (r4 type Manager)

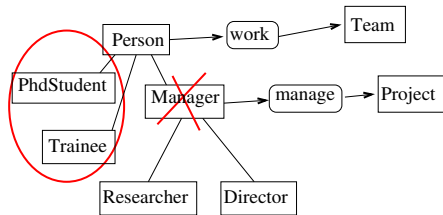
(r5 work v5) (r5 type Researcher)

(r6 manage v6) (r6 type Researcher)

(r7 work v7) (r7 type Director)

(r8 manage v8) (r8 type Director)

Modifying the ontology



(r1 work v1) (r1 type Person)

(r2 work v2) (r2 type PhdStudent)

(r3 work v3) (r3 type Manager)

(r4 manage v4) (r4 type Manager)

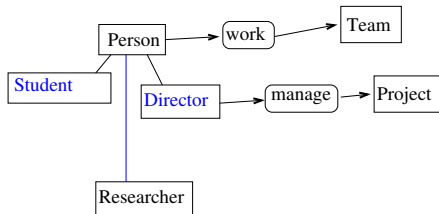
(r5 work v5) (r5 type Researcher)

(r6 manage v6) (r6 type Researcher)

(r7 work v7) (r7 type Director)

(r8 manage v8) (r8 type Director)

Result of the modifications



(r1 work v1) (r1 type Person)

(r2 work v2) (r2 type PhdStudent)

(r3 work v3) (r3 type Manager)

(r4 manage v4) (r4 type Manager)

(r5 work v5) (r5 type Researcher)

(r6 manage v6) (r6 type Researcher)

(r7 work v7) (r7 type Director)

(r8 manage v8) (r8 type Director)

Analogy between ontologies and type systems

- Concepts are viewed as types
- Subsumptions become type inclusions
- Properties get signatures
- Instances are typed

Analogy between ontologies and type systems

- Concepts are viewed as types
- Subsumptions become type inclusions
- Properties get signatures
- Instances are typed

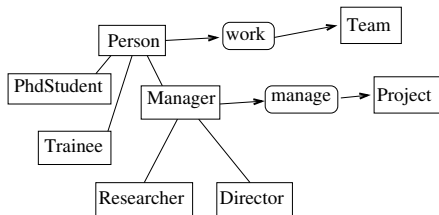
Analogy between ontologies and type systems

- Concepts are viewed as types
- Subsumptions become type inclusions
- Properties get signatures
- Instances are typed

Analogy between ontologies and type systems

- Concepts are viewed as types
- Subsumptions become type inclusions
- Properties get signatures
- Instances are typed

Example of translation



Person, Team, Project : type

PhdStudent <= Person

Trainee <= Person

Manager <= Person

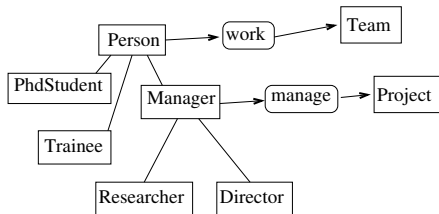
Researcher <= Manager

Director <= Manager

work : Person -> Team

manage : Manager -> Project

Example of translation



Person, Team, Project : type

PhdStudent \leq Person

Trainee \leq Person

Manager \leq Person

Researcher \leq Manager

Director \leq Manager

work : Person \rightarrow Team

manage : Manager \rightarrow Project

r1 : Person

r2: PhdStudent

r3, r4 : Manager

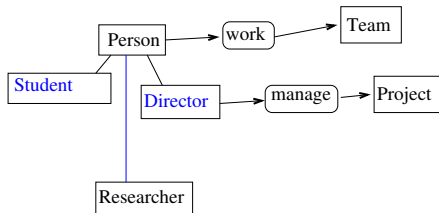
r5, r6 : Researcher

r7, r8 : Director

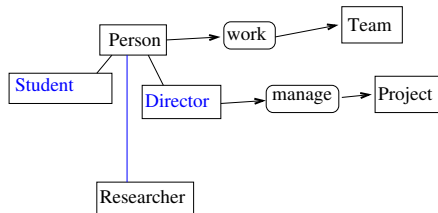
v1, v2, v3, v5, v7 : Team

v4, v6, v8 : Project

Applying modifications



Result of the modification



Person, Team, Project : type

Student \leq Person

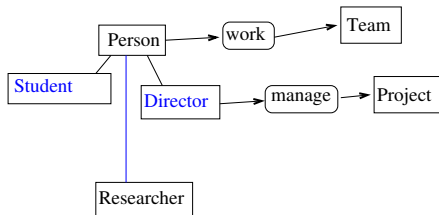
Researcher \leq Person

Director \leq Person

work : Person \rightarrow Team

manage : Director \rightarrow Project

Type checking annotations



Person, Team, Project : type
Student \leq Person
Researcher \leq Person
Director \leq Person
work : Person \rightarrow Team
manage : Director \rightarrow Project

r1 : Person

r2 : PhdStudent **The type PhdStudent is not defined**

r3, r4 : Manager **The type Manager is not defined**

r5, r6 : Researcher

r7, r8 : Director

v1, v2, v3, v5, v7 : Team

va, v6, v8 : Project

r1 work v1

r2 work v2 **r2 is not of type Person**

r3 work v3 **r3 is not of type Person**

r4 manage v4 **r4 is not of type Person**

r5 work v5

r6 manage v6 **r6 is not of type Director**

r7 work v7

r8 manage v8

- Developing I.S. or Web sites and programs is very similar
- Ontologies can be viewed as type systems
- Question: What else in type theory and in the world of programming can be useful to develop and use ontologies ? modularity, polymorphism, types as parameter...

- Developing I.S. or Web sites and programs is very similar
- Ontologies can be viewed as type systems
- Question: What else in type theory and in the world of programming can be useful to develop and use ontologies ? modularity, polymorphism, types as parameter...

- Developing I.S. or Web sites and programs is very similar
- Ontologies can be viewed as type systems
- Question: What else in type theory and in the world of programming can be useful to develop and use ontologies ? modularity, polymorphism, types as parameter...

- Thank you