

# COMP318

## Ontologies and Semantic Web

# Ontology Engineering - Part 1



Dr Valentina Tamma

v.Tamma@liverpool.ac.uk

# Ontology Engineering

*“The set of activities that concern the ontology development process, the ontology life cycle, and the methodologies, tools and languages for building ontologies”*

Gomez-Perez et al, 2004



# Ontology Engineering

- Defining terms in the domain and relations among them
  - Defining concepts in the domain (**classes**)
  - Arranging the concepts in a hierarchy (**subclass-superclass hierarchy**)
  - Defining which attributes and properties classes can have
    - and constraints on their values
  - Defining individuals and filling in property values



# Methodological questions

- What part of the domain do we need to model?
- What are the constraints on the use of this knowledge?
- How can tools and techniques best be applied?
- Which languages and tools should be used in which circumstances, and in which order?
- What about issues of quality control and resource management?
  - Many of these questions for ontology engineering have been studied in other contexts
    - E.g. software engineering, object-oriented design, and knowledge engineering

# Principles for the design of an ontology 1

- **Clarity:**

- To communicate the intended meaning of defined terms
  - An ontology should communicate **effectively** the intended meaning of defined terms.
  - Definitions should be objective and should be stated with formal axioms.
  - A complete definition (defined by necessary and sufficient conditions) is preferred over a partial definition (defined by only necessary or sufficient conditions)

- **Coherence:**

- To sanction inferences that are consistent with definitions
  - If a sentence that can be inferred from the axioms contradicts a definition or example given informally, then the ontology is incoherent.

- **Extendibility:**

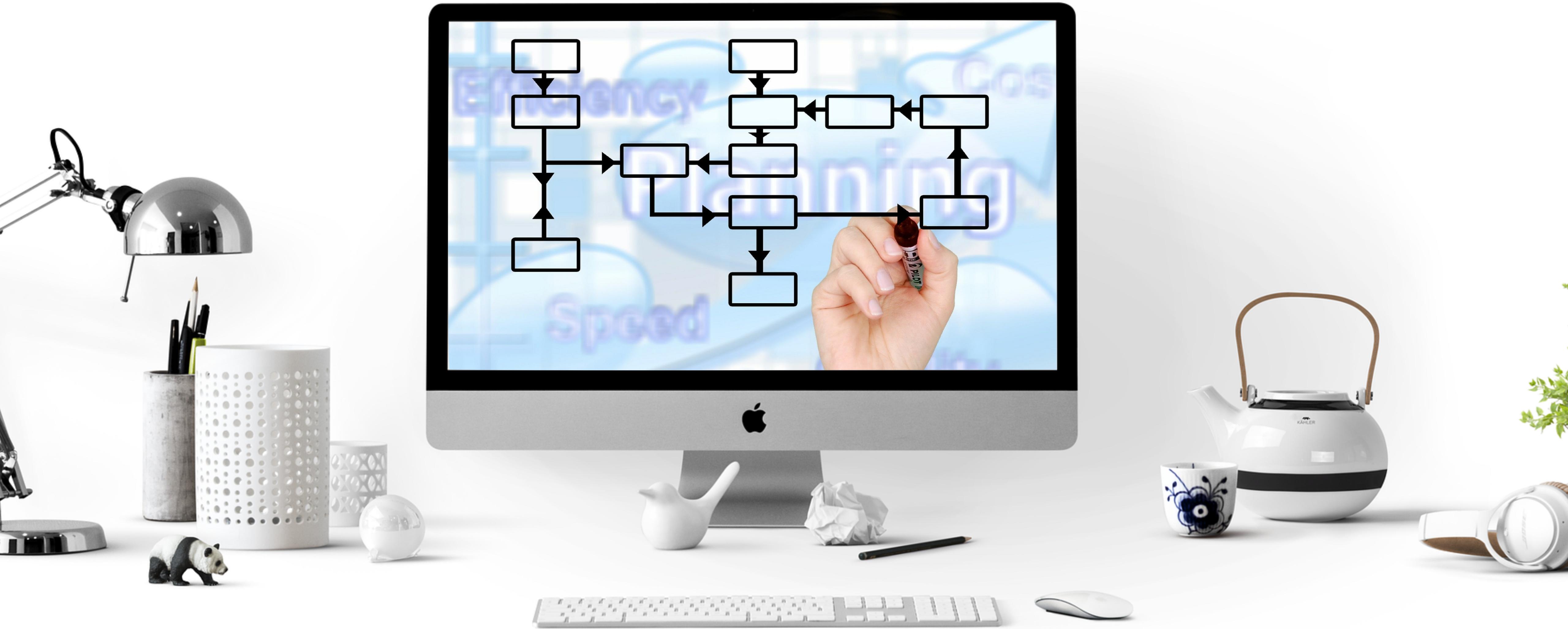
- To anticipate the use of the shared vocabulary
  - define new terms for special uses based on the existing vocabulary, in a way that does not require the revision of the existing definitions.

# Principles for the design of an ontology - 2

- **Minimal Ontological Commitments:**
  - To make as few claims as possible about the world
    - ontological commitment is based on the consistent use of the vocabulary,
    - ontological commitment can be minimised by specifying the weakest theory and defining only those terms that are essential to the communication of knowledge consistent with the theory.

- **Minimal Encoding Bias:**
  - To be independent of the symbolic level
  - The conceptualisation is specified at the knowledge level without depending on a particular symbol-level encoding.
  -

# Ontology engineering methodologies



# Methodologies

- Various methodologies have been proposed to formalise the development process, or even some of its phases
  - Uschold & King
  - Gruninger & Fox
  - Methontology (Gomez Perez et al)
  - Ontology Design Patterns (Gangemi et al)
  - Ontology 101 (Noy & McGuinness)

# Uschold & King

- Identify purpose
- Build ontology
  - Capture
  - Coding
  - Integrating
- Evaluation
- Documentation

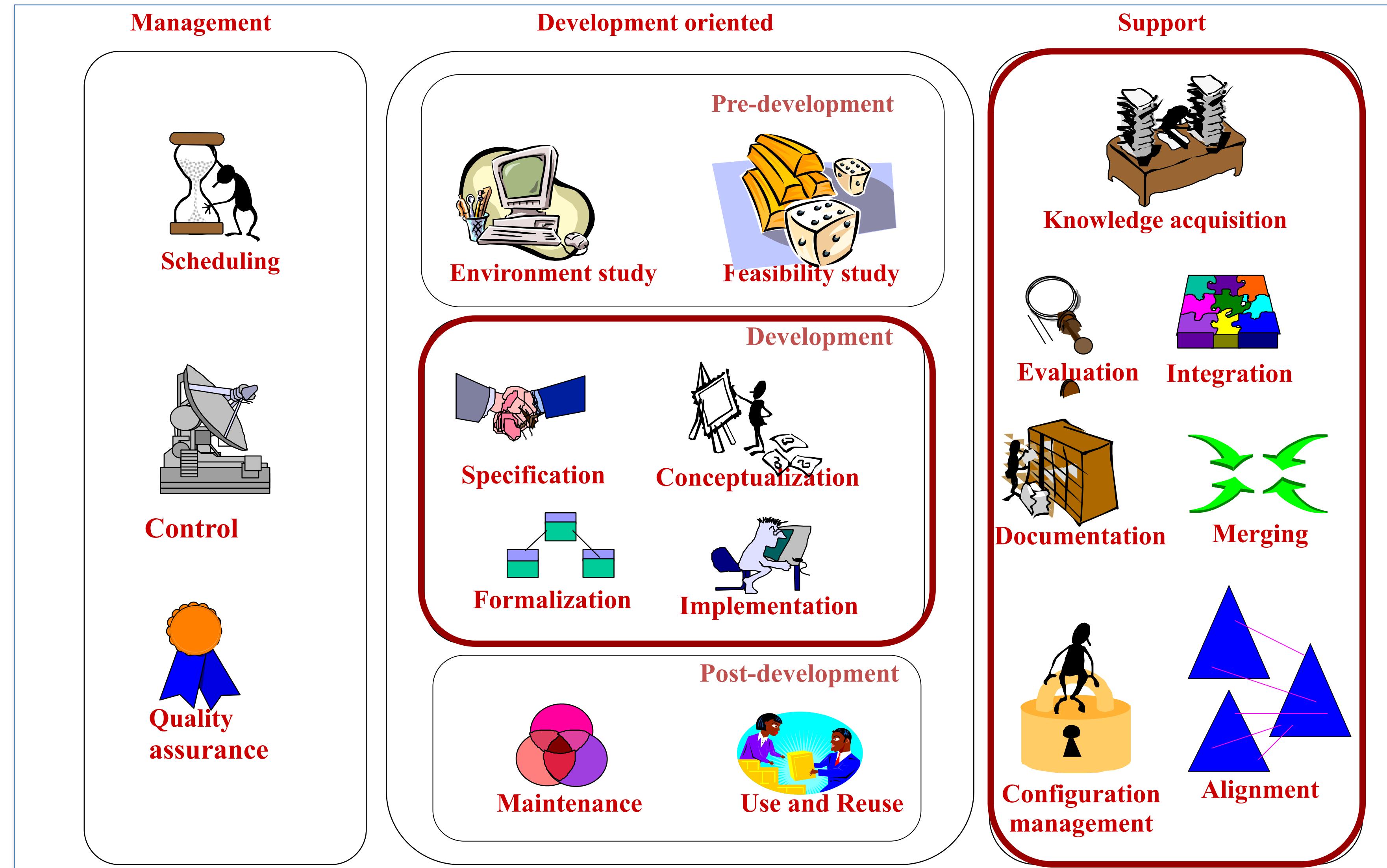
# Gruninger & Fox

- Identify motivating scenarios
- Elaborate informal competency questions
- Specify terminology in FOL
  - Identify objects
  - Identify predicates
- Formal competency questions
- Specify axioms in FOL
- Specify completeness theorems

# Methontology

- Development + Management Activities + Support in parallel
- Development
  - Specification
  - Conceptualisation
  - Formalisation
  - Implementation
  - Maintenance
- Focus on the conceptualisation activity
  - Advantage: Integration of existing ontologies considered from early on.
- Conceptualization is evaluated early on, which prevents propagation of errors.

# Ontology development process



# Management activities

Management

- **Scheduling**

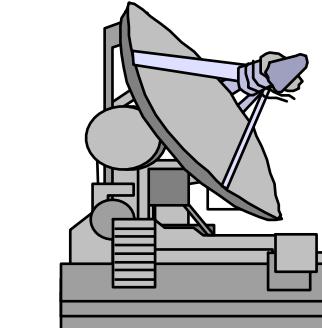
- Identification of tasks/problems to solve
- Arrangement/planning of tasks/problems to solve
- Identification of required resources (time, memory, resources)



Scheduling

- **Control**

- Ensuring the correct execution of tasks / problems to solve



Control

- **Quality Assurance**

- Ensuring the quality of all the artefacts produced during development
  - ontologies, software, and documentation



Quality  
assurance

# Development activities of the design process

- Pre-Development
- Development
- Post-Development
- Support



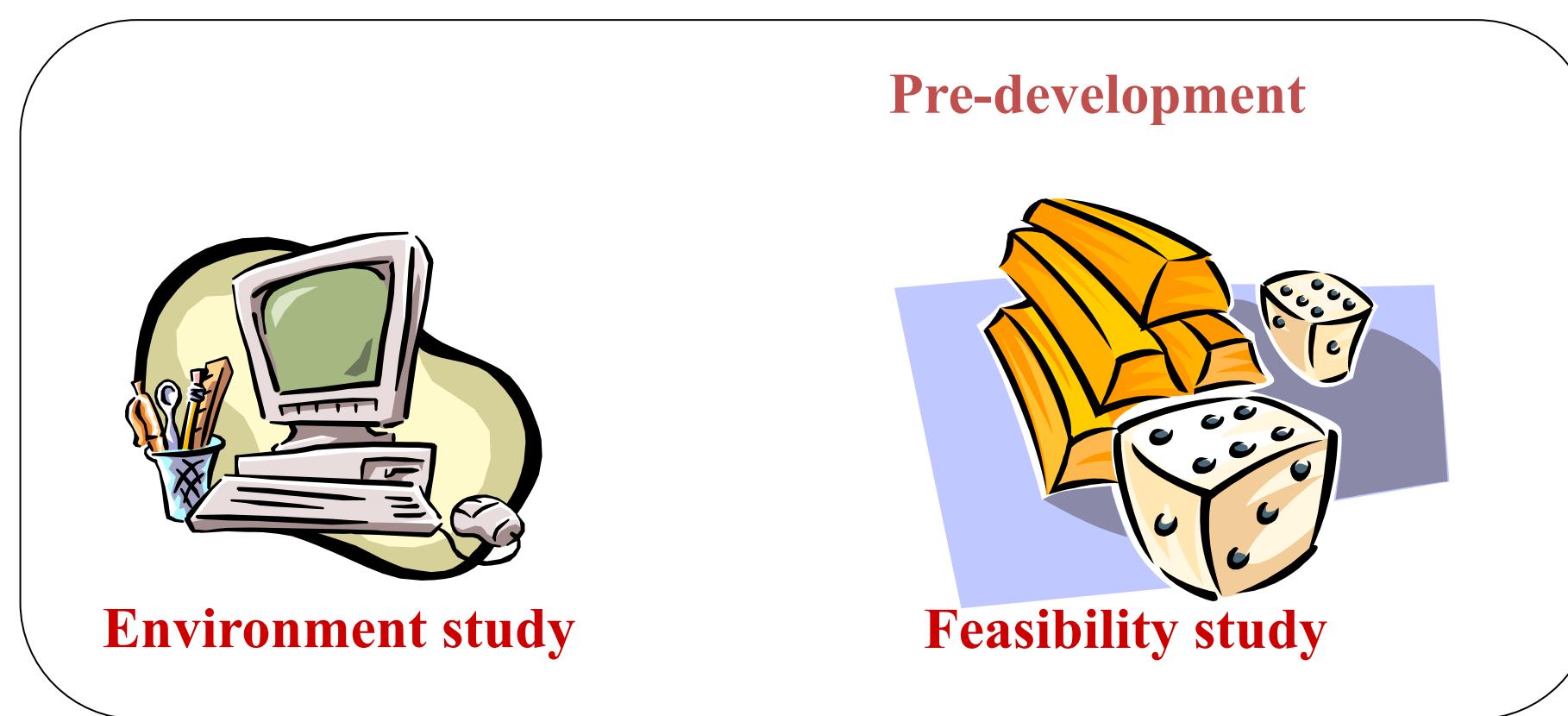
# Pre-development activities

- **Environment study**

- Determine the environment in which the ontology should operate:
  - What is the designated software platform for the ontology?
  - Which applications should use the ontology?

- **Feasibility study**

- Assesses the feasibility and value of the ontology
  - Can the ontology really be developed?
  - Does it make sense to develop the ontology?



# Development activities

- **Specification**

- Use-cases and requirements
  - Why is the ontology developed, what is the benefit and who are the end-users?
  - What are the questions the ontology should answer?

- **Conceptualisation**

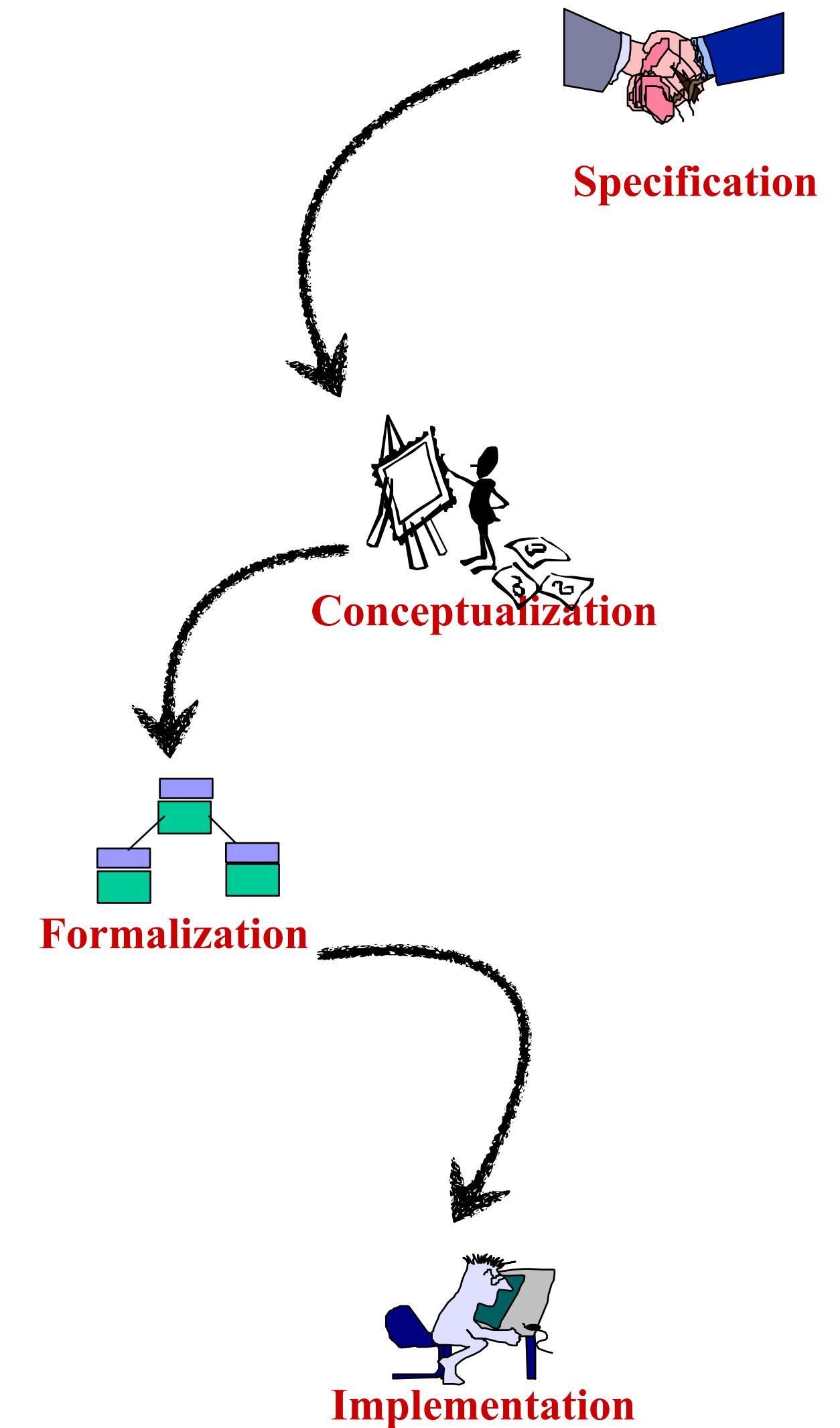
- Generation of a conceptual model that provides an abstraction of the domain model

- **Formalisation**

- Translation the conceptual model into a (semi) computable model

- **Implementation**

- Construction of a of a computable model in an ontology representation language (e.g. RDFS, OWL)



# Post-Development Activities

- **Maintenance**

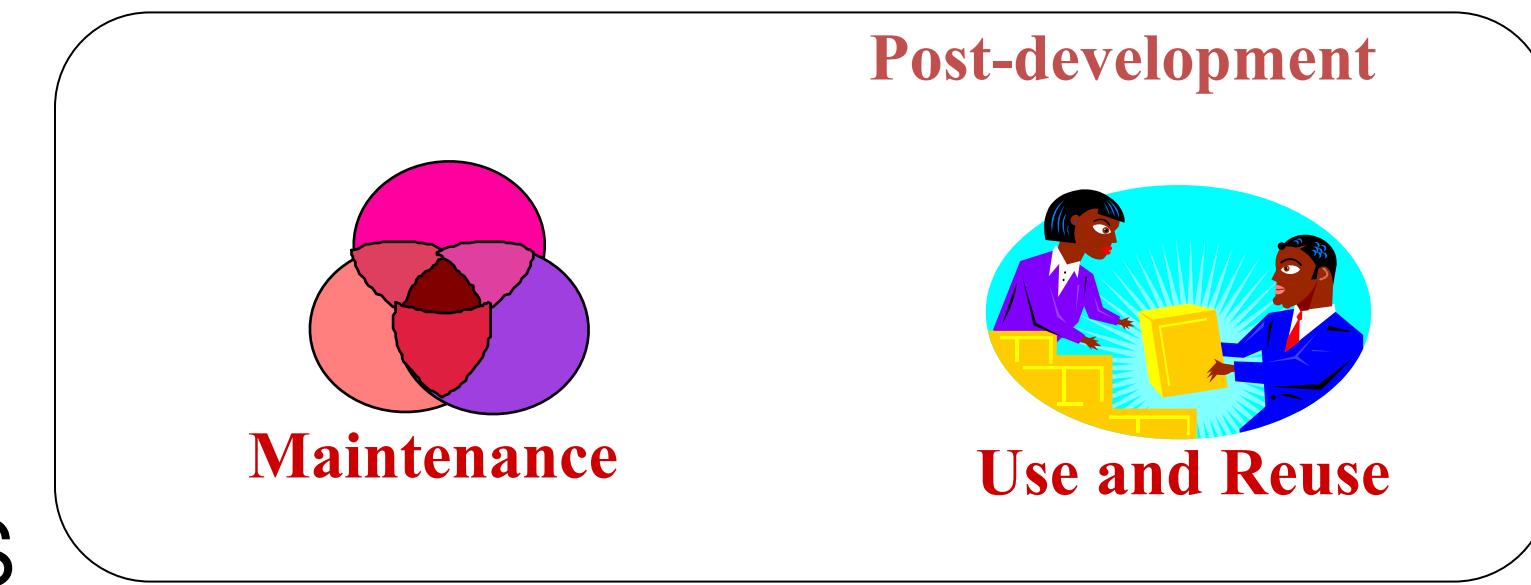
- Update and amendment of the ontology (if necessary)

- **Use**

- Usage of the ontology within the designated applications

- **Reuse**

- Use of the ontology in novel, unplanned applications
  - very common for top level ontologies



# Support Activities

## • Knowledge Acquisition

- The process of eliciting (tacit) knowledge from domain expert
- Learning semi-automatically the ontology from text

## • Evaluation

- Technical evaluation of the outcome of each step of the ontology development process

## • Documentation

- Accurate documentation of each step of the ontology development process

## • Configuration management

- Management of the different versions of the ontology produced and its documentation
  - Versioning

Support



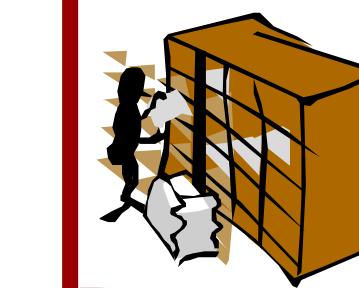
Knowledge acquisition



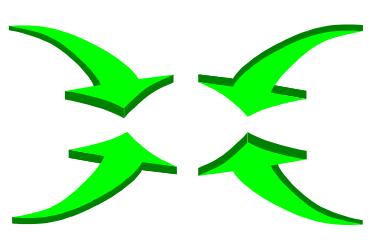
Evaluation



Integration



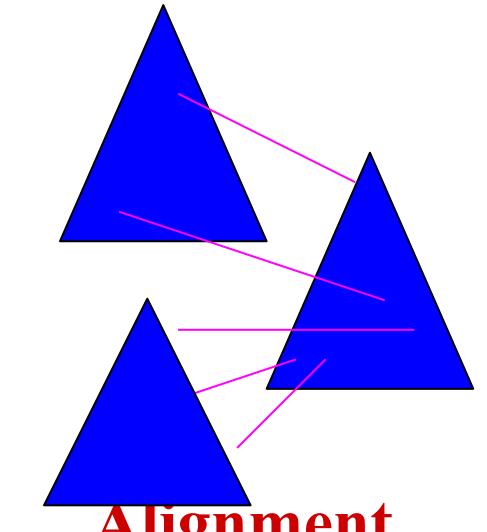
Documentation



Merging



Configuration management



Alignment

# Support Activities

## • Integration

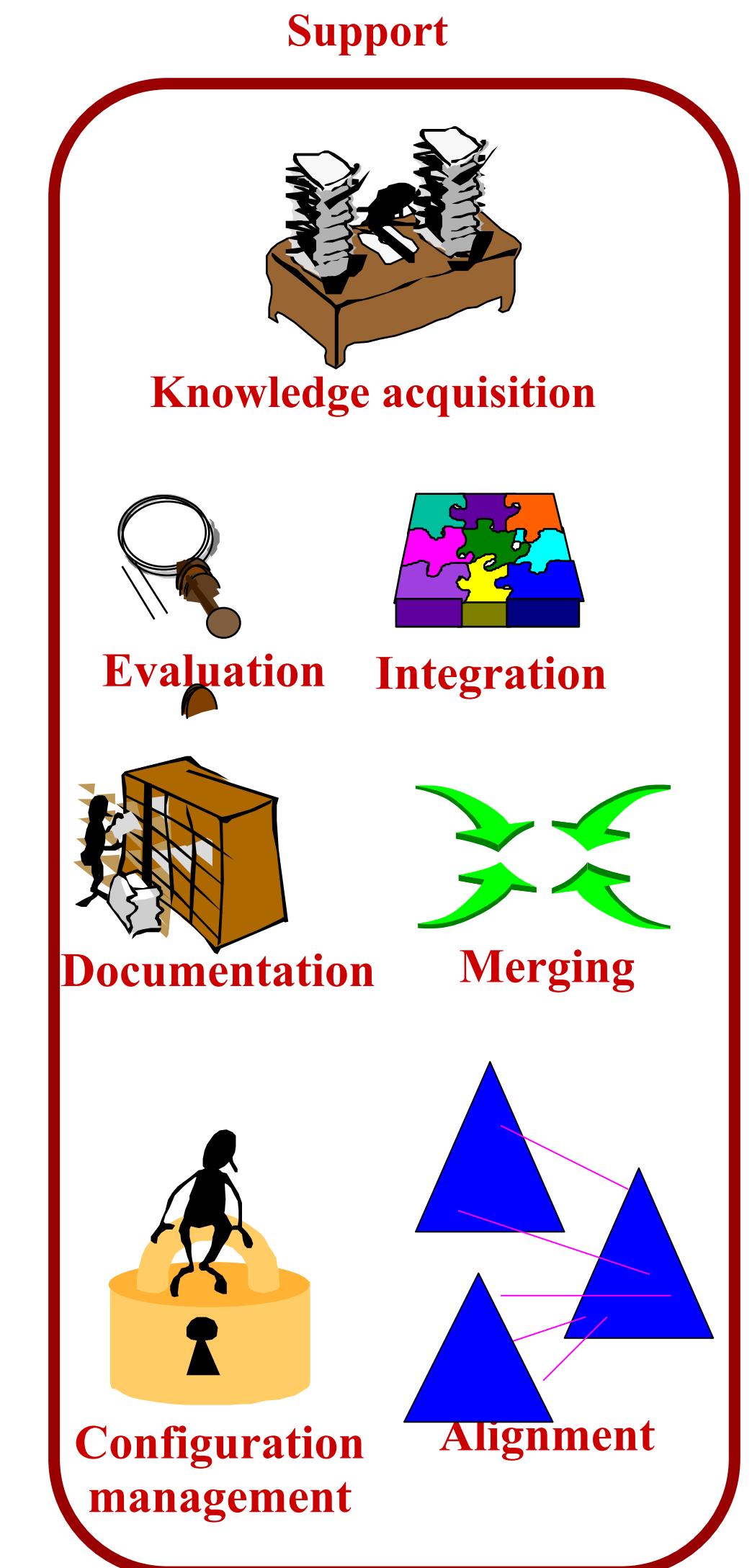
- Integrating two ontologies through the definition of mappings between them
  - Definition of a global schema

## • Merging

- Ontology merging describes the process of integrating two (or more) ontologies into a single one.

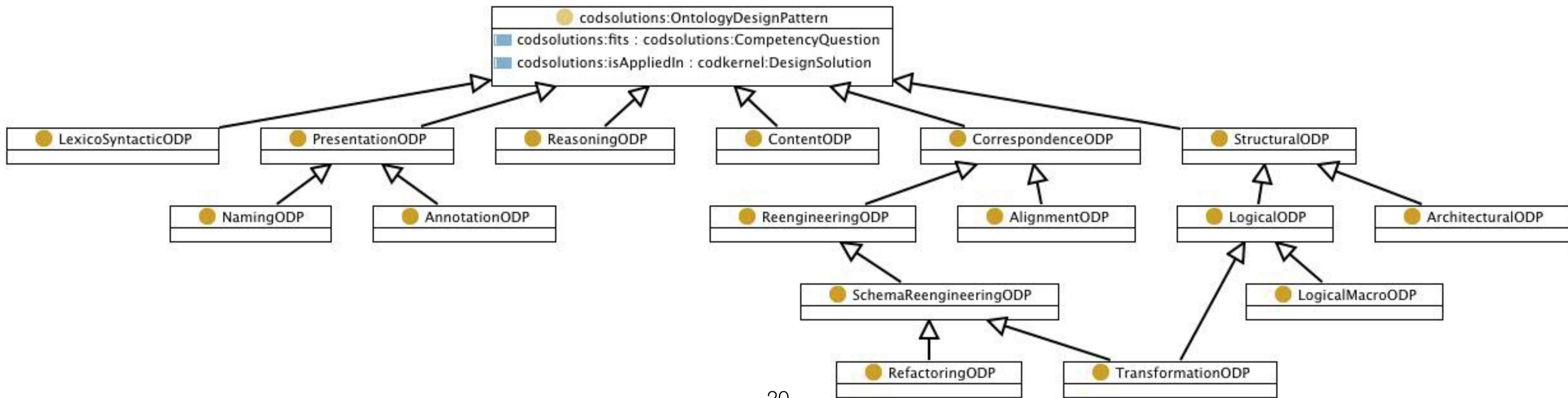
## • Alignment

- Determine or apply mapping rules for reconciling the involved ontologies



# Ontology design patterns

- Adapting a design idea originally from architecture
  - recurring modeling problems
  - providing a set of adaptable standard solutions
  - a “**pattern**” is a solution to a problem in a given context



# COMP318

## Ontologies and Semantic Web

# End of Ontology Engineering - Part 1



Dr Valentina Tamma

v.Tamma@liverpool.ac.uk