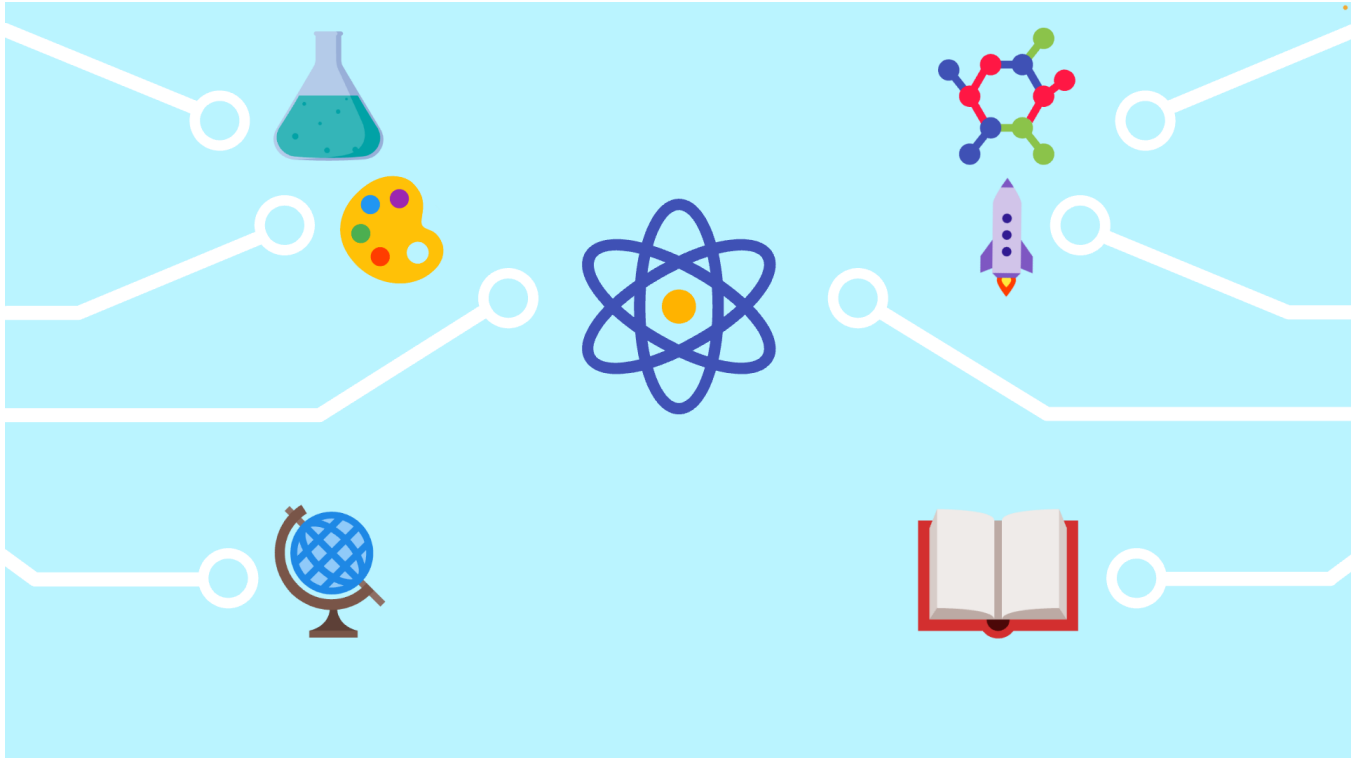
Practical Knowledge Modelling: Ontology Development 101



graftilo

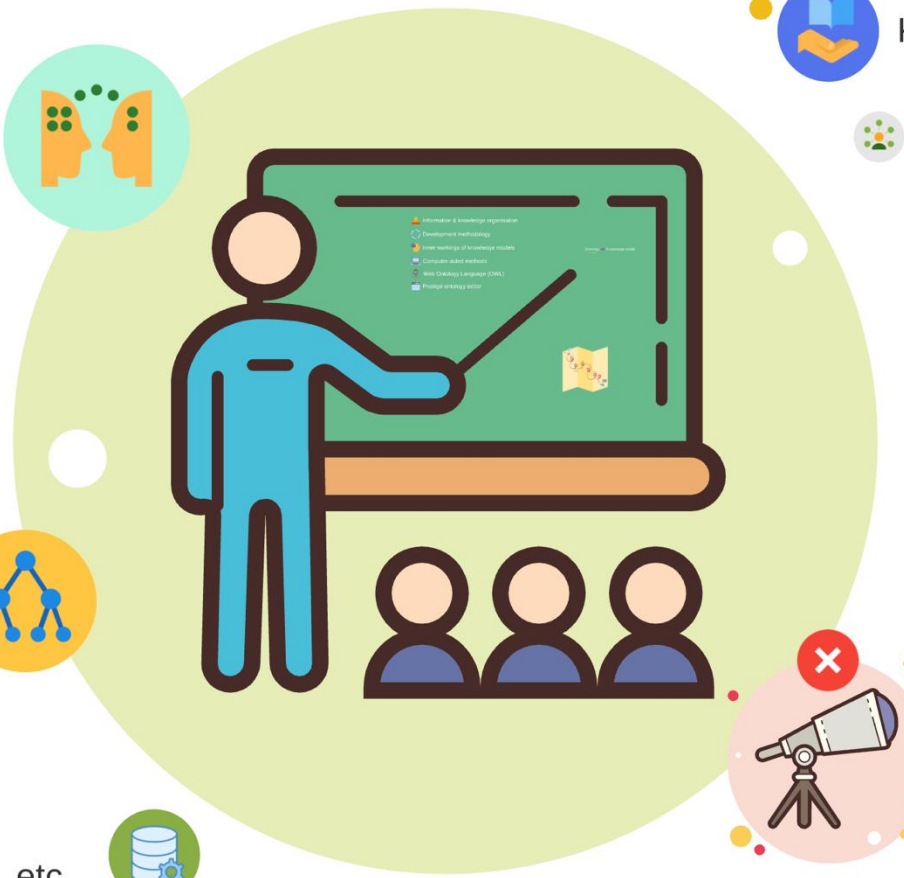
01 Introduction

Knowledge management

Business analysis

Enterprise architecture

Knowledge sharing



Formal methods for knowledge capture

Information systems, database technologies, etc.

Out-of-scope:

- Advanced ontology engineering
- Ontology theory
- Mathematical logic
- Graph RAG using ontology

Information & knowledge organisation

Development methodology

Inner workings of knowledge models

Computer-aided methods

Web Ontology Language (OWL)

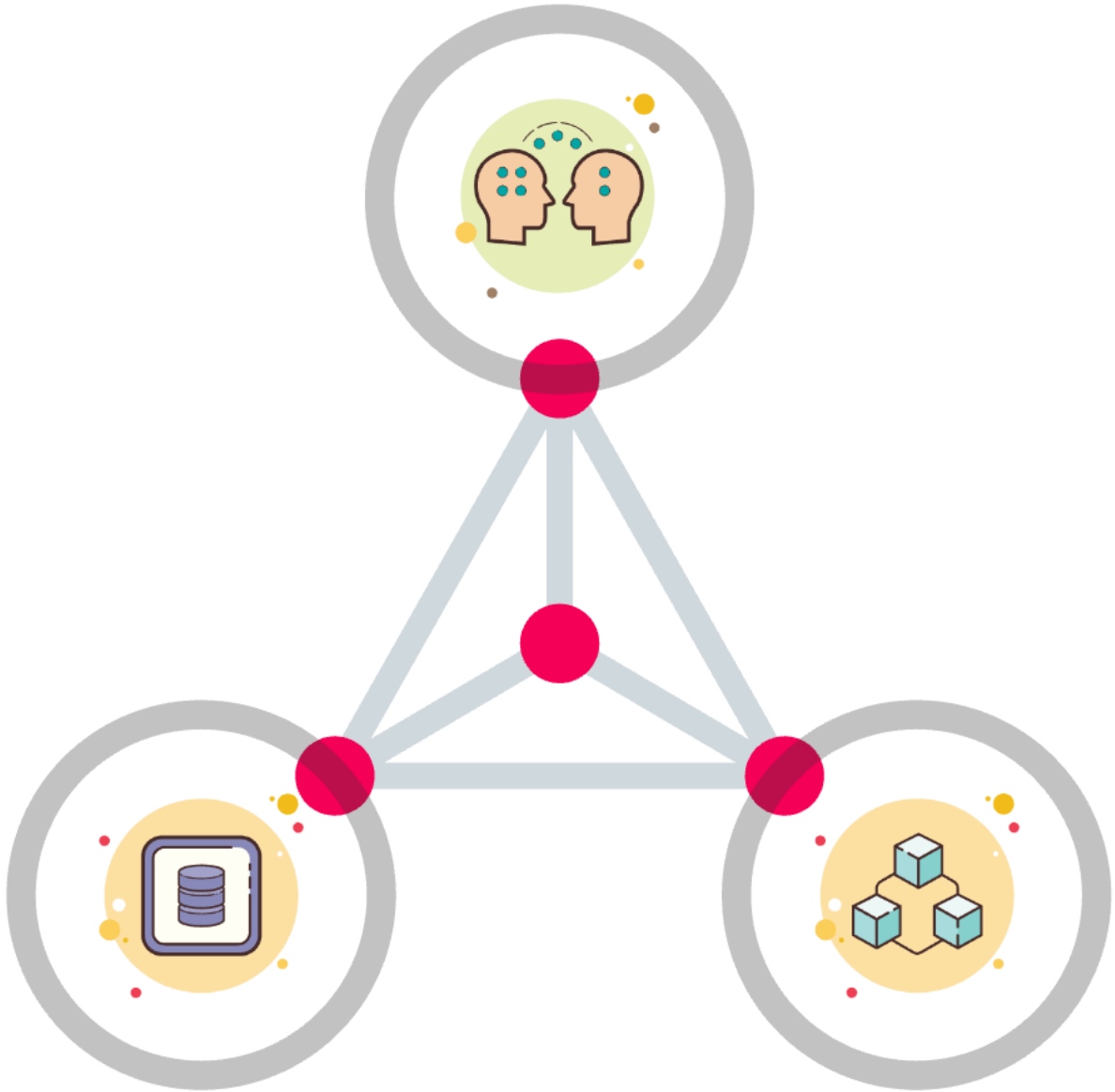
Protégé ontology editor


Introduction
What is knowledge modelling?
A methodology for knowledge modelling
Initial structuring
Formalization
Deployment
Evaluation
Course wrap-up

Ontology = Knowledge model

pl. ontologies

02 What is knowledge modelling?




Knowledge
management

Knowledge management

Explicit knowledge

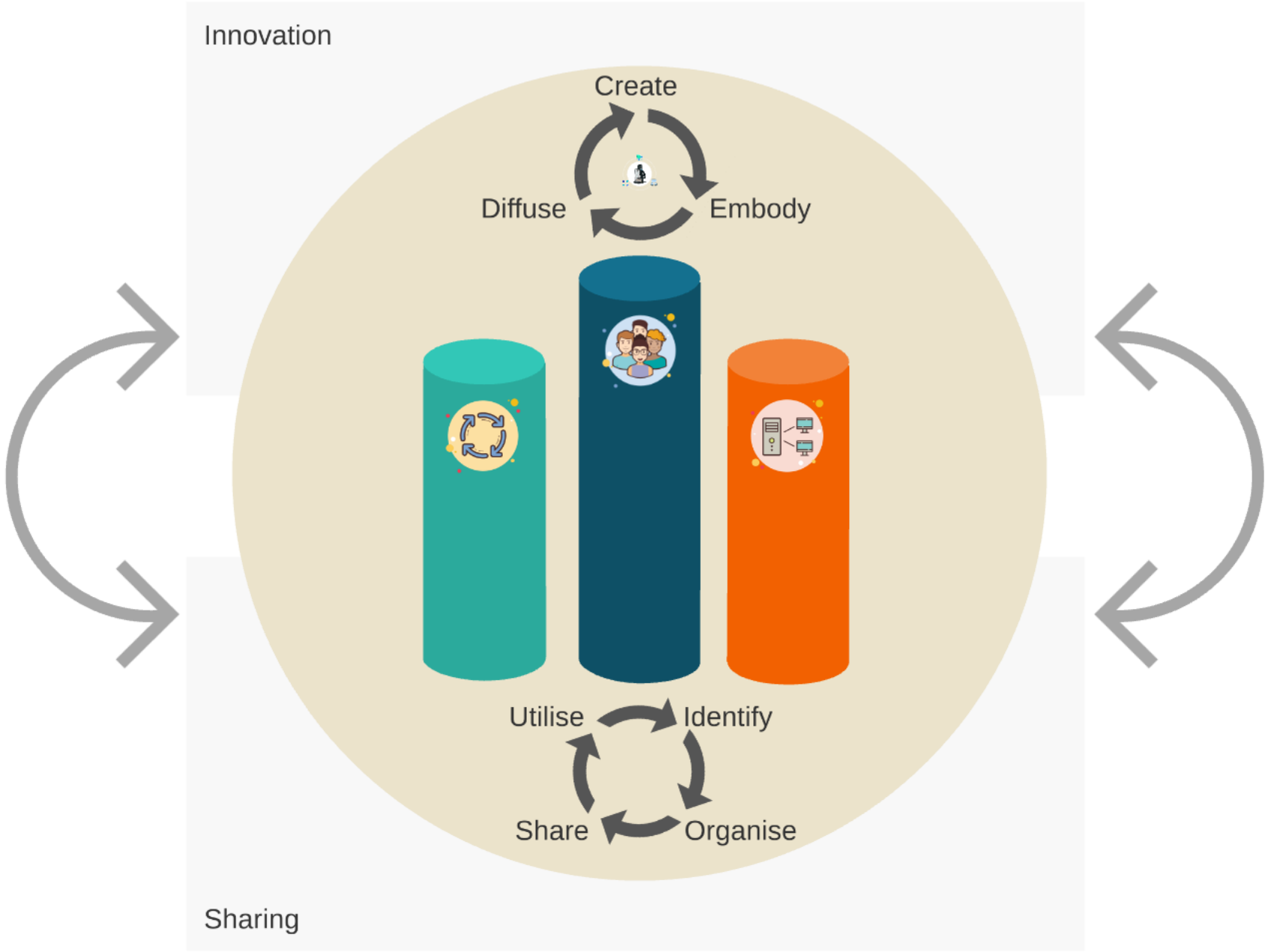
- Easy to identify, store & share
- Multiple sources
- Expressed externally

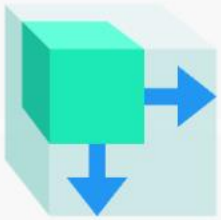
Tacit knowledge

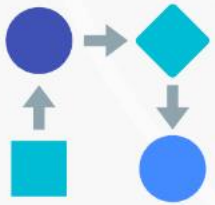
- Acquired from experience
- Hard to express externally

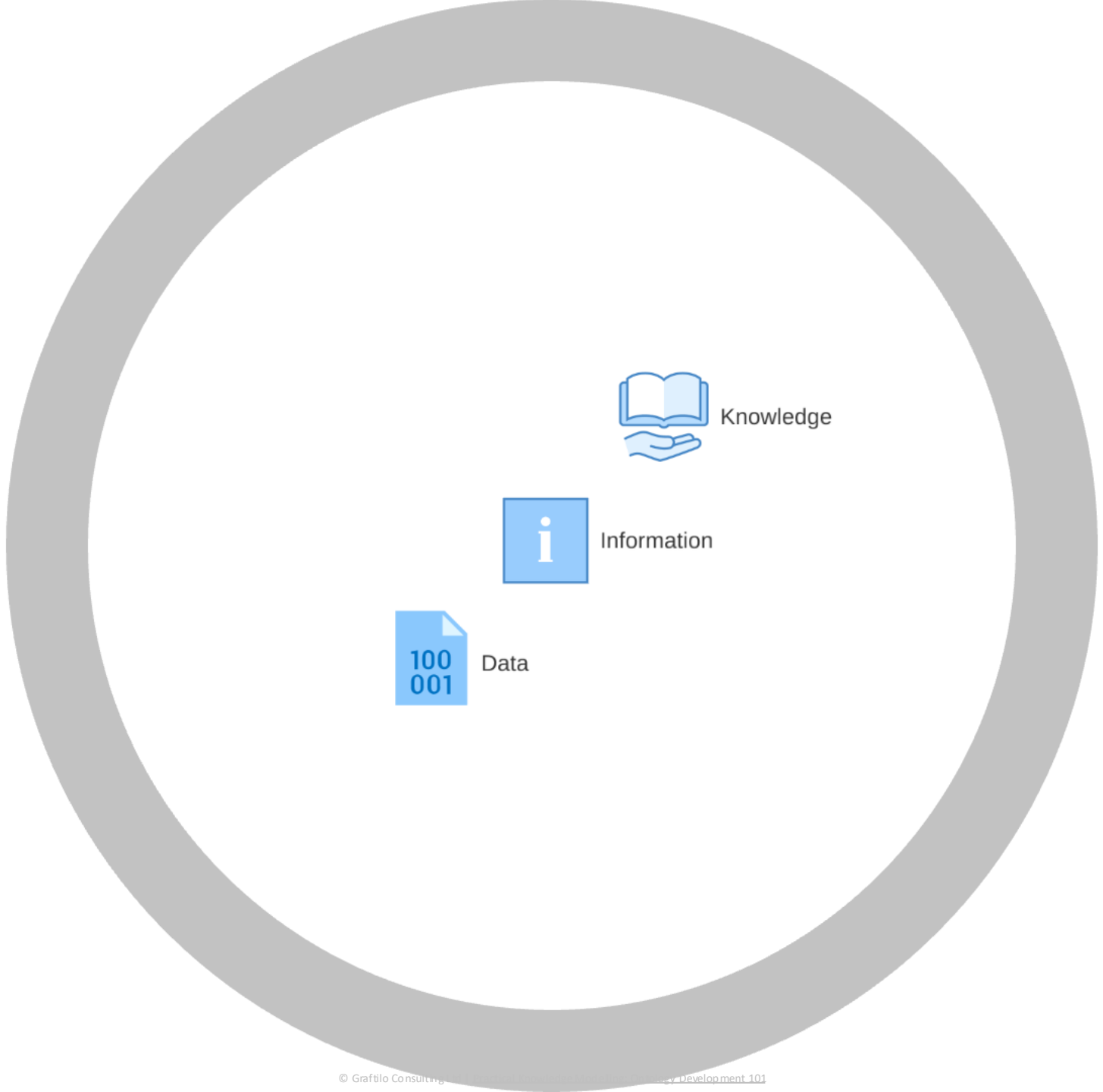
Implicit knowledge

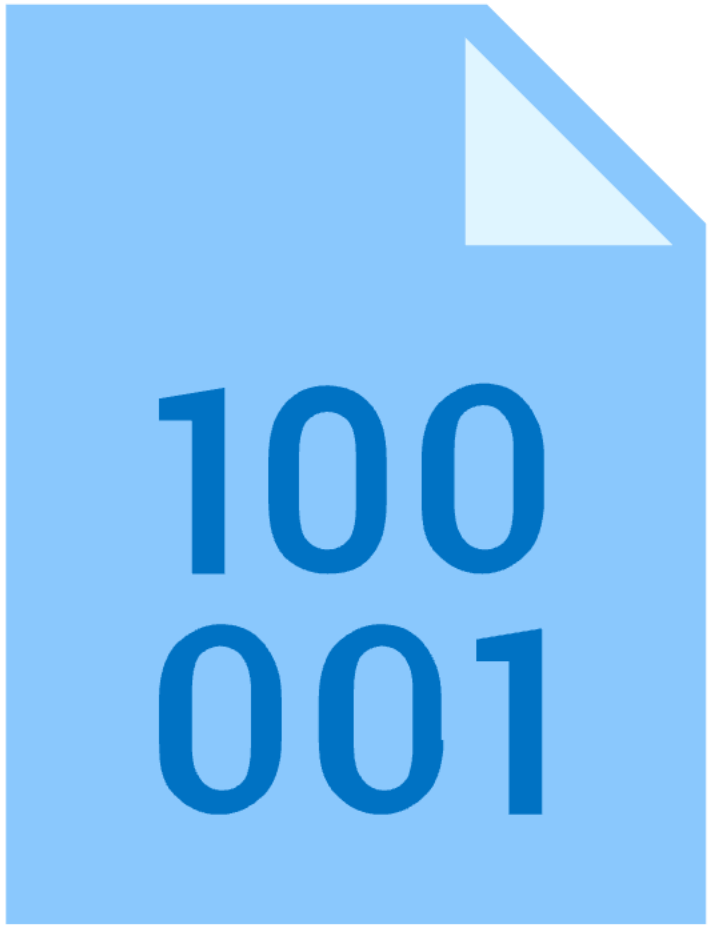
- Internal
- Can be expressed externally


Innovation
Create
Diffuse
Embody
Utilise
Identify
Share
Organise
Sharing









Data

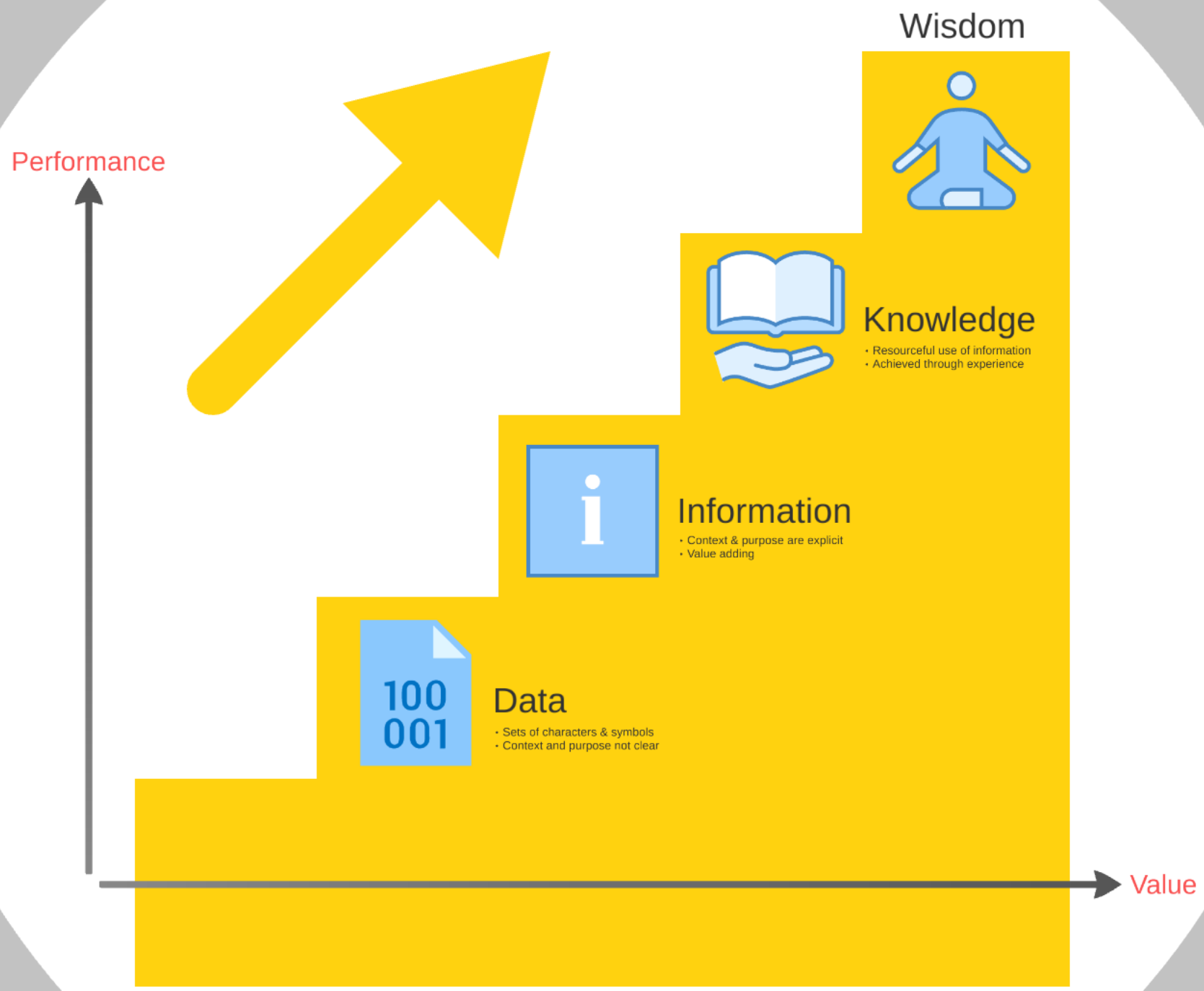
- Sets of characters & symbols
- Context and purpose not clear

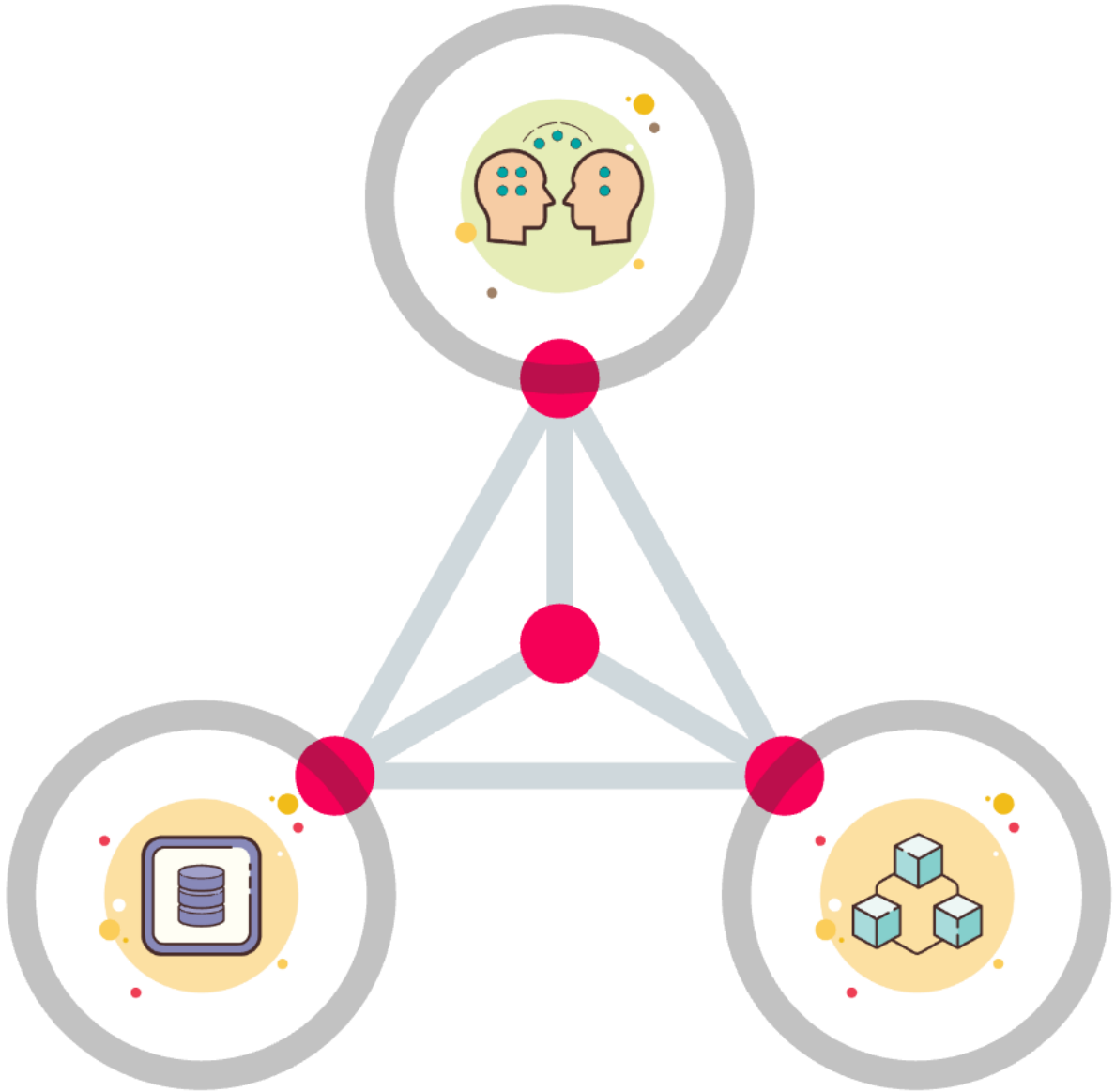
Information

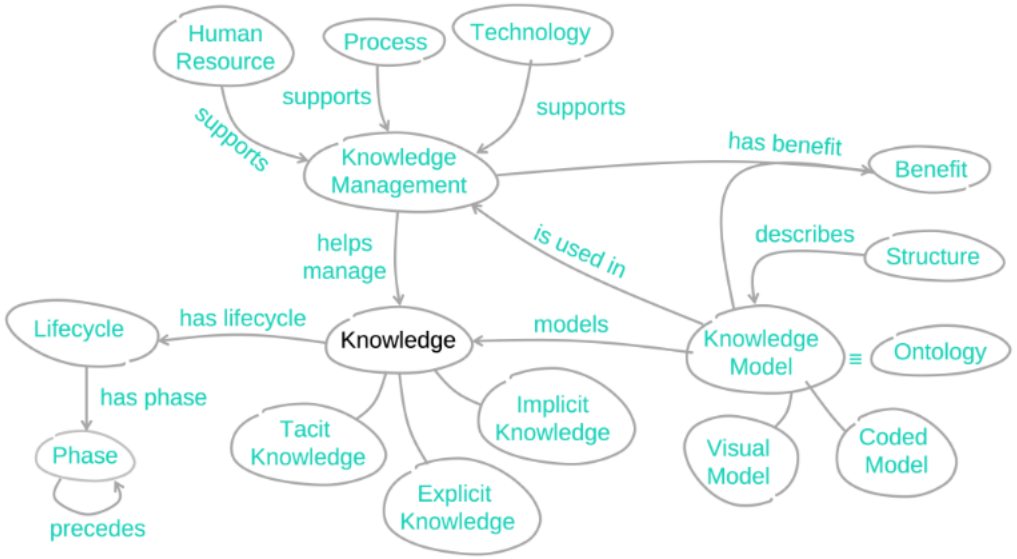
- Context & purpose are explicit
- Value adding

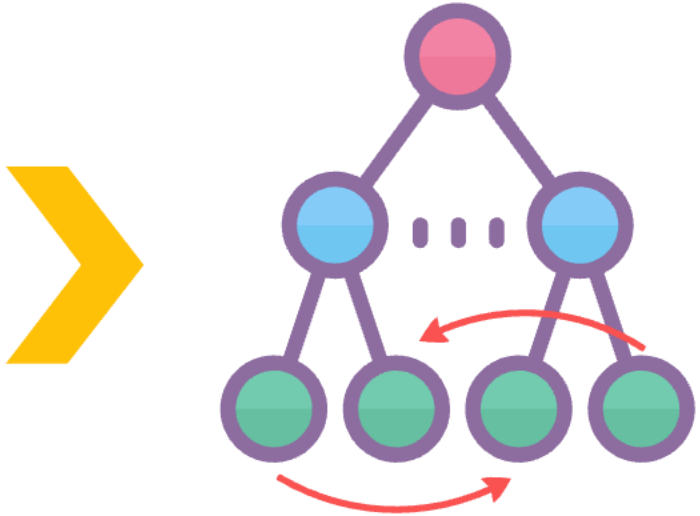
Knowledge

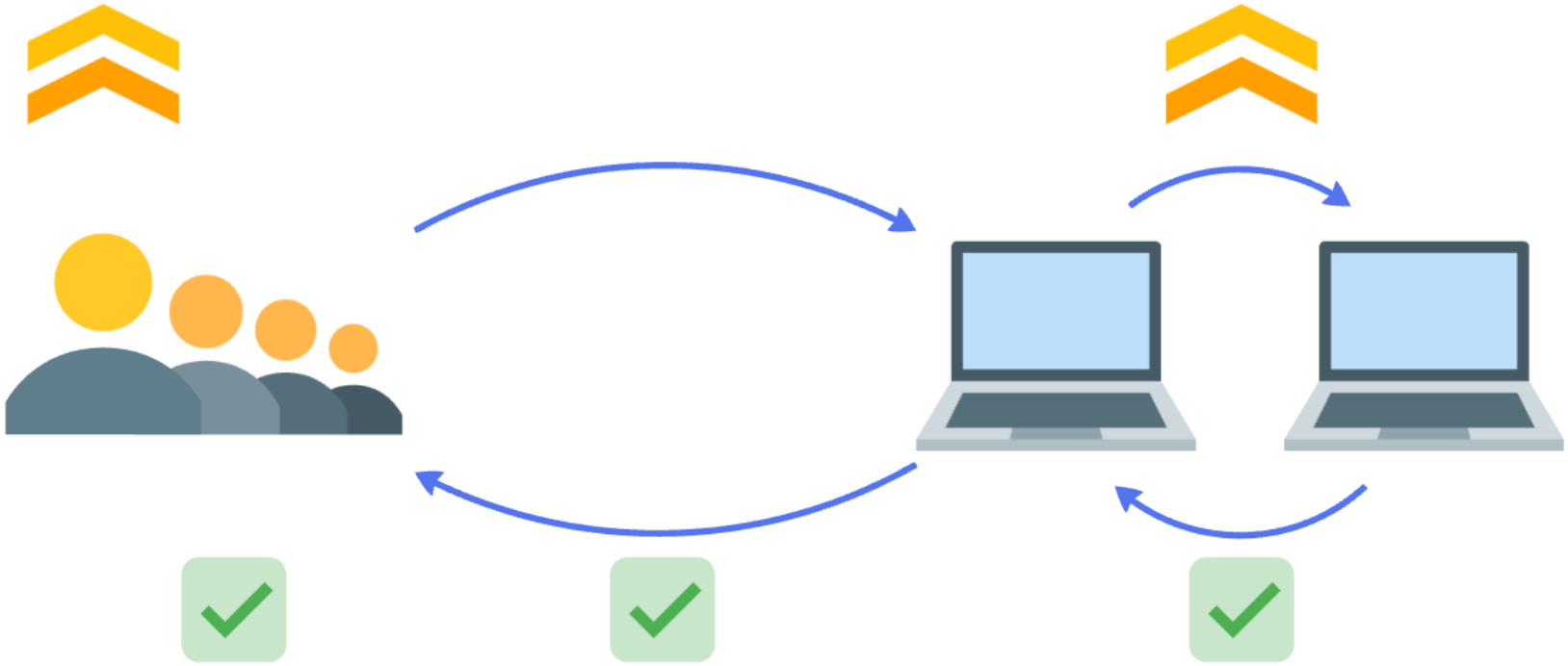
- Resourceful use of information
- Achieved through experience


Wisdom
Performance
Knowledge
• Resourceful use of information
• Achieved through experience
Information
• Context & purpose are explicit
• Value adding
100
001
Data
• Sets of characters & symbols
• Context and purpose not clear
Value




Human Resource
Process
Technology
supports
supports
supports
Knowledge Management
has benefit
Benefit
helps manage
is used in
describes
Structure
Lifecycle
has lifecycle
Knowledge
models
Knowledge Model
≡
Ontology
has phase
Phase
precedes
Tacit Knowledge
Implicit Knowledge
Explicit Knowledge
Visual Model
Coded Model





A translation approach to portable ontologies

Gruber, 1993

An explicit specification of a conceptualization

A formal, explicit specification of a shared conceptualization

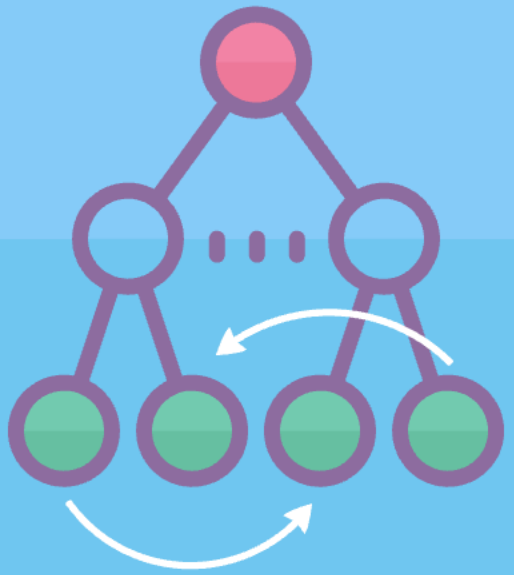
Machine readable

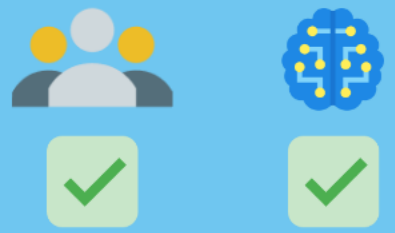
Precise representation

Derived from consensus

Subject matter

Studer et al., 1997

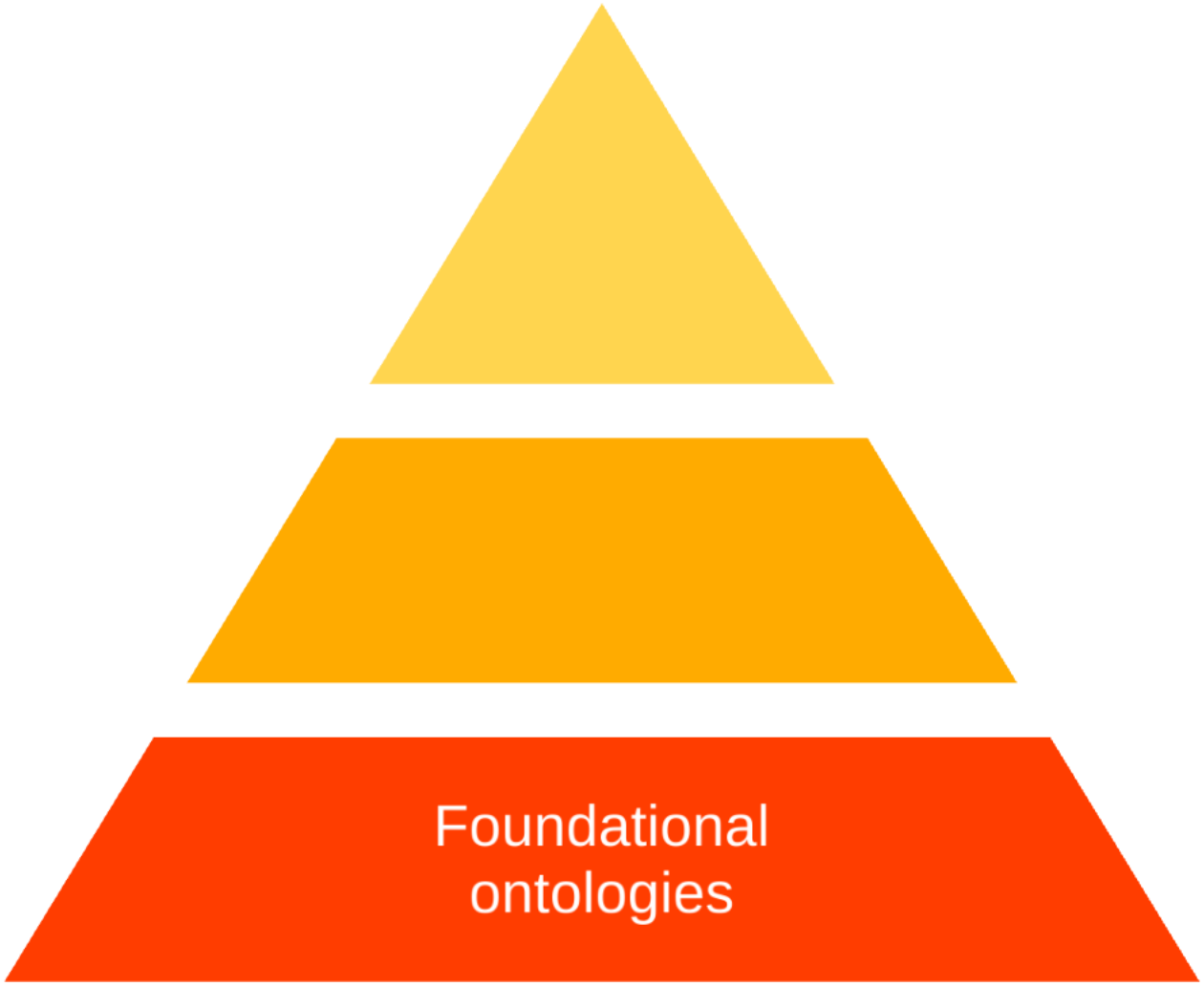


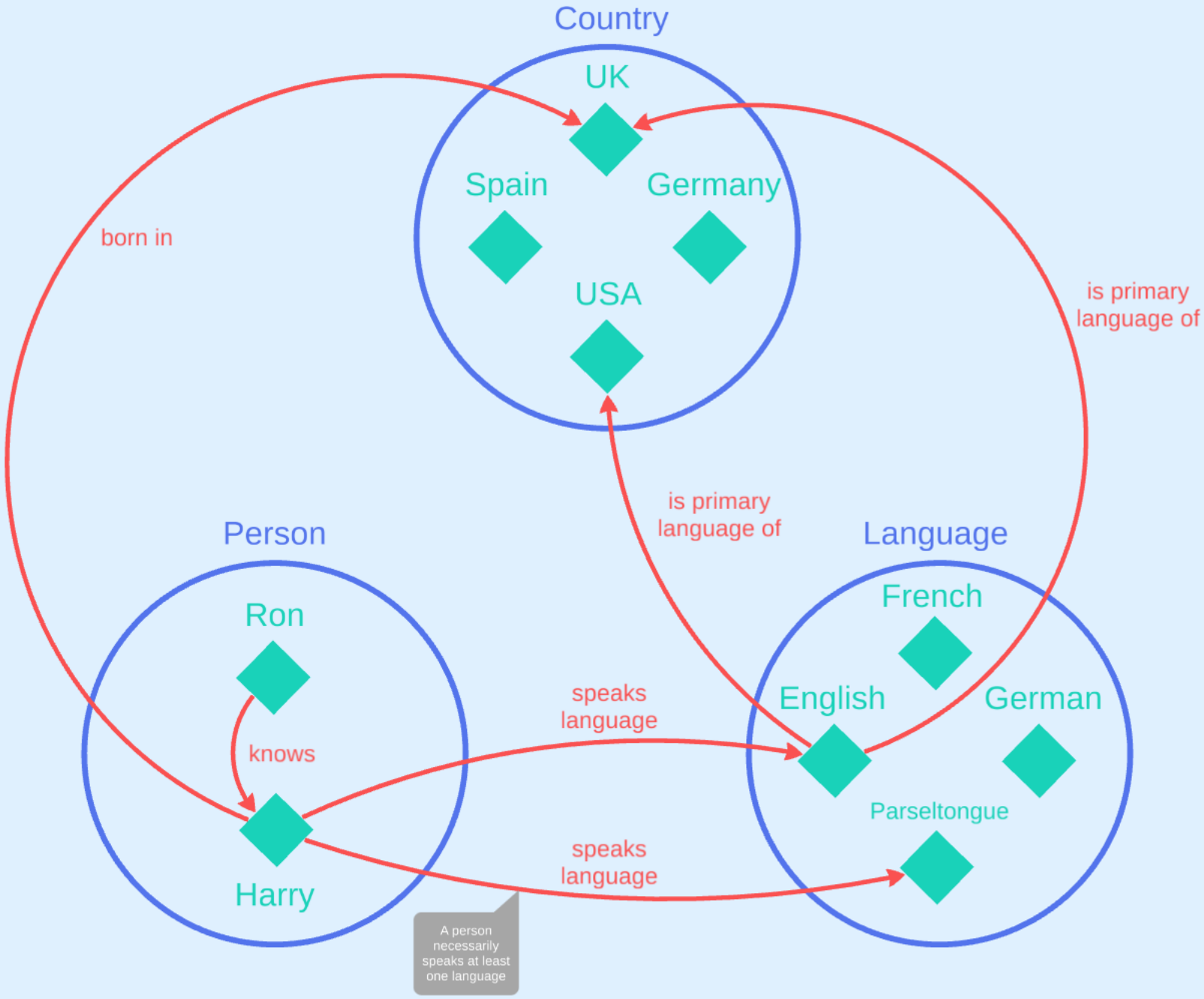


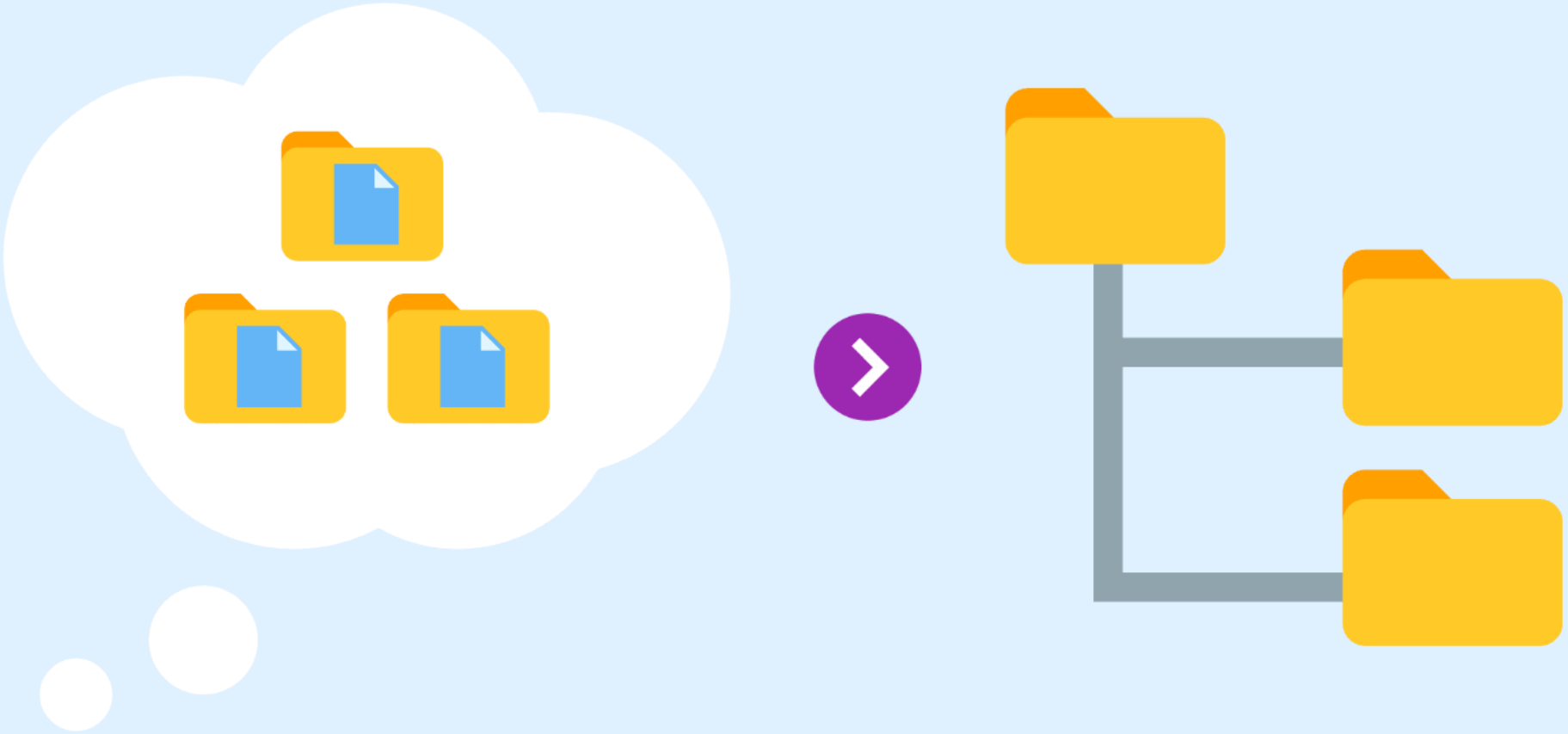
Ontology (philosophy)

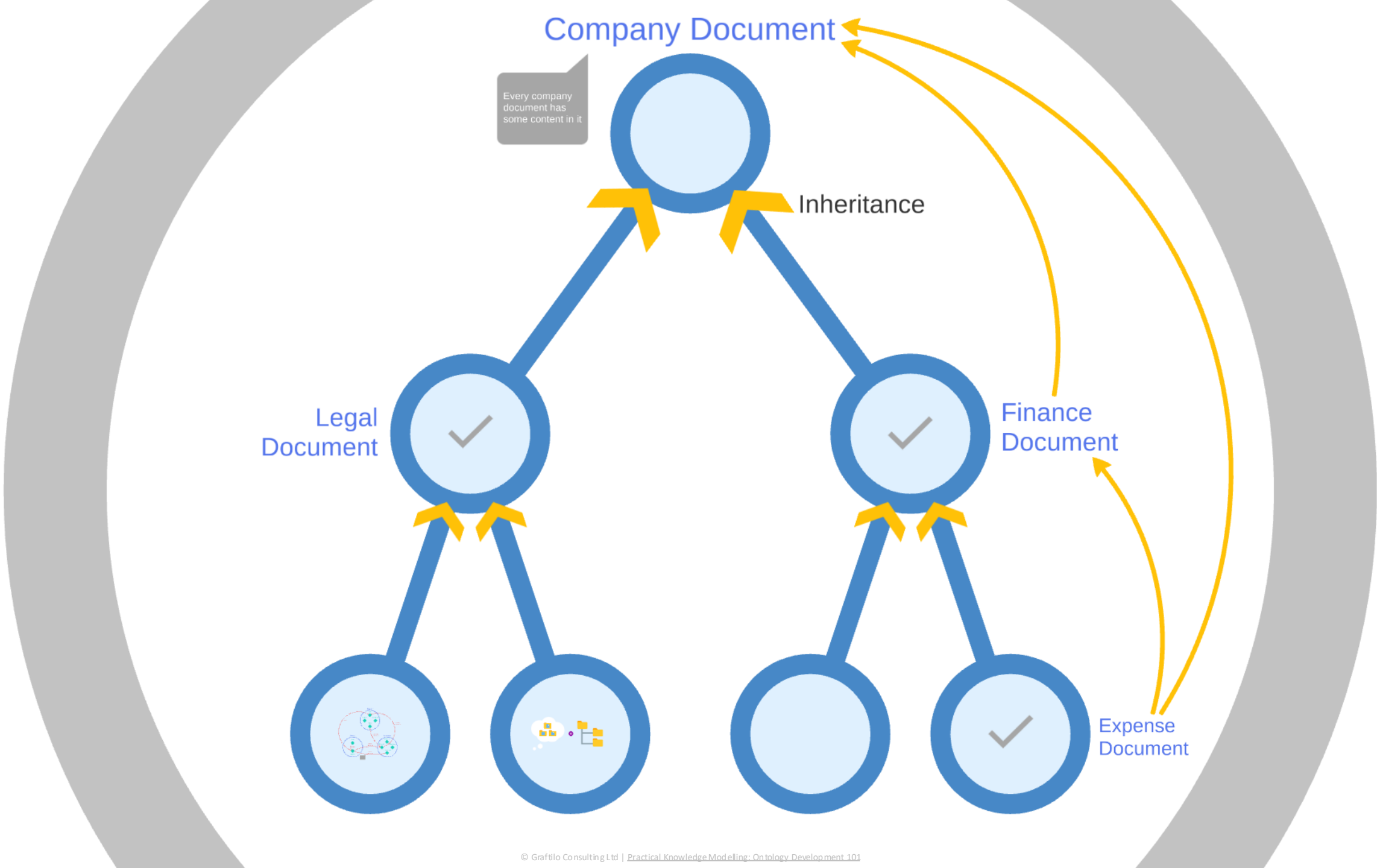
The study of being & existence

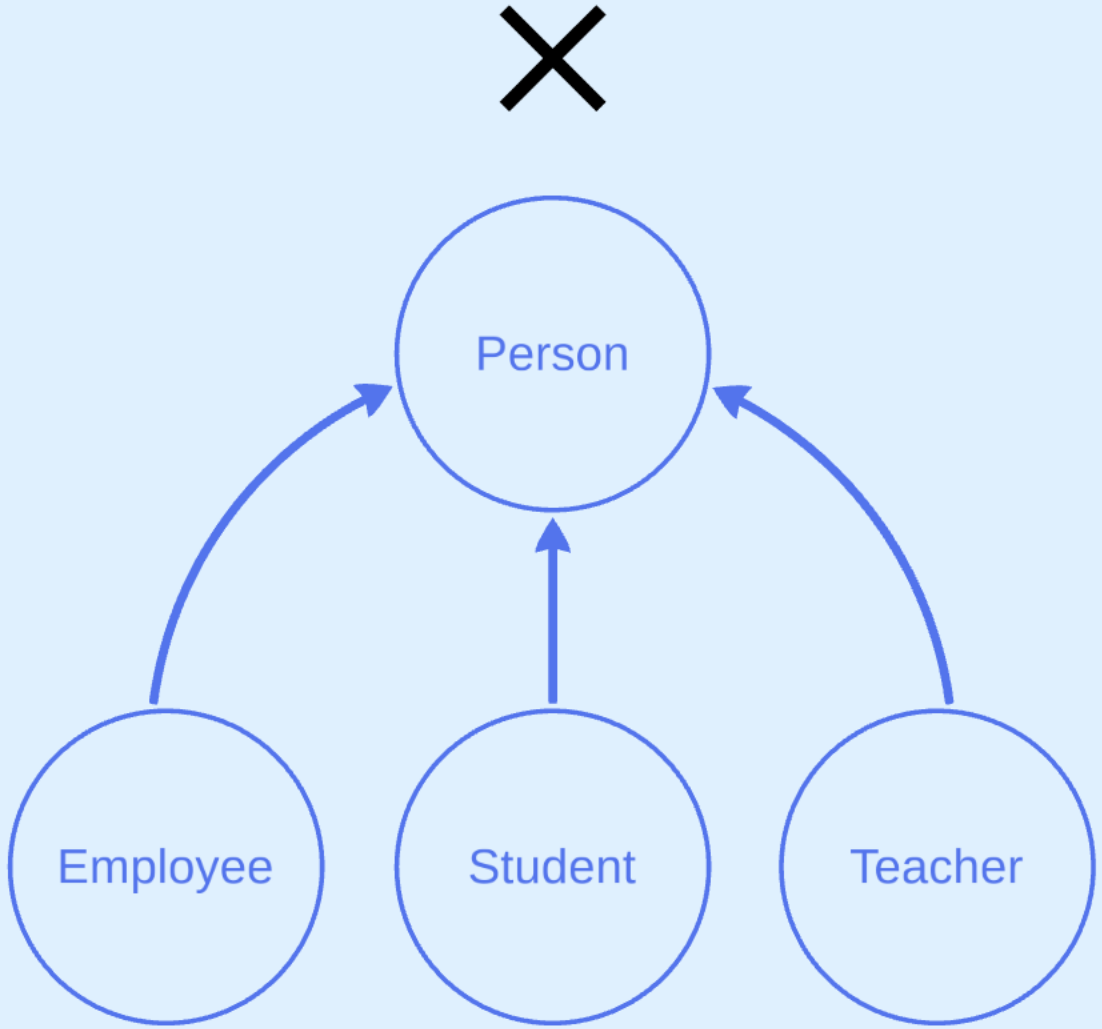
- What is the meaning of existence?
- What is an entity?
- What are its characteristics and features?
- What are the most fundamental entities and properties?
- ...

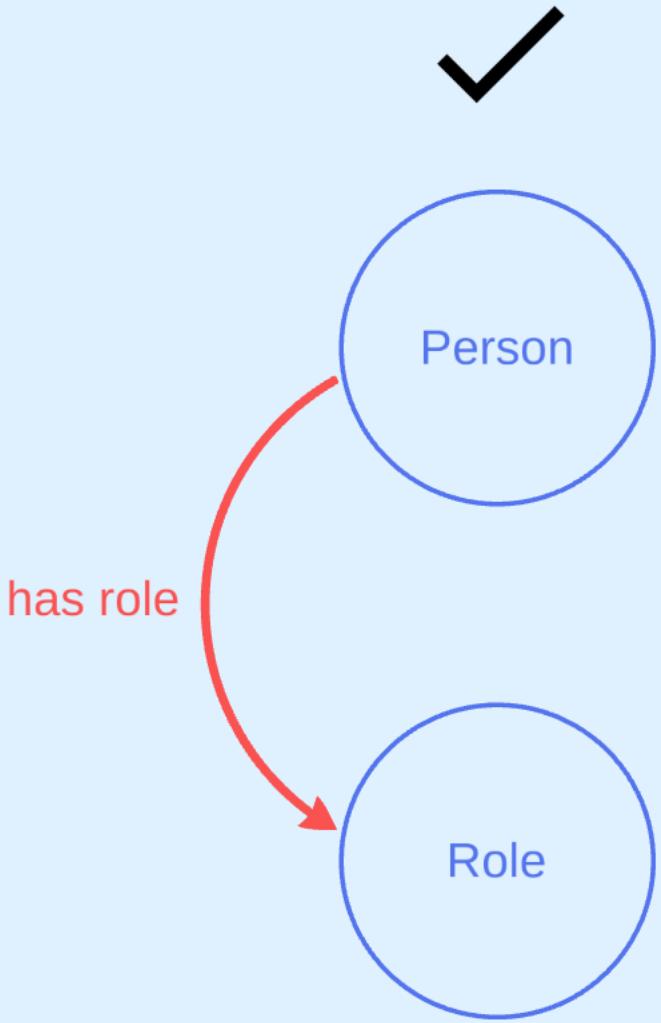

Foundational
ontologies



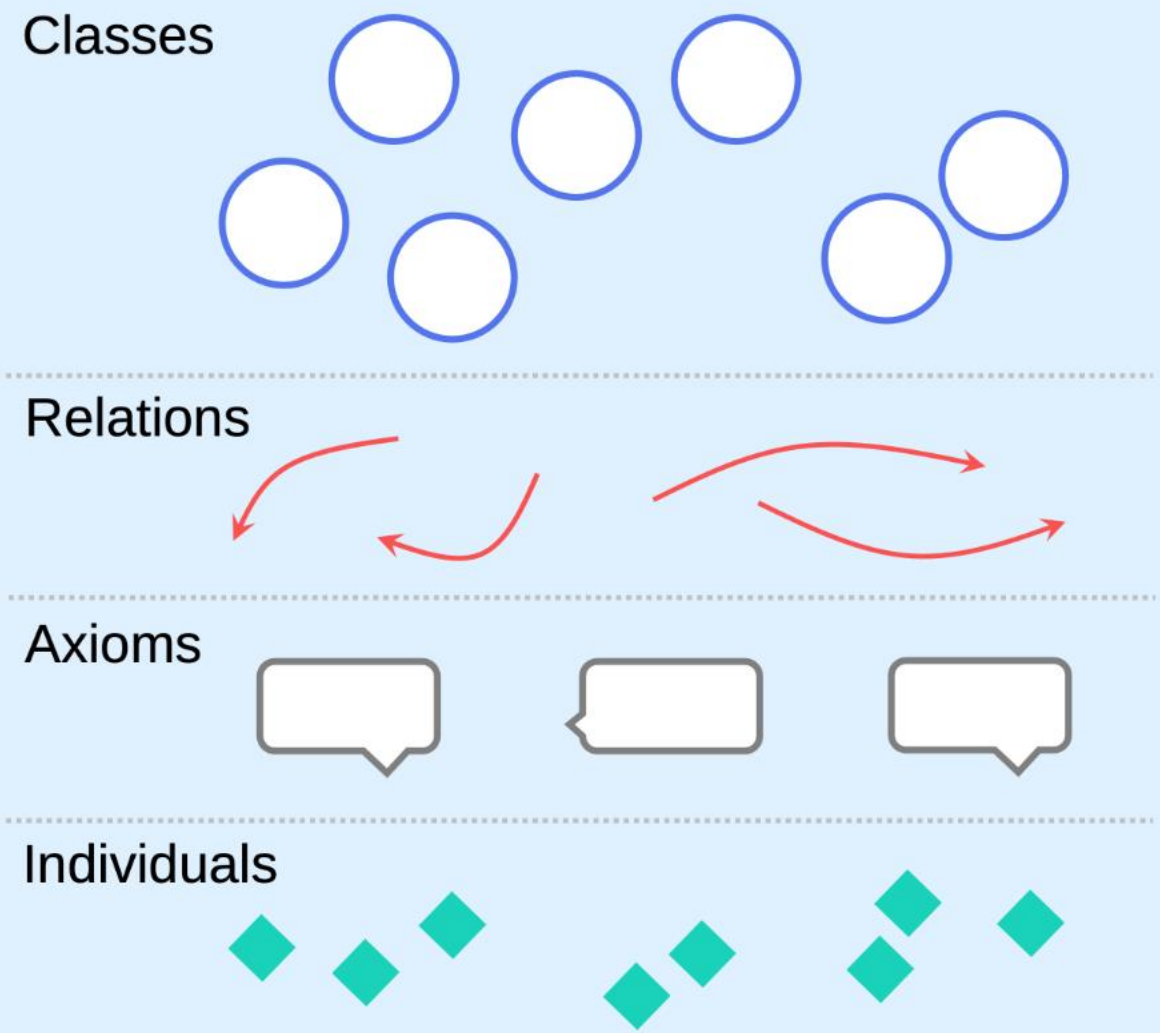



Company Document
Every company document has some content in it
Inheritance
Legal Document
Finance Document
Expense Document


Person
Employee
Student
Teacher


Person
has role
Role

Universe of discourse



Visual representation

- Good for sharing between humans
- Less formal
- Less rigorously defined

Coded representation


Lightweight
ontologies

Visual representation

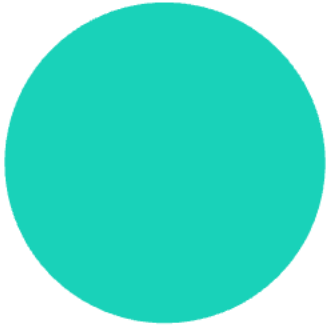
- Good for sharing between humans
- Less formal
- Less rigorously defined

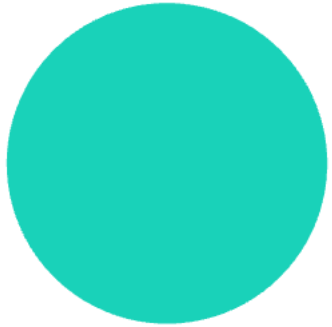
Coded representation

- Machine readable & formal
- Knowledge representation languages

Web Ontology Language

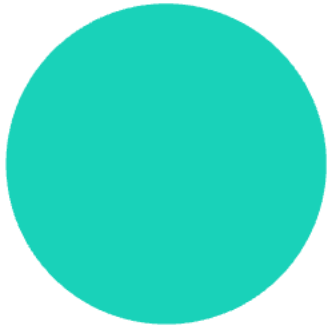
W3C, 2004





Common Logic

ISO/IEC 24707:2007



Knowledge Interchange Format

Genesereth & Fikes, 1992

Expressivity

Semantics

Visual representation

- Good for sharing between humans
- Less formal
- Less rigorously defined

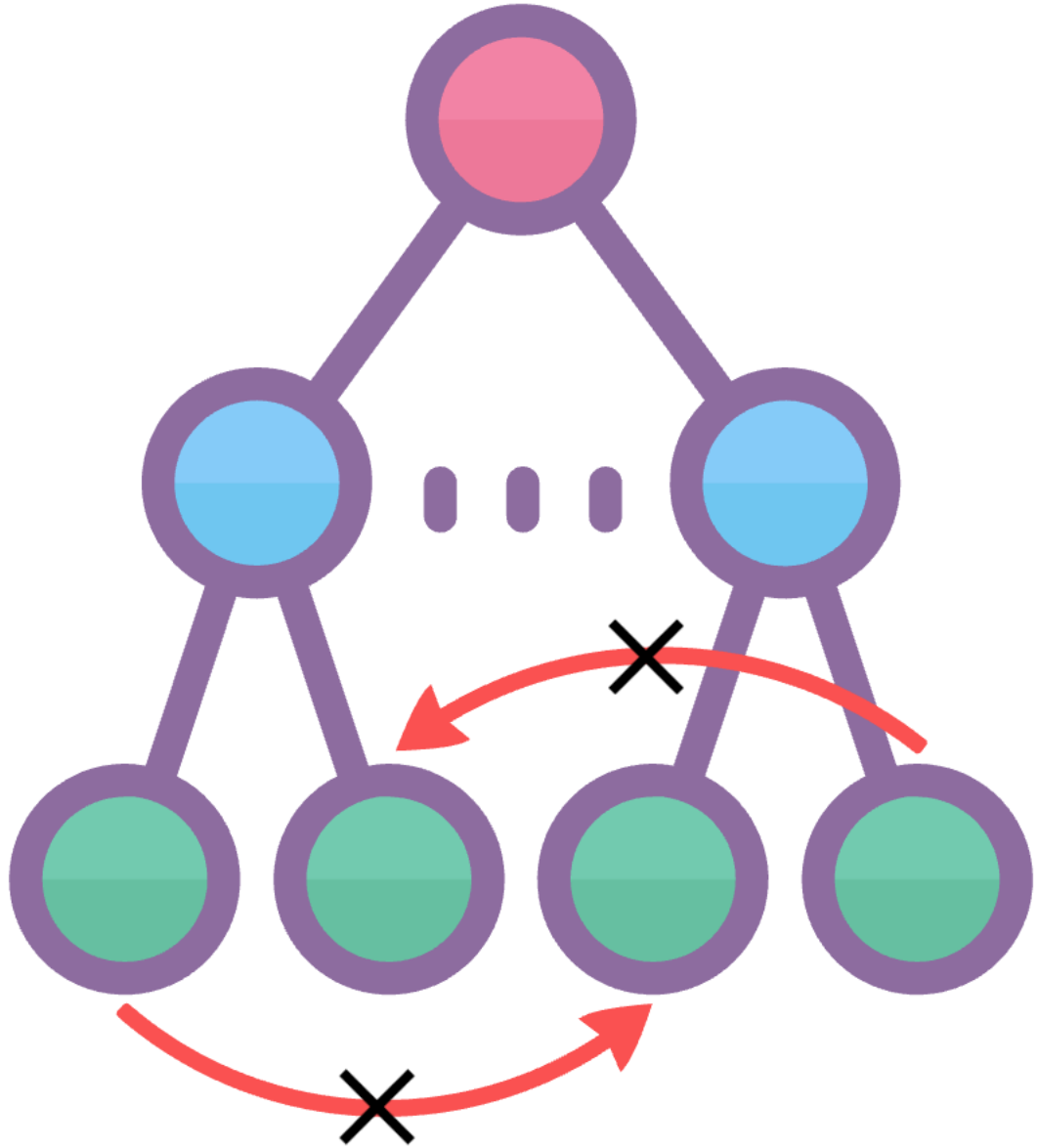
Coded representation

- Machine readable & formal
- Knowledge representation languages

Mathematical rigour

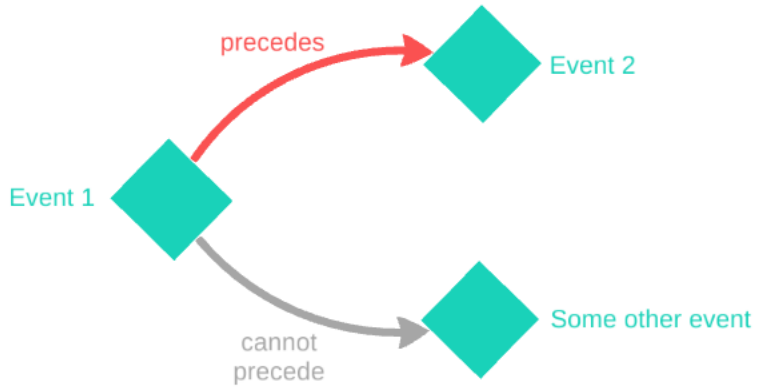
Expressivity

Range of constructs for explicit representation



Semantics

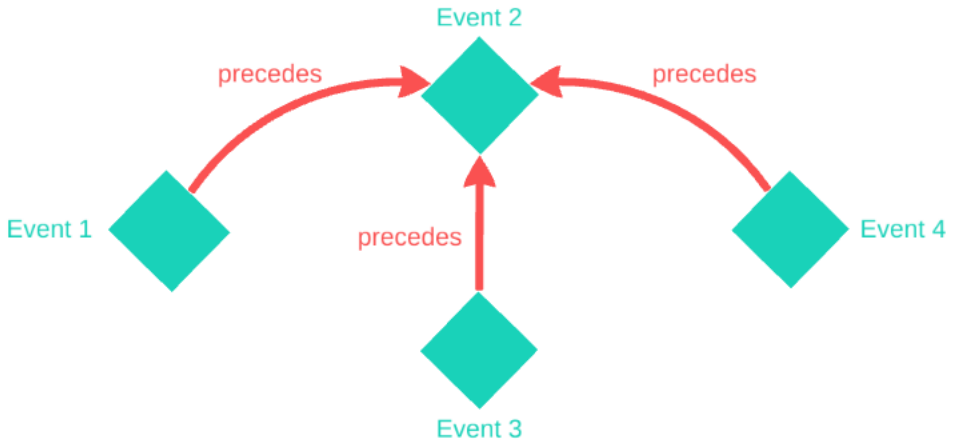
Clarity of meaning in a language specification

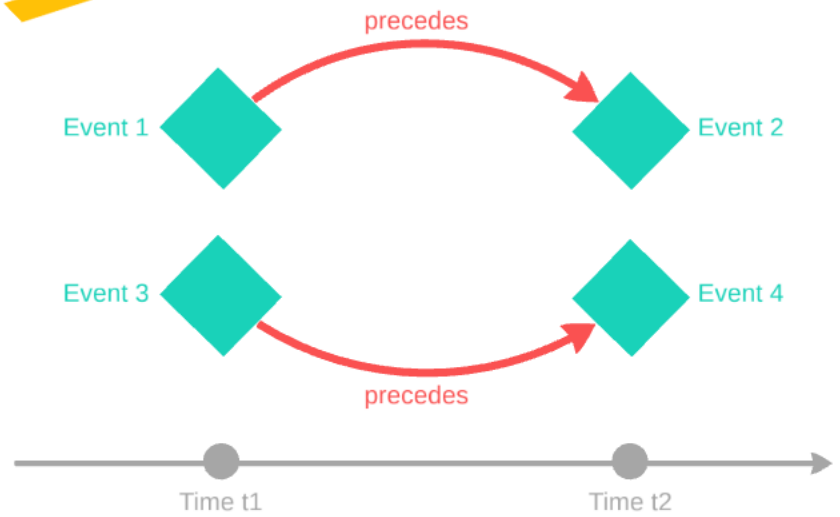

precedes
Event 2
Event 1
Some other event
cannot
precede

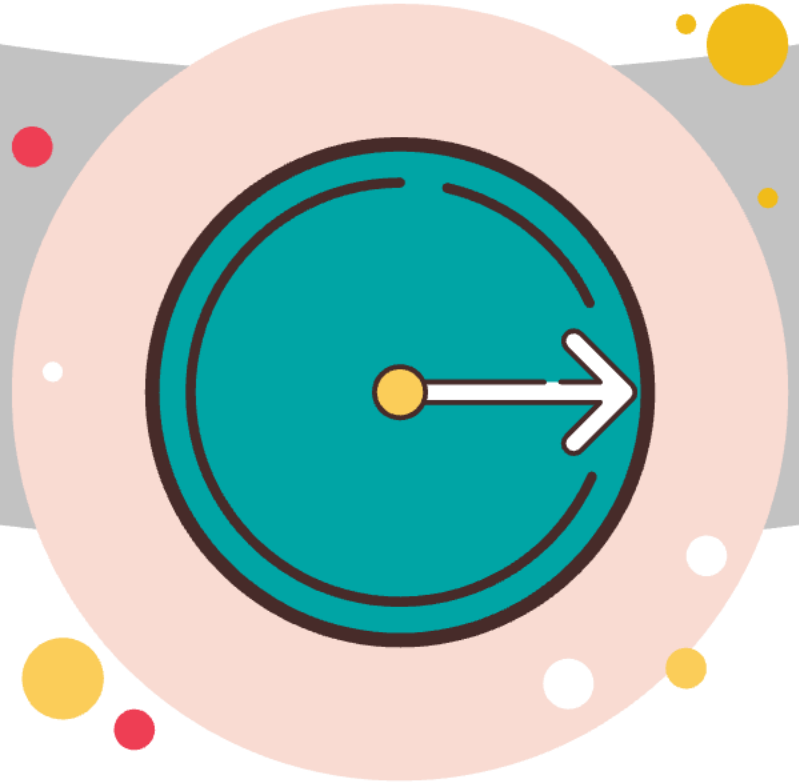
?

Precedence

? ?


Event 2
precedes
precedes
Event 1
precedes
Event 4
Event 3


precedes
Event 1
Event 2
Event 3
Event 4
precedes
Time t1
Time t2

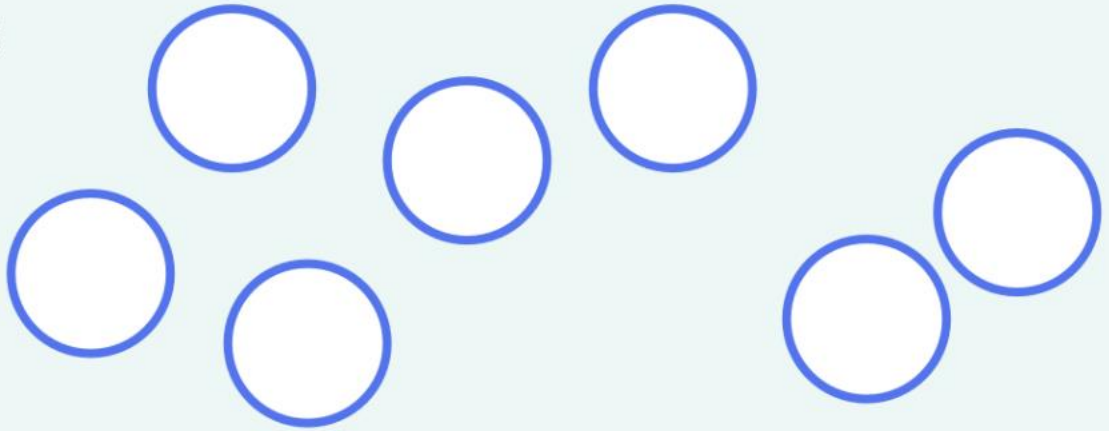


Mathematical rigour

Integrity & soundness provided by a language

Heavyweight ontologies

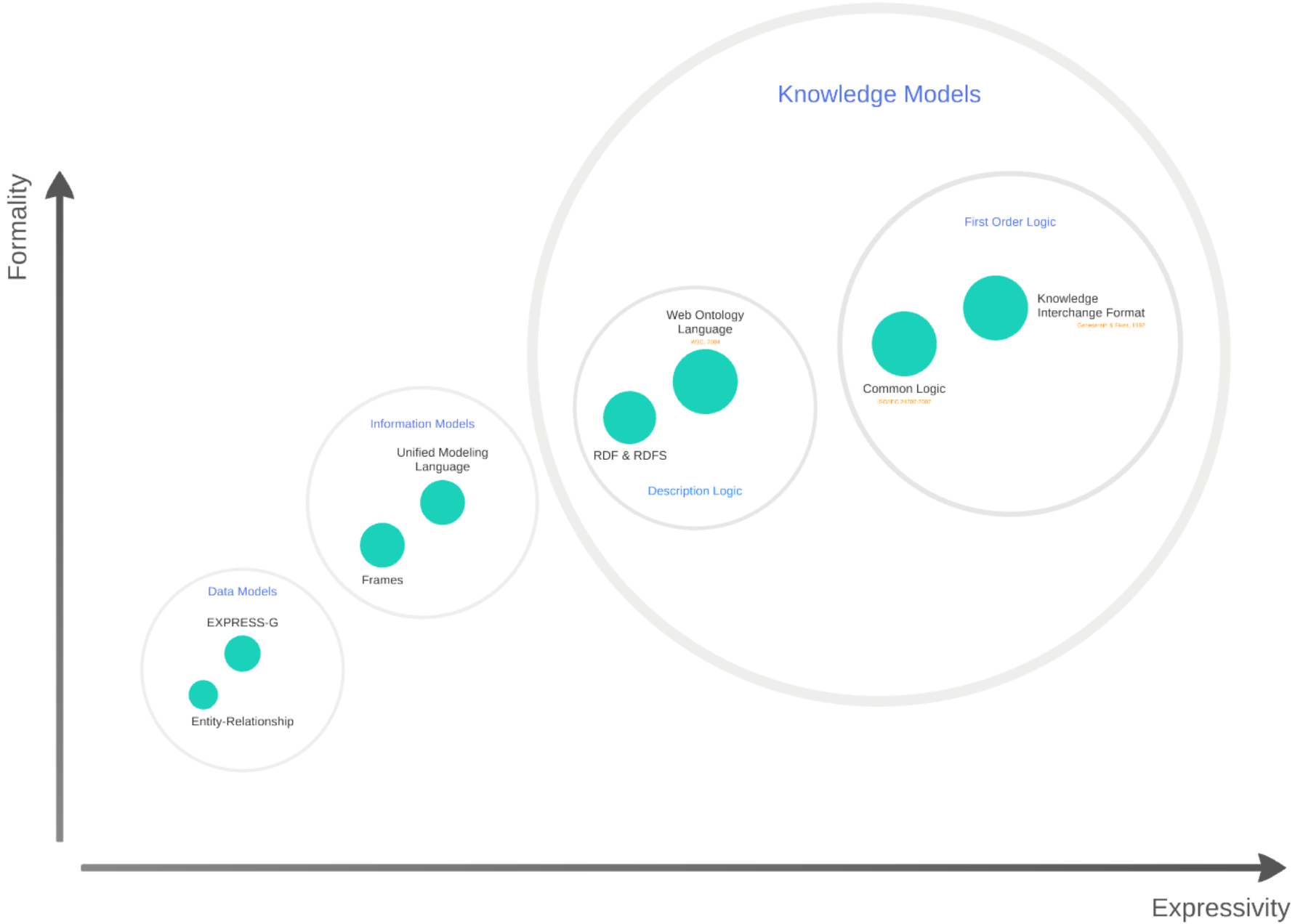
Classes

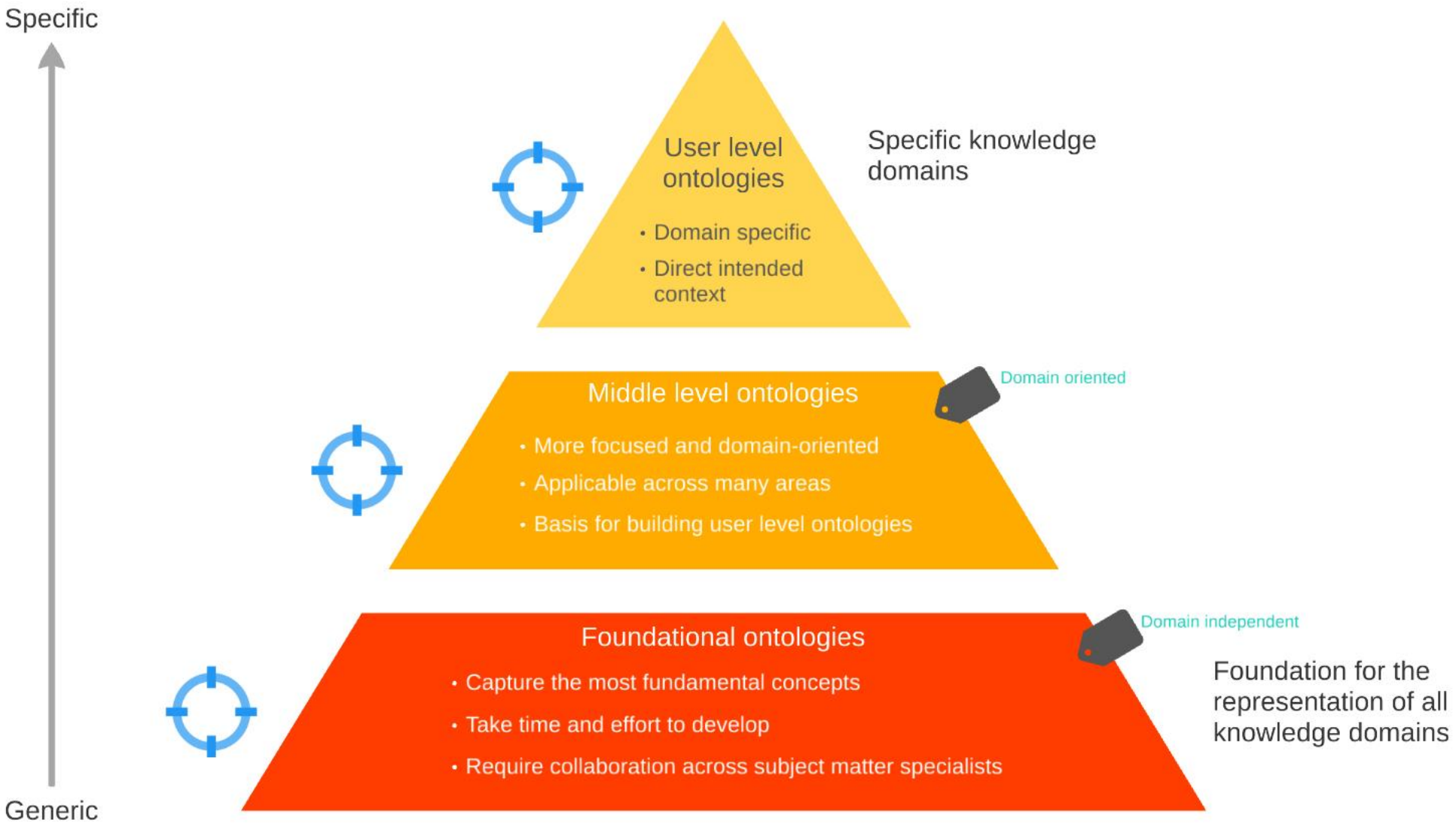


Relations

Axioms

Individuals


Knowledge Models
First Order Logic
Knowledge
Interchange Format
Common Logic
Web Ontology
Language
RDF & RDFS
Description Logic
Information Models
Unified Modeling
Language
Frames
Data Models
EXPRESS-G
Entity-Relationship
Formality
Expressivity


Specific
Generic
User level ontologies
• Domain specific
• Direct intended context
Specific knowledge domains
Middle level ontologies
• More focused and domain-oriented
• Applicable across many areas
• Basis for building user level ontologies
Domain oriented
Foundational ontologies
• Capture the most fundamental concepts
• Take time and effort to develop
• Require collaboration across subject matter specialists
Domain independent
Foundation for the representation of all knowledge domains

SUMO

DOLCE

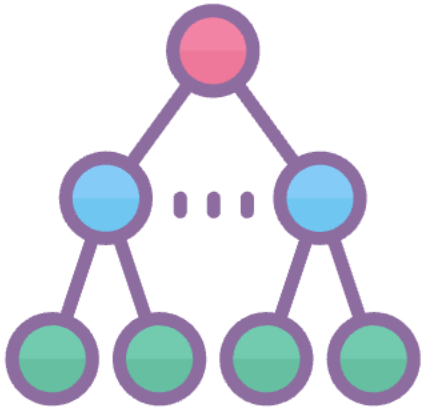
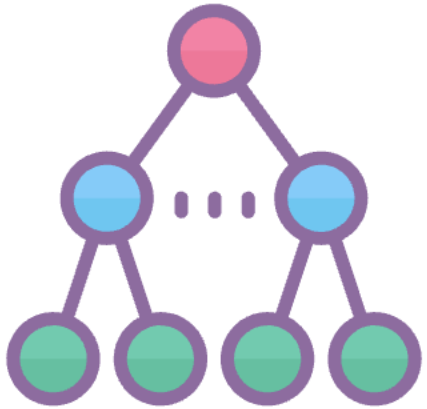
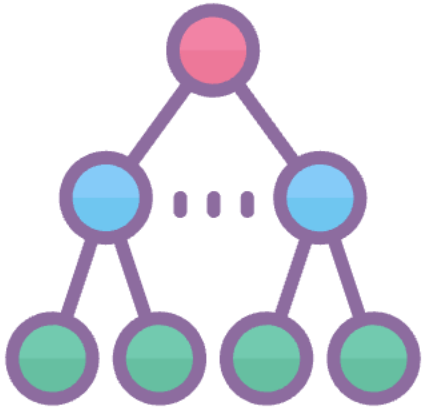
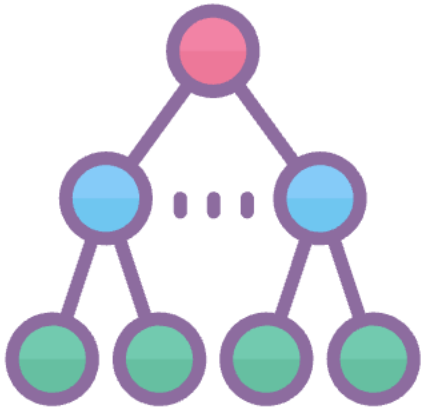
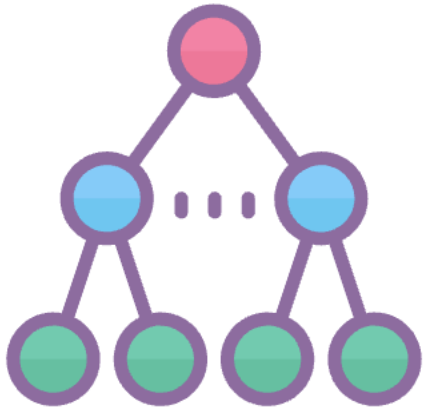
gist

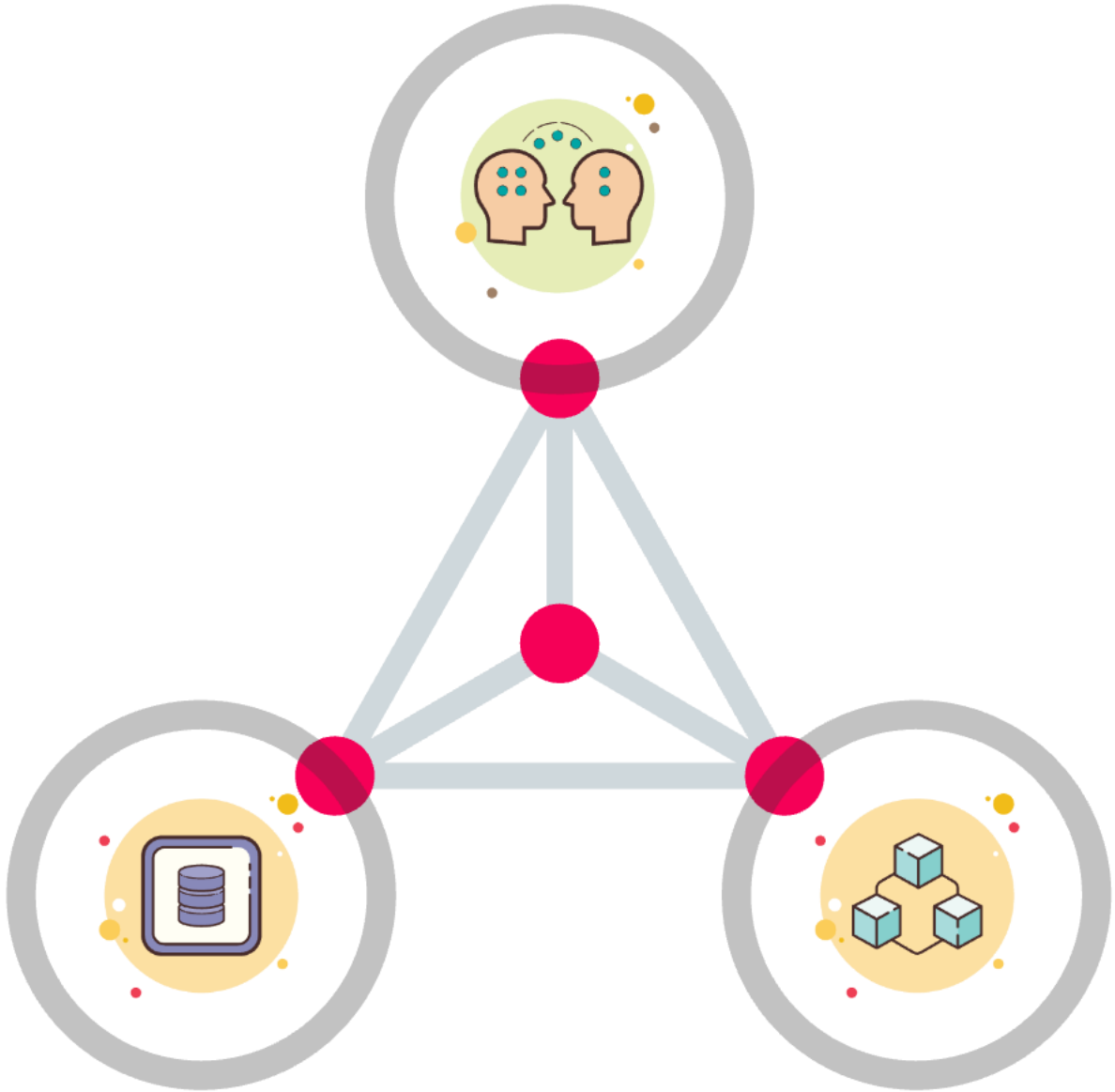
BFO

PSL

The Zachman Framework

The Music Ontology



Share meaning & knowledge

- Reuse domain knowledge
- Domain assumptions explicitly specified

Information systems & AI

- Systems interoperability
- Vast range of useful applications

Semantic Web

NLP

Intelligent search

Digital twins

Medical knowledge representation

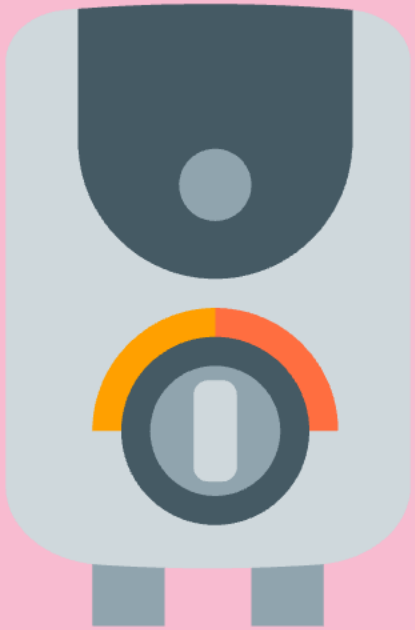
Software development

Intelligent document retrieval

Problem solving & decision-making

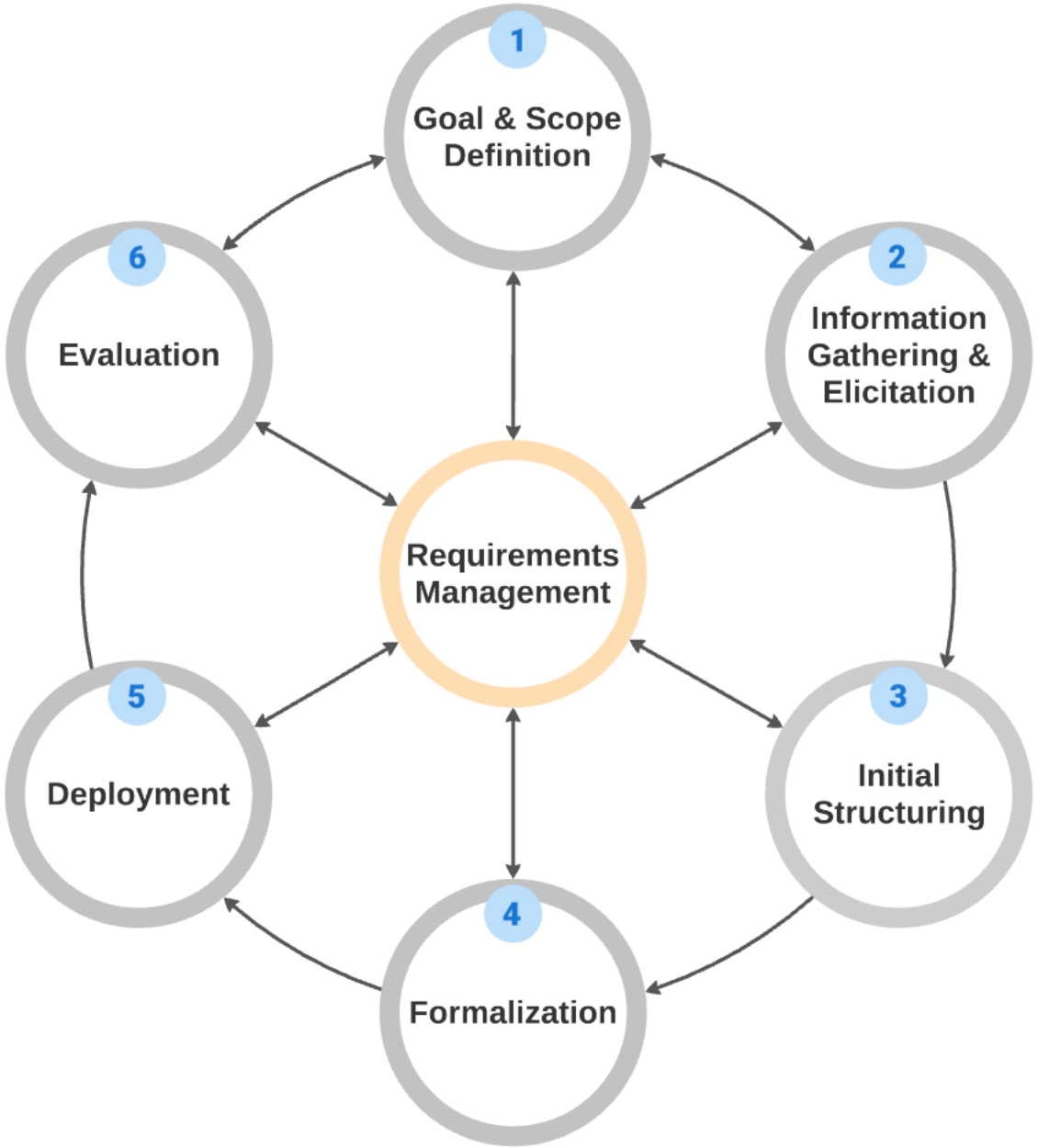
- Knowledge bases (graph databases)
- Querying, reasoning & knowledge retrieval

Service knowledge for combination boilers



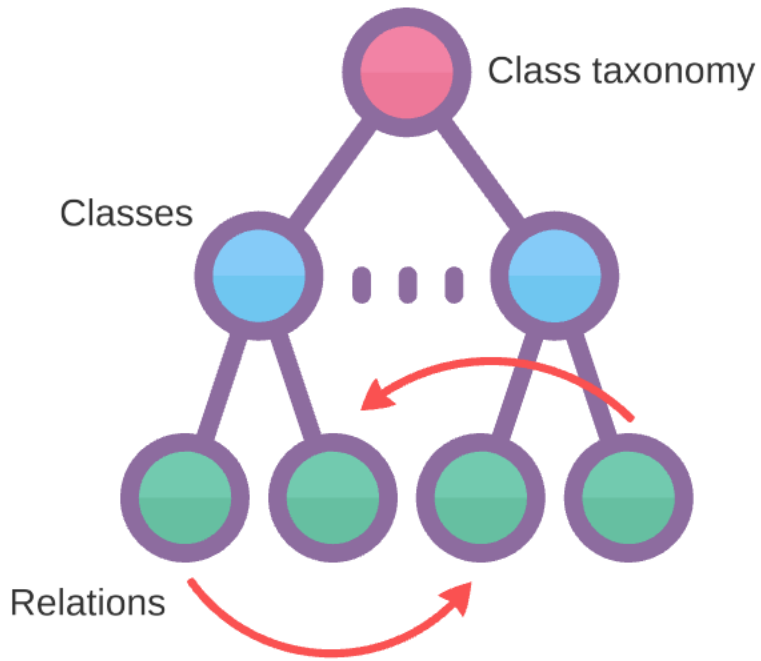

Share meaning
& knowledge
• Reuse domain knowledge
• Domain assumptions explicitly specified
Applied ontology
Problem solving & decision-making
• Knowledge bases (graph databases)
• Querying, reasoning & knowledge retrieval
Information systems & AI
• Systems interoperability
• Vast range of useful applications

03 A methodology for knowledge modelling

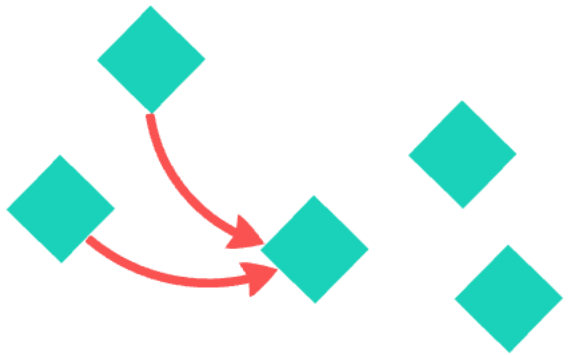


- Covers full ontology development cycle
- Does not make any assumptions on development timescales
- Independent of tools
- Independent of subject matter

Knowledge model (ontology)



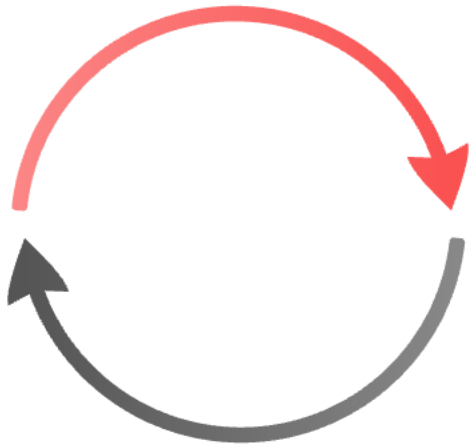
Knowledge base (graph database)



Initial structuring

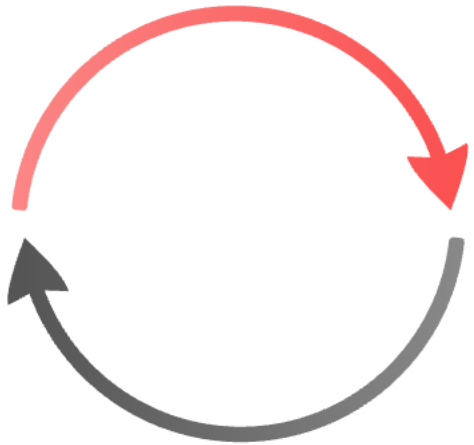
Formalization

Deployment



Simple Knowledge Engineering Methodology

Protégé



IDEF5 Ontology Description Capture Method

Integration DEFinition Methods

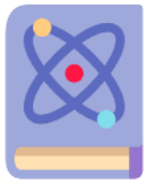
Requirements Management

- List competency questions
- Generate ontology documentation
- ...
- Encode heavyweight ontology in OWL
- Publish ontology on company portal
- ...

- Provides the 'glue' across phases
- Promotes visibility and clarity
- Helps avoid 'over-the-wall' development

Goal & Scope Definition

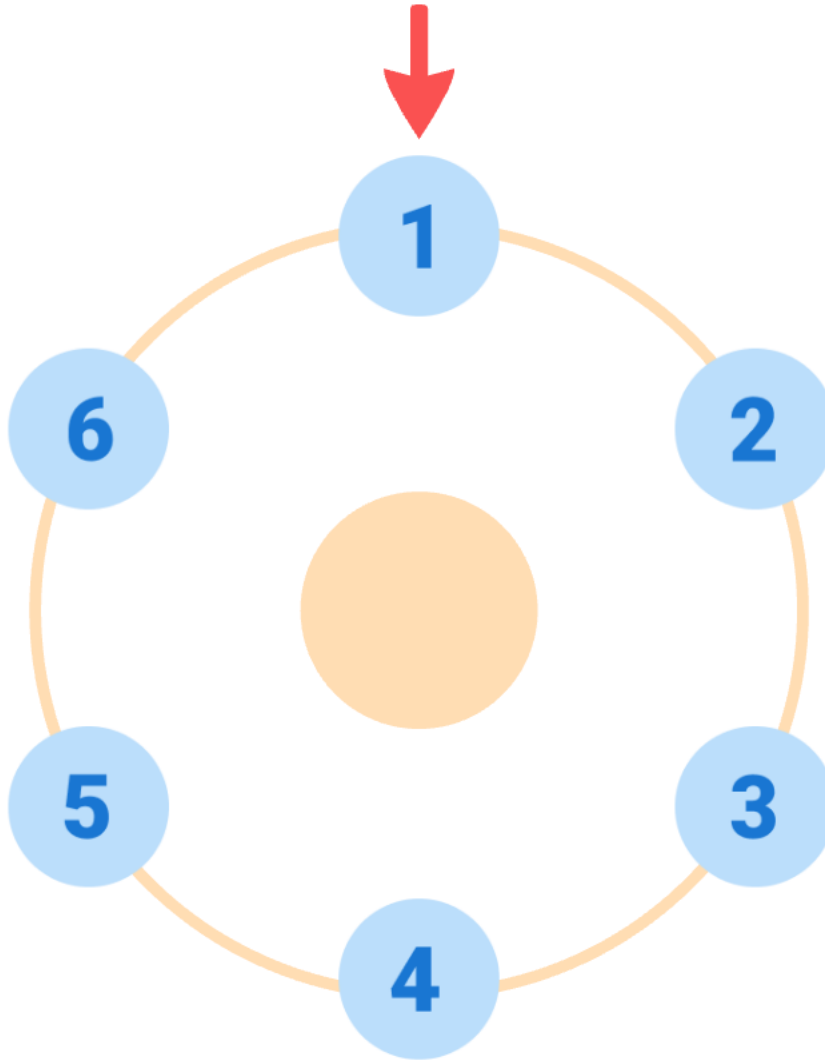
Domain of interest



Aims & objectives

Scope

Roles



To develop and understanding of the different types of ballpoint pens by gathering and organising information from various sources

To derive and build a lightweight ontology (in diagrammatic form) for review purposes and to be shared with non-experts

To derive and build a corresponding heavyweight ontology

To publish the heavyweight ontology to a test server (for testing purposes) followed by publishing to the company portal system upon successful testing

To evaluate the effectiveness of the ontology by collating and analysing fortnightly feedback from users over a period of two months after deployment

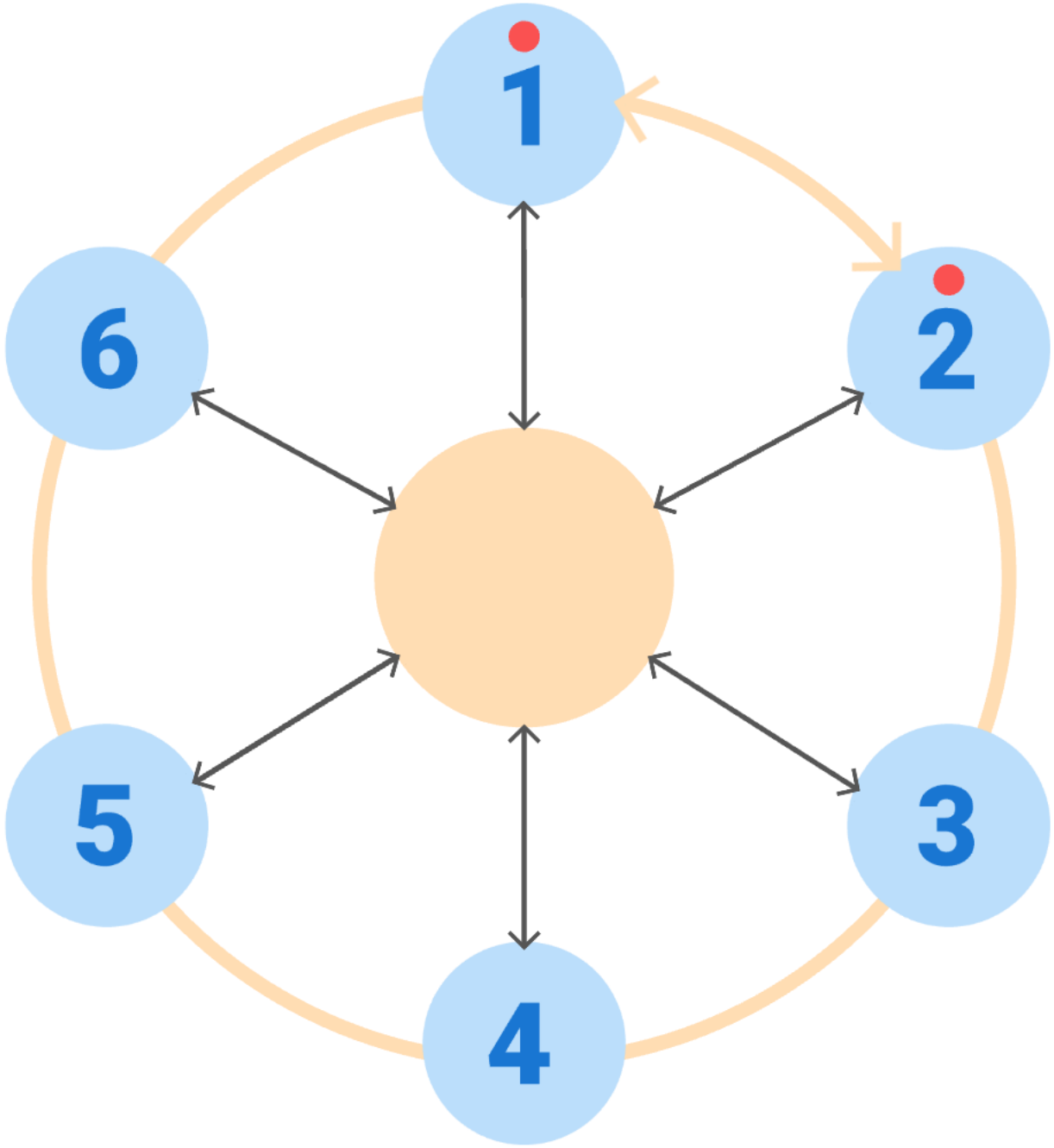
Information Gathering & Elicitation

Collect information

Target several sources

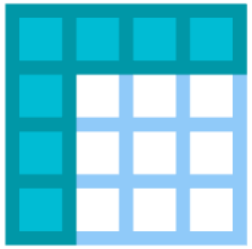
Apply suitable methods

Method	Depth of information	Cost	Time efficiency	Ease of following up	Level of structure supported
Brainstorming					
Survey					
Interview					
Focus group					
Content analysis					

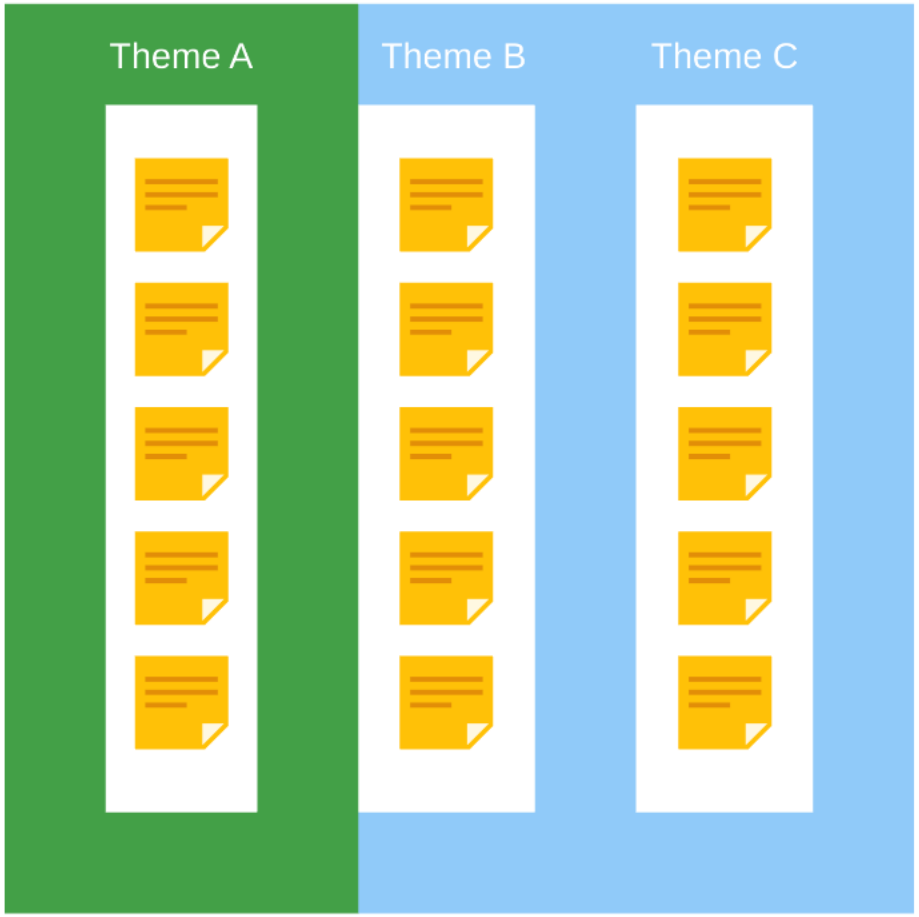

1
2
3
4
5
6


Collation of information

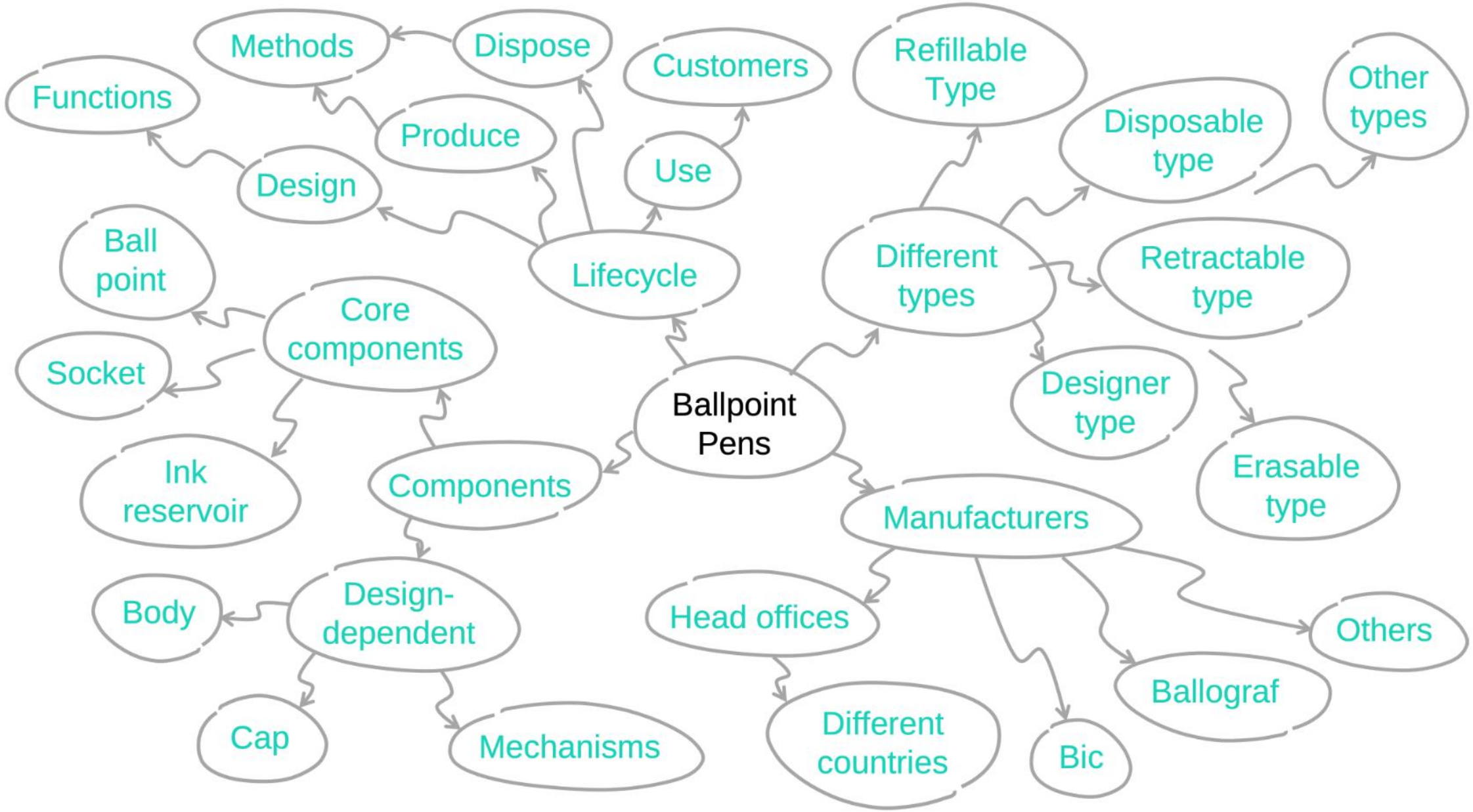
Collation of information



Affinity diagram



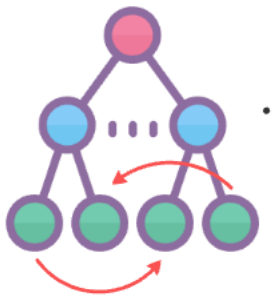
- Useful for group-based information gathering
- Sorts unorganised ideas, terms & concepts
- Visual technique


Ballpoint Pens
Lifecycle
Methods
Dispose
Customers
Functions
Produce
Use
Design
Core components
Ball point
Socket
Ink reservoir
Components
Design-dependent
Body
Cap
Mechanisms
Different types
Refillable Type
Disposable type
Other types
Retractable type
Designer type
Erasable type
Manufacturers
Head offices
Different countries
Bic
Ballograf
Others

04 Initial structuring

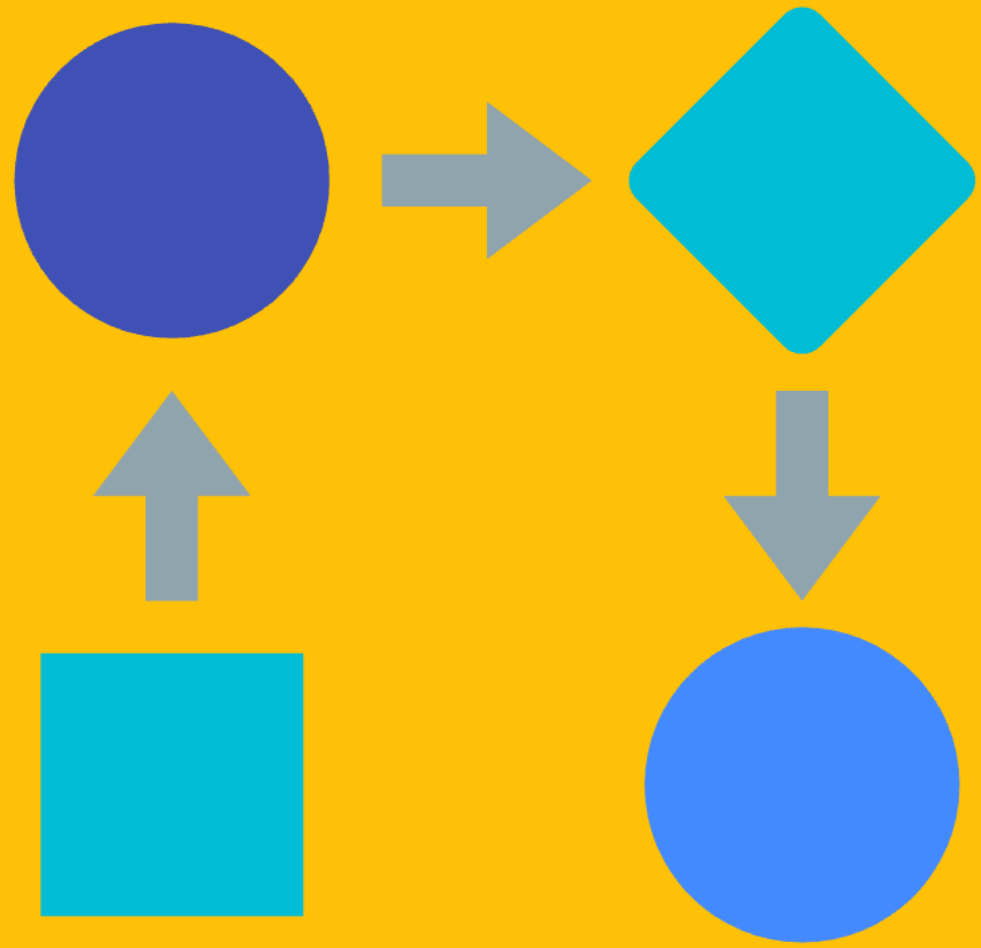
Initial Structuring

- Transform loosely organised information



- Produce visual lightweight ontologies

- Visualise entities
- Review & share
- Streamline formalization

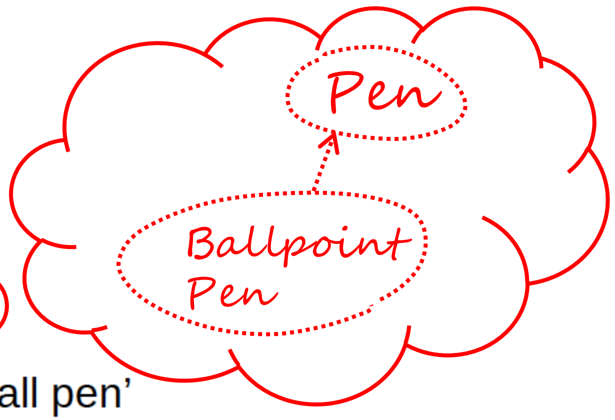


A ballpoint pen, also known as “biro” and “ball pen”, is a pen that dispenses ink over a metal ball at its point, i.e. over a “ball point”

Wikipedia

Listing and analysing statements

Ballpoint Pen Example



General

- A pen is a writing implement/instrument
- Ballpoint pen is a pen
- Ballpoint pen is also known as ‘biro’ and ‘ball pen’
- Ballpoint pen dispenses ink over a metal ball at its point, i.e. over a ‘ball point’
- Ballpoint pen is manufactured
- Most ballpoint pens rely on gravity to coat the ball with ink
- Most ballpoint pens cannot be used to write upside-down

Types of ballpoint pens

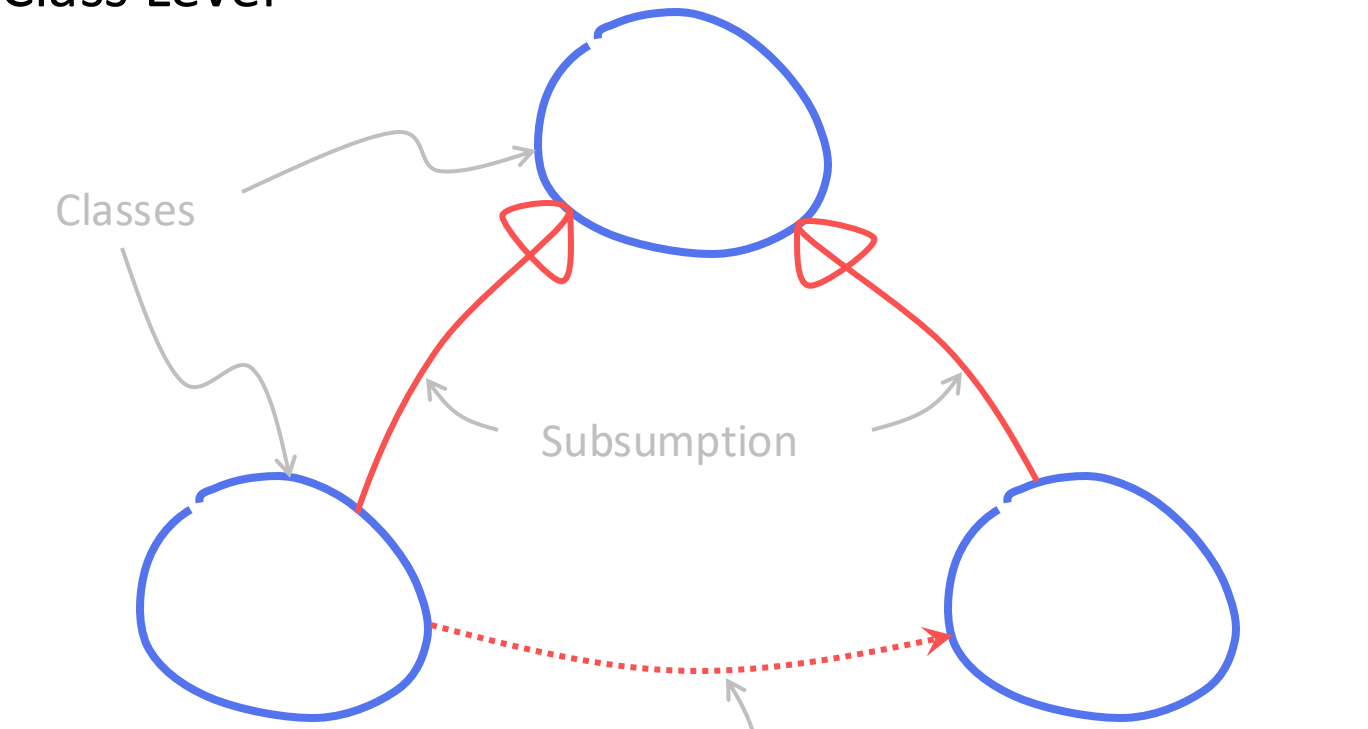
- Designer ballpoint pen
  - Intended for high-end and collectors’ markets
  - Is produced by manufacturers
- Bic Cristal is a type of ballpoint pen
  - Is disposable
  - Most widely sold pen worldwide
  - Cap has a hole to prevent suffocation

    - Can be a characteristic for designer ballpoint pens
- Rollerball pen
    - Uses water-based inks instead of oil-based inks
- Space Pen
    - A.k.a. Zero Gravity Pen
    - Overcomes limitation of common ballpoint pens
    - Uses more viscous ink with pressurized ink reservoir
- Ballpoint pen with erasable ink
    - Pioneered by Paper Mate pen company
    - Ink has special formula
    - Ink is much thicker than standard ballpoint inks
    - Cartridge is pressurized for better ink flow
    - Can be used to write upside-down

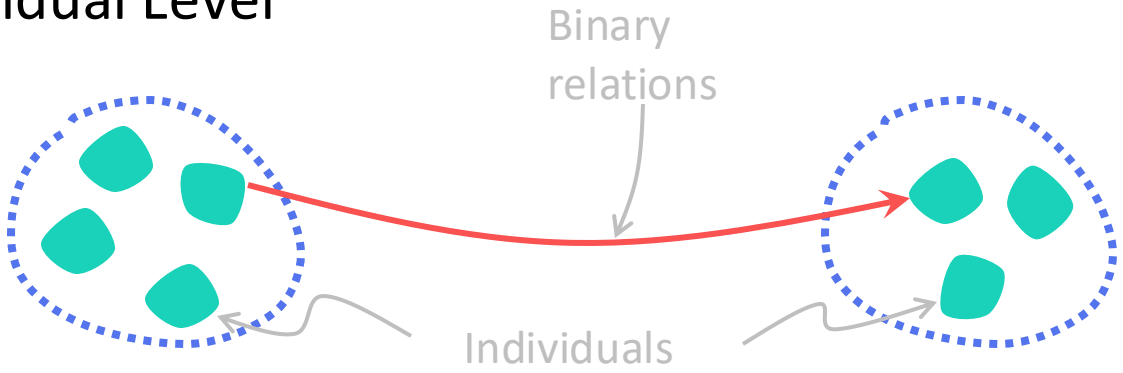
Parts and manufacturing

- Disposable type
    - Can have a cap to cover tip when pen is not in use
    - Can have a mechanism for retracting the tip
- Designs and constructions vary between brands
- Universal basic components
    - Free-floating ball point – distributes ink
    - Socket – holds the ball in place
    - Self-contained ink reservoir – supplies ink to the ball

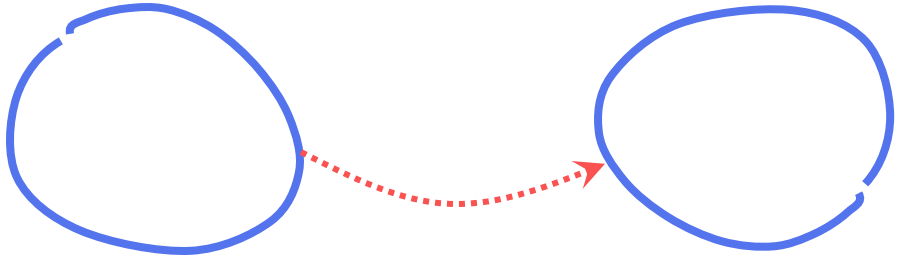
Class Level



Individual Level

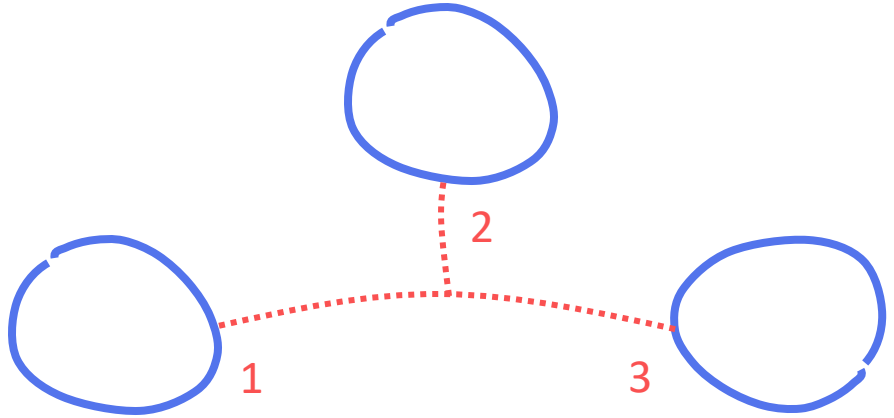


Binary Relation

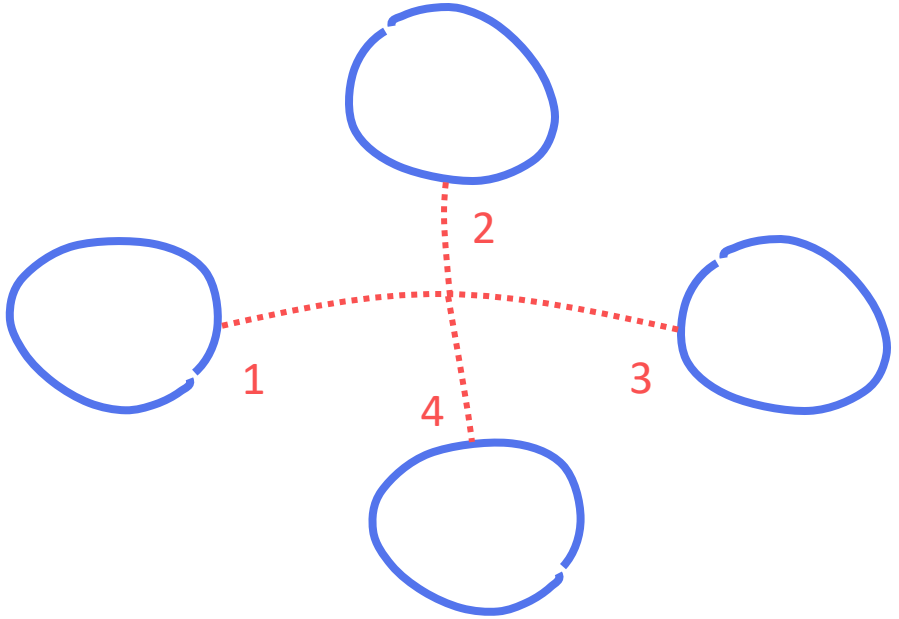


- Most common type of relation

Ternary Relation

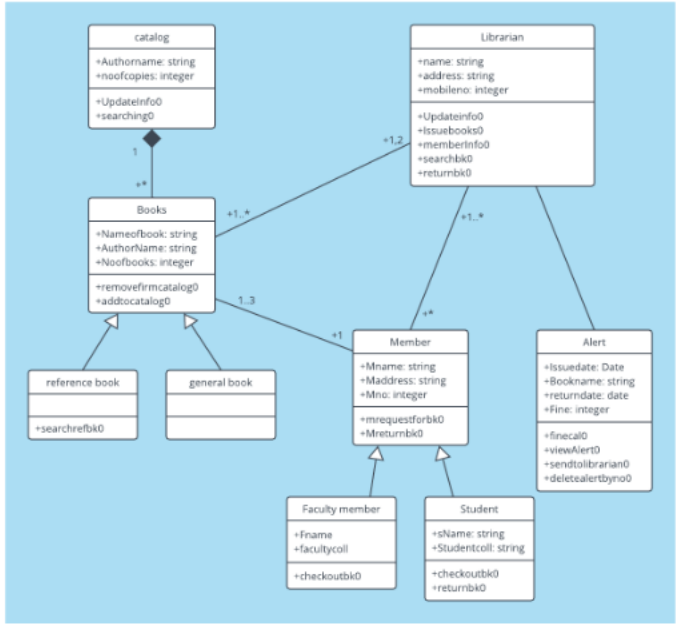


Quaternary Relation



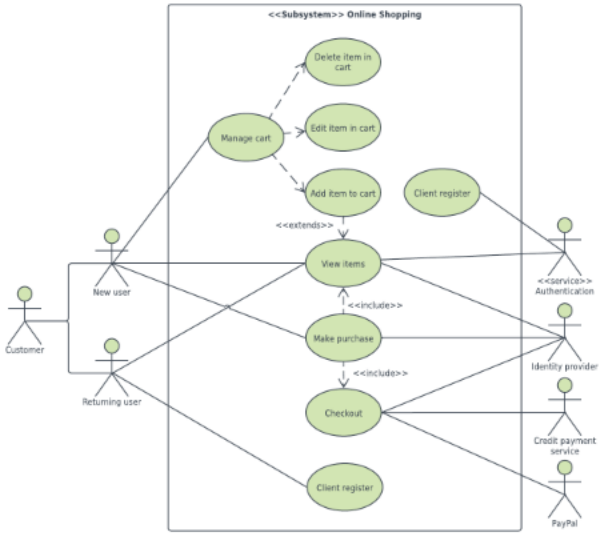
Naming Conventions

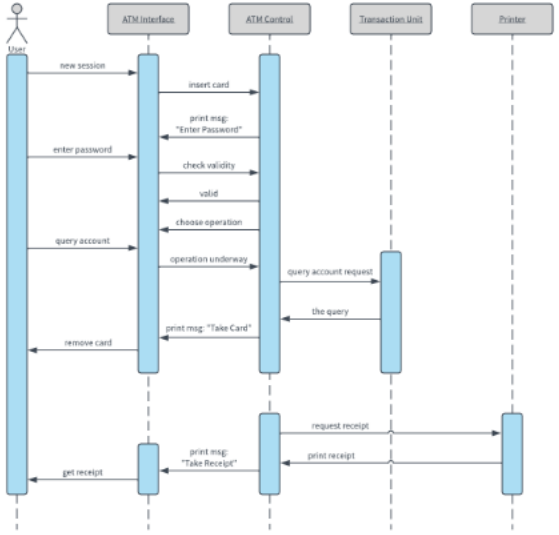
Entity	Camel Case Convention (Examples)	Underscore Convention (Examples)
Class	Pen, BallpointPen	Pen, Ballpoint_Pen
Relation	dispenses, hasComponent	dispenses, has_component
Individual	Pen001, MyFavouritePen	Pen_001, My_Favourite_Pen



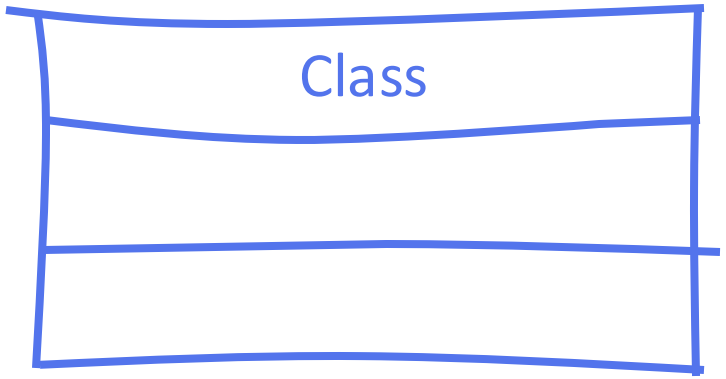
UML

- Plan & model systems
- Rich set of graphical constructs
- Capture different perspectives

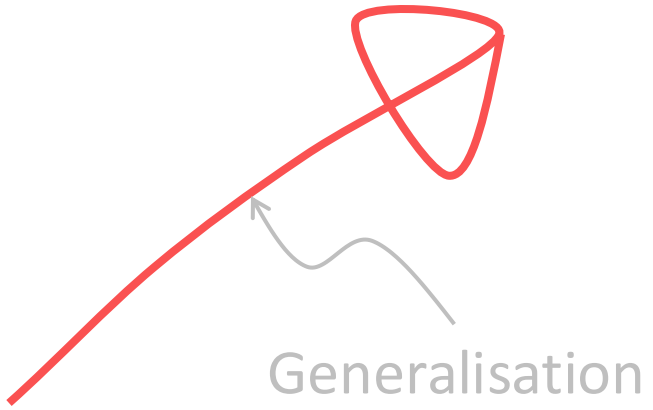




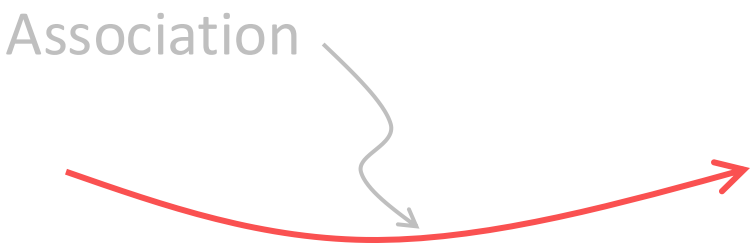
Class



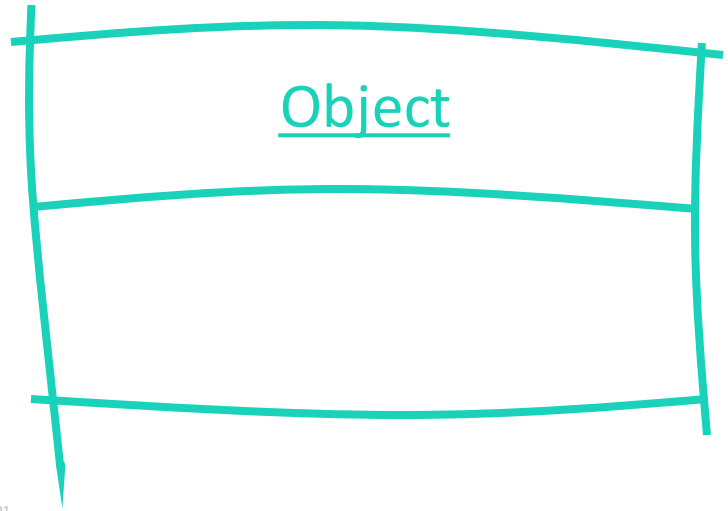
Subsumption

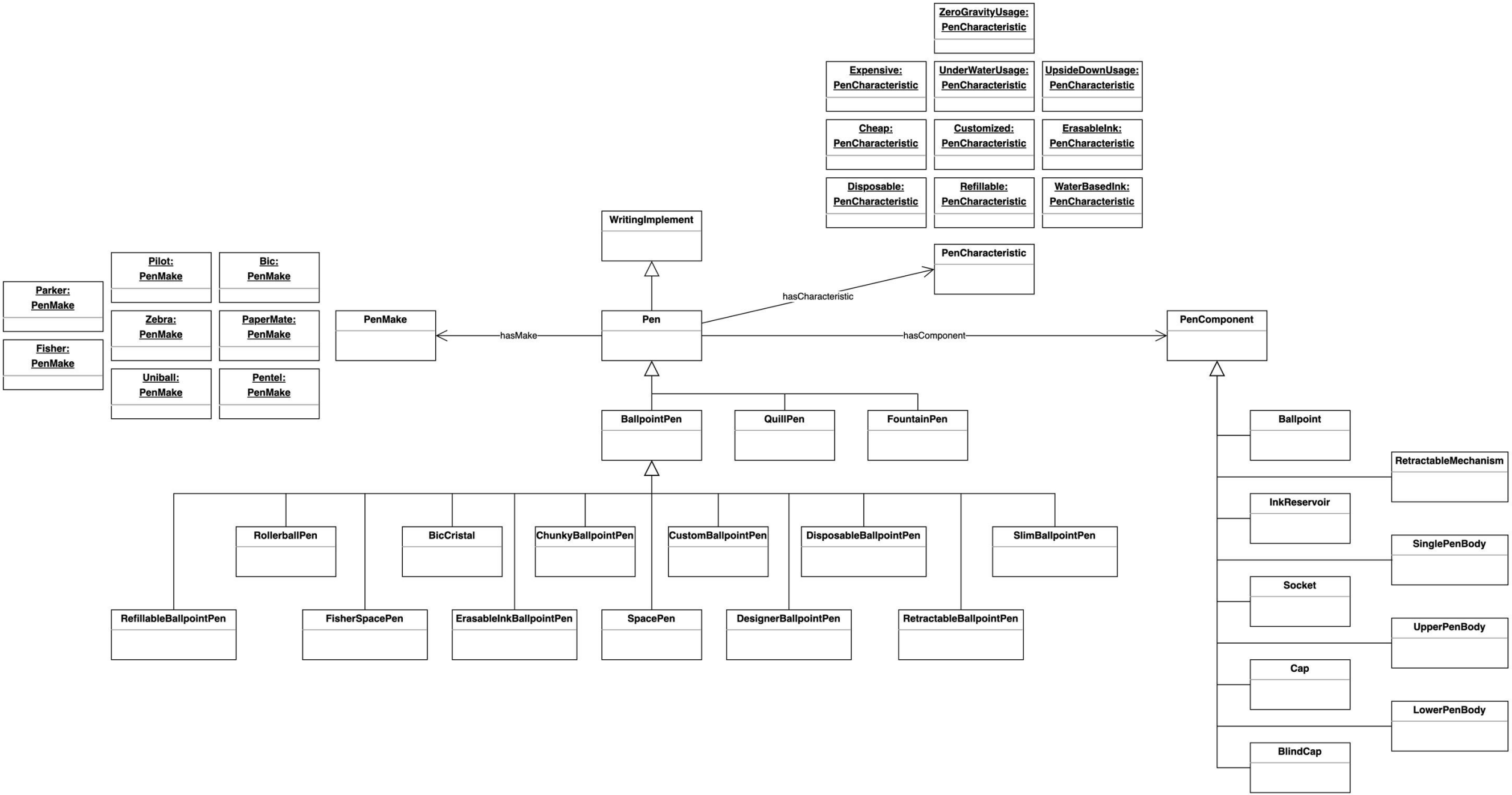


Binary Relation



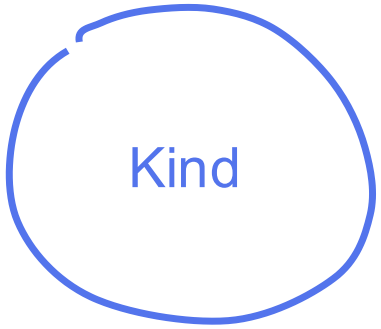
Individual



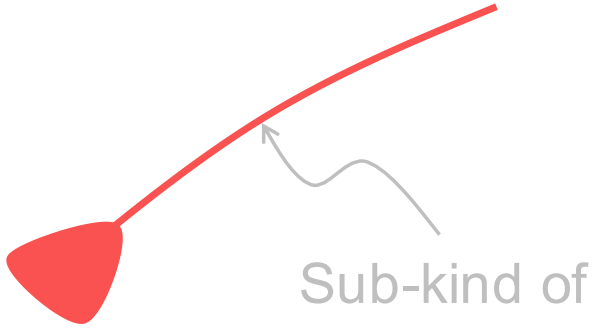


IDEF5

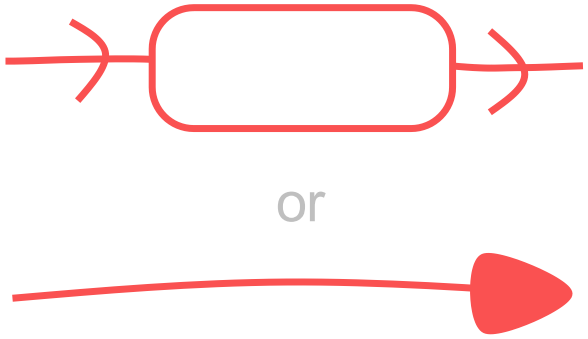
Class



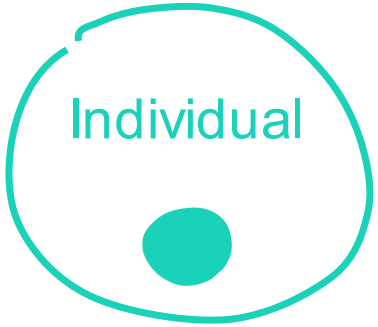
Subsumption

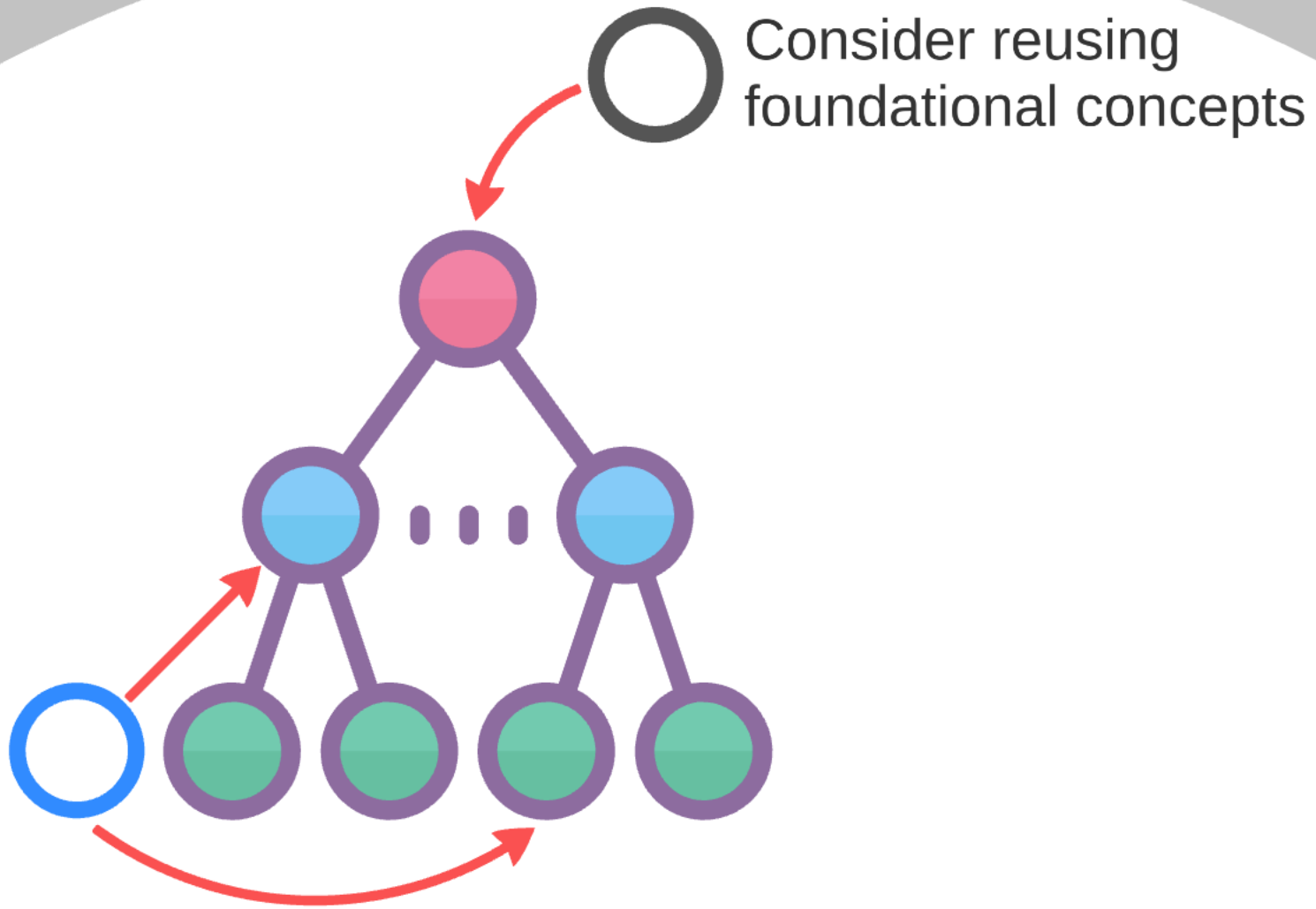


Binary Relation

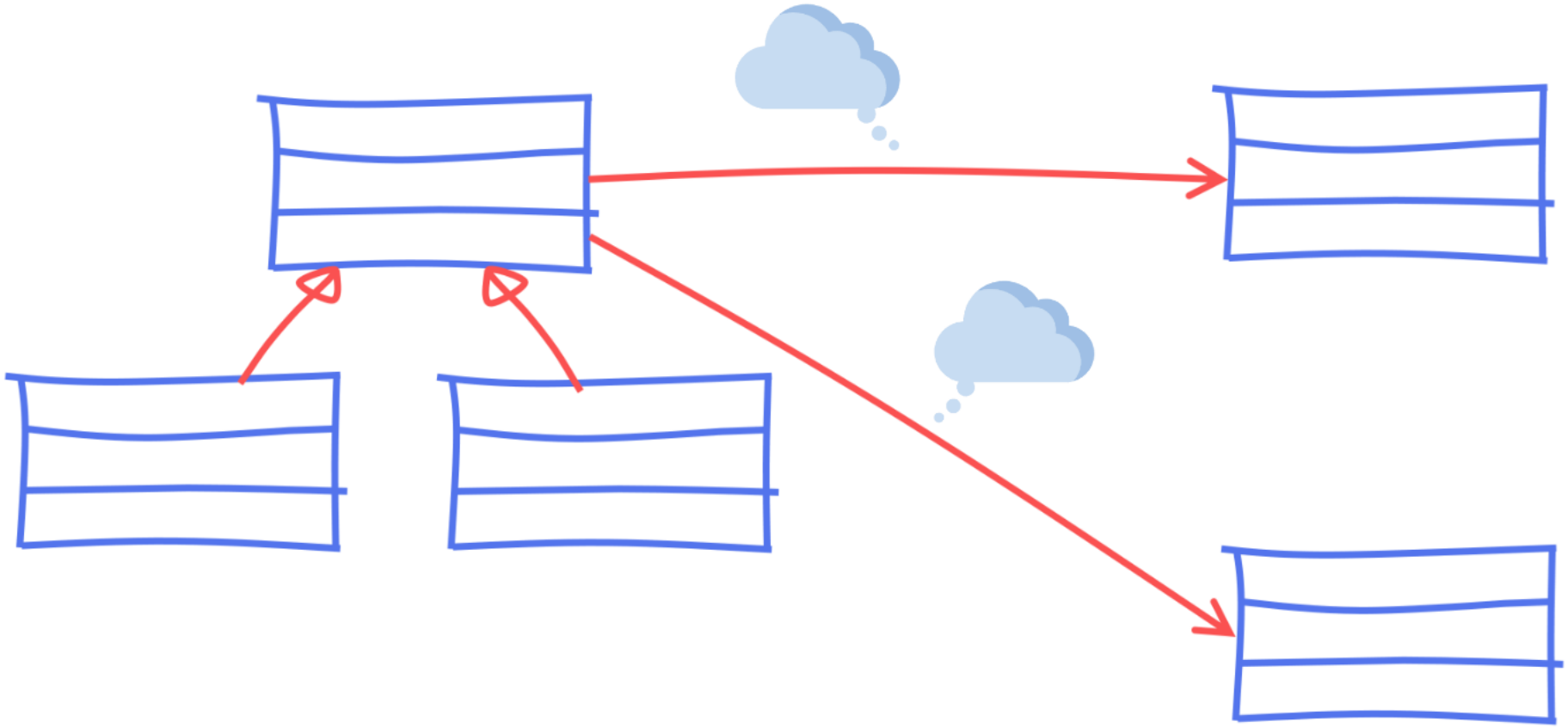


Individual



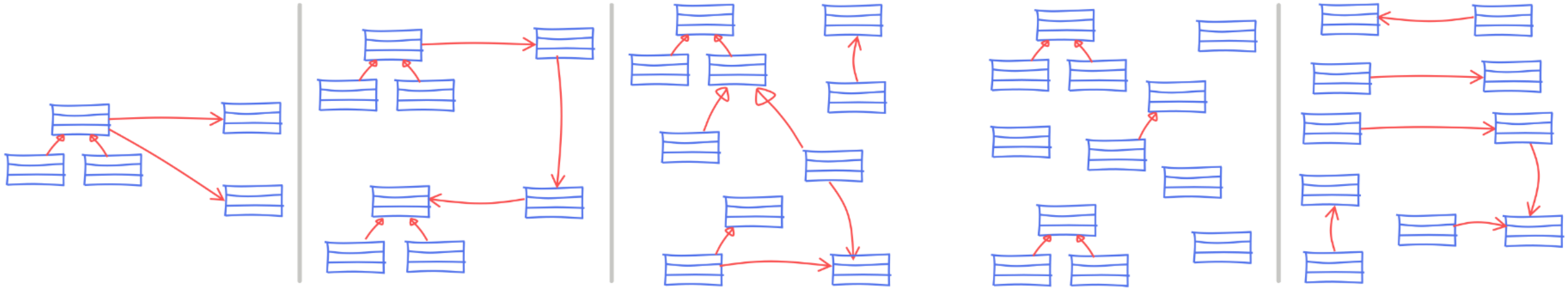

Consider reusing
foundational concepts
Fill any gaps, e.g. define
new classes and relations

Rationalise relations where possible



Fill any gaps, e.g. define
new classes and relations

Modularise the presentation of visuals



Perspective A | Perspective B | Perspective C | Class hierarchy | Relations

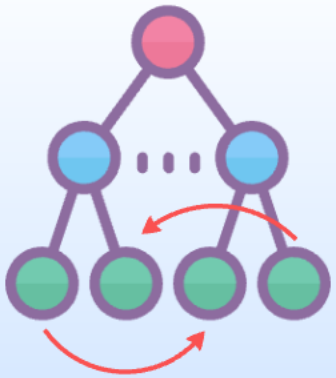
Rationalise relations where possible

05 Formalization

Formalization

Wide range of applications

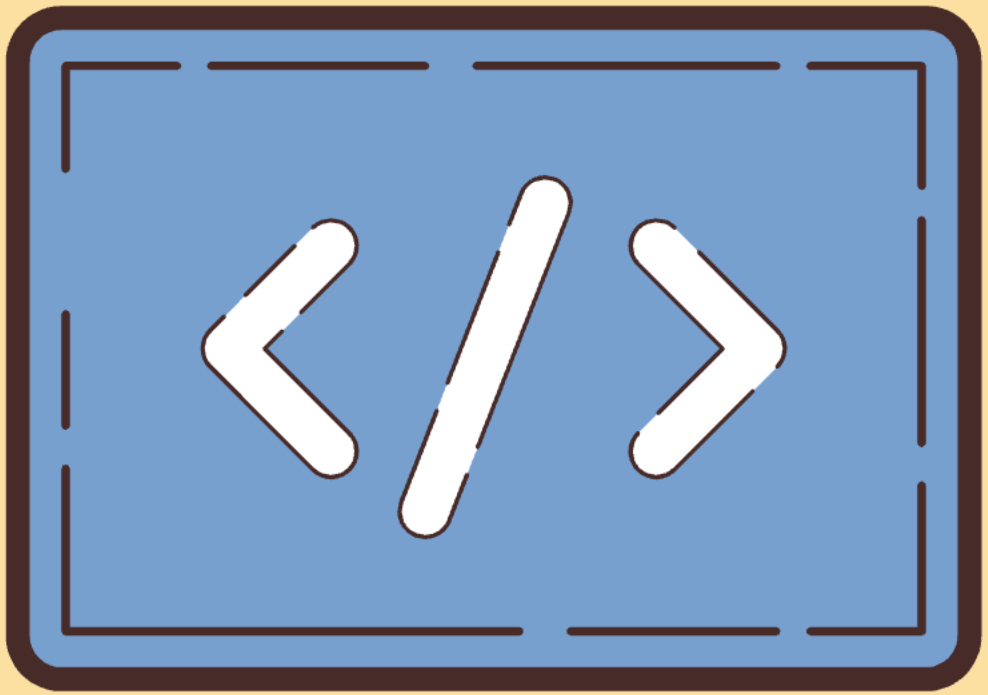
Encode knowledge models

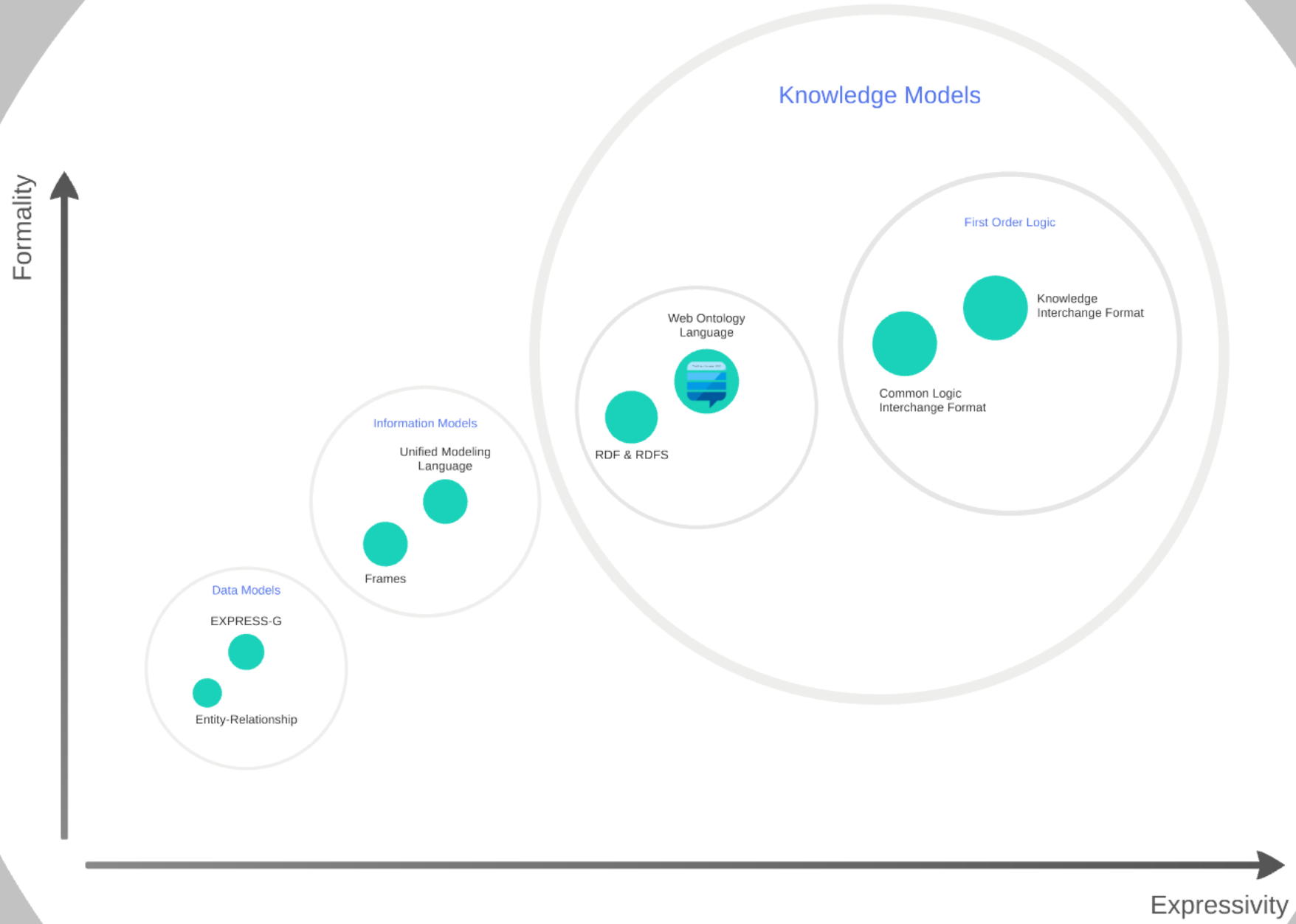


Formal knowledge representation language

Editing environment

Ontology development environment




Knowledge Models
First Order Logic
Knowledge
Interchange Format
Common Logic
Interchange Format
Web Ontology
Language
RDF & RDFS
Information Models
Unified Modeling
Language
Frames
Data Models
EXPRESS-G
Entity-Relationship
Formality
Expressivity

Protégé

Information Models

Unified Modeling Language

Frames

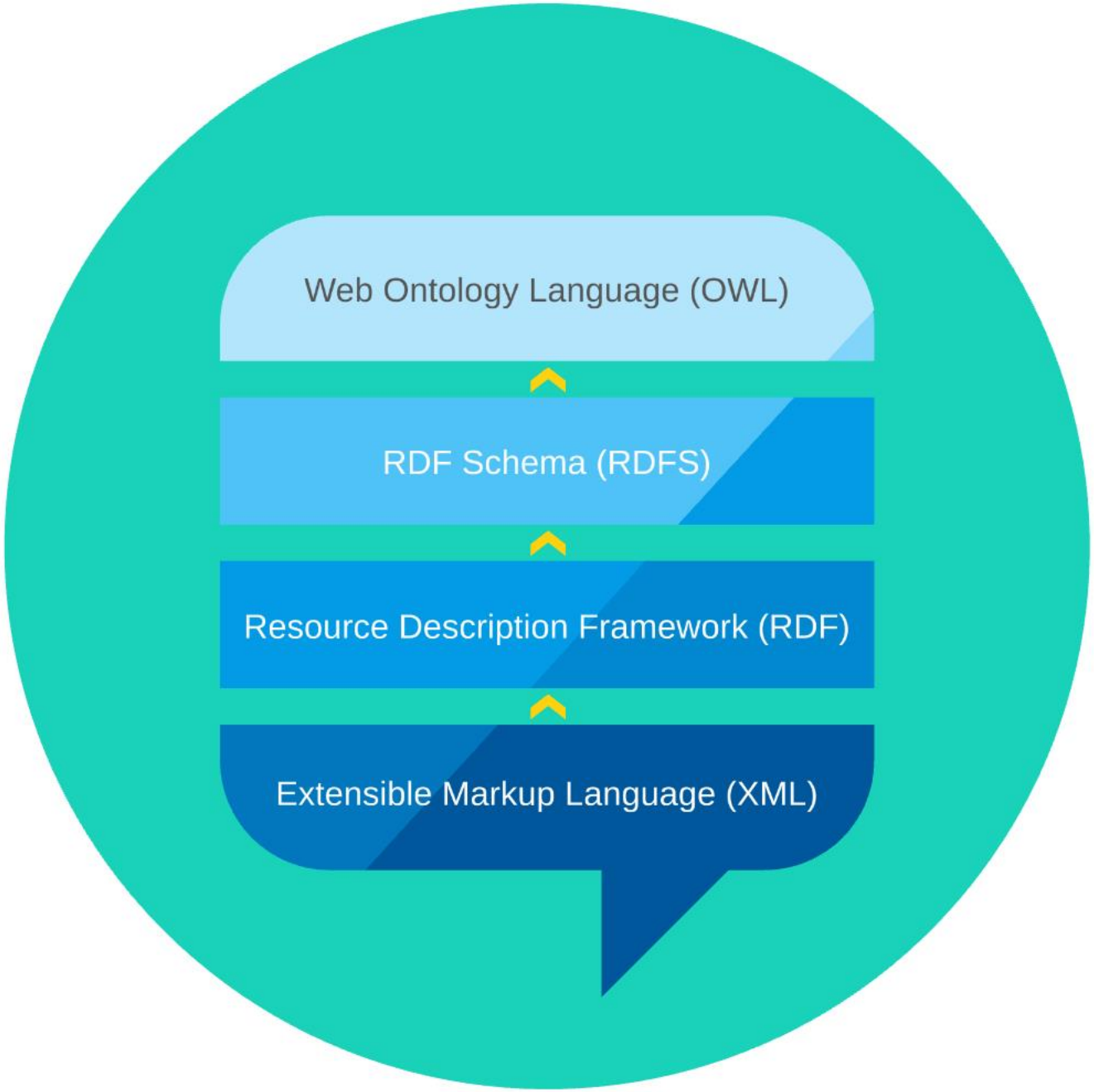
Web Ontology Language

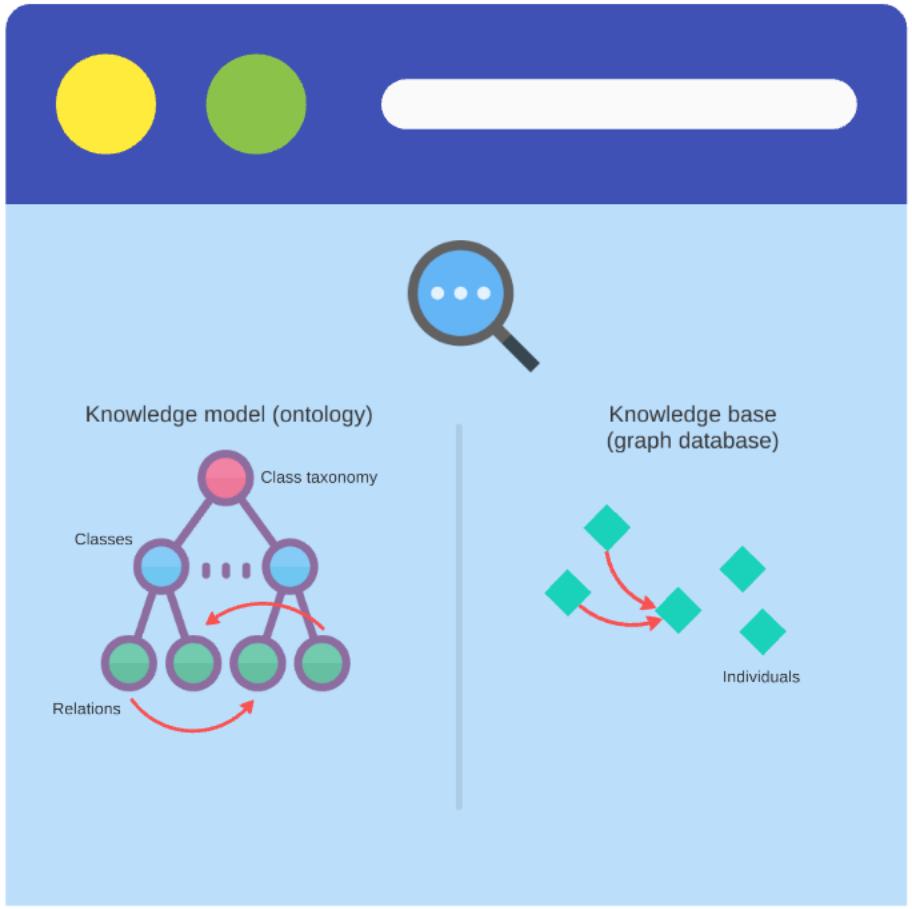
RDF & RDFS

Description Logic

First Order Logic

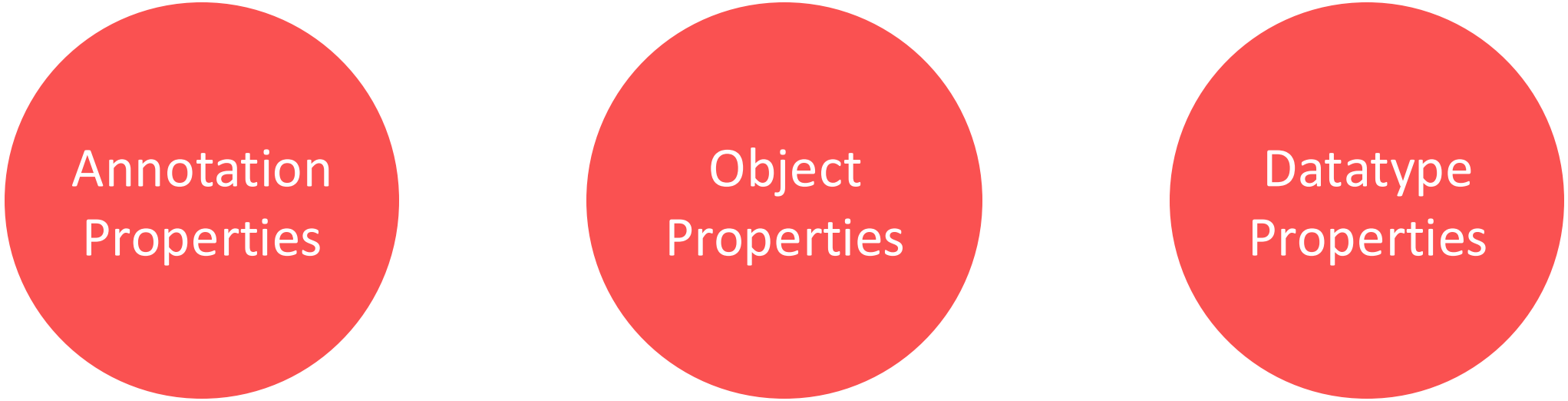
Common Logic Interchange Format


Web Ontology Language (OWL)
RDF Schema (RDFS)
Resource Description Framework (RDF)
Extensible Markup Language (XML)

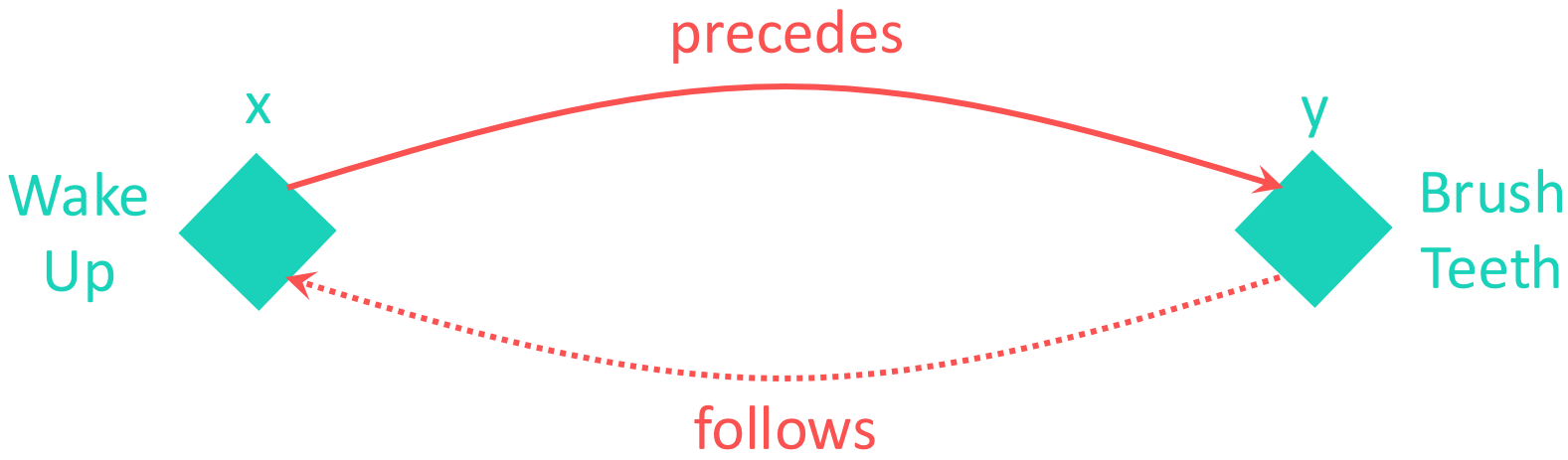

Knowledge model (ontology)
Class taxonomy
Classes
Relations
Knowledge base
(graph database)
Individuals

SPARQL

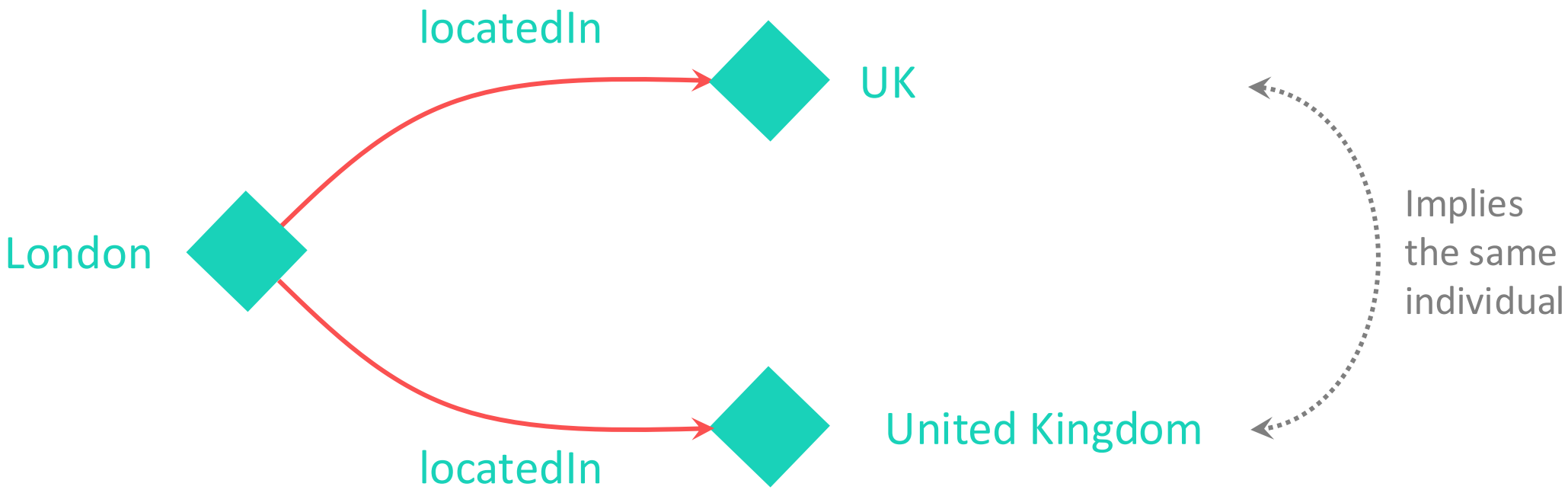
OWL Properties



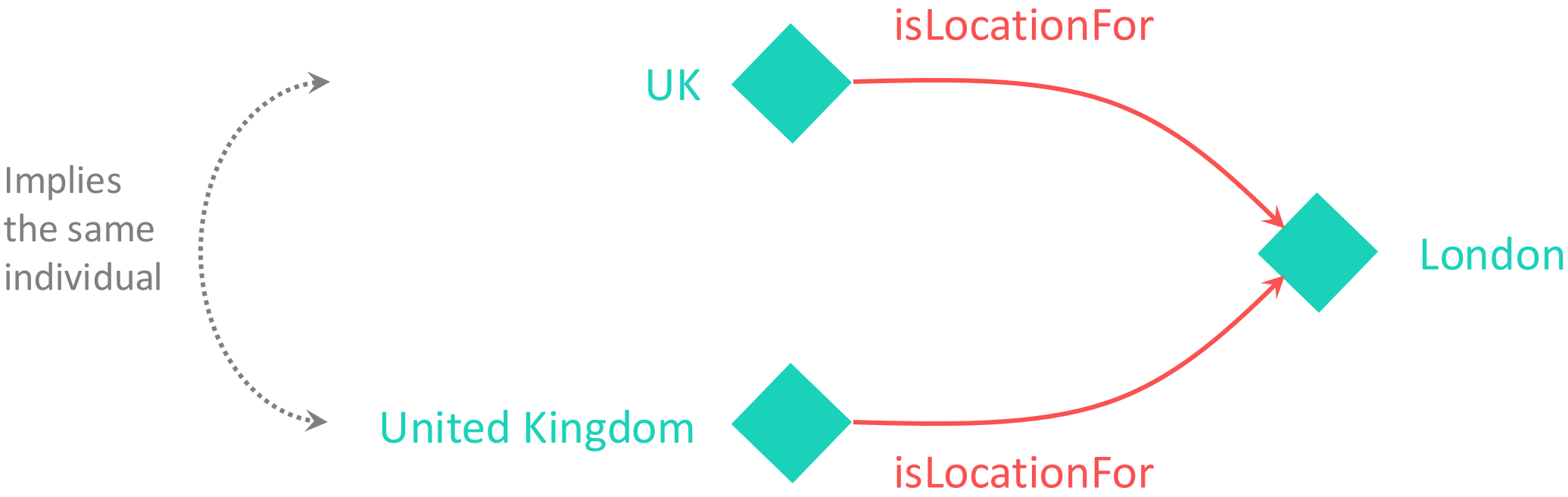
Inverse Properties



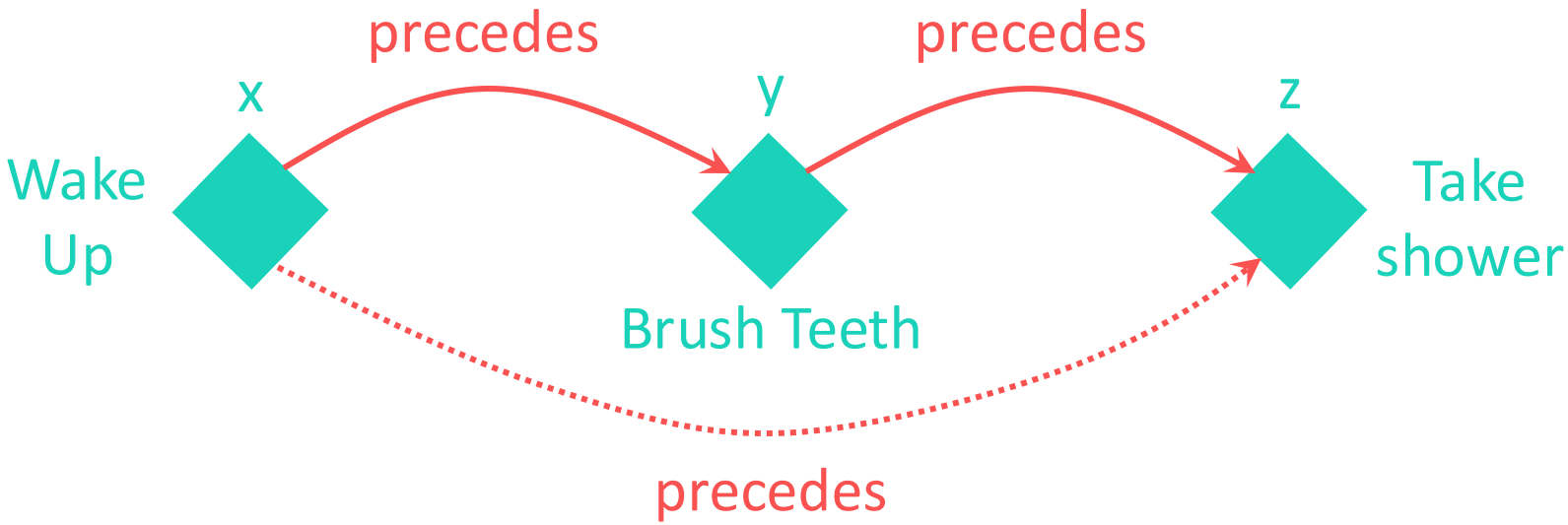
Functional Properties



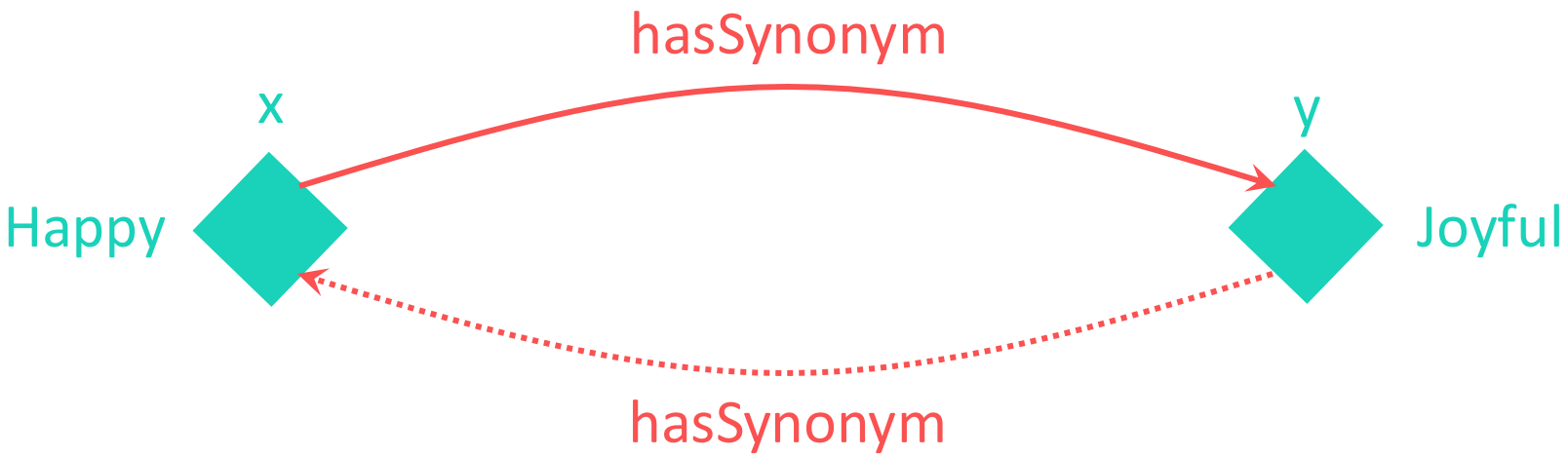
Inverse Functional Properties



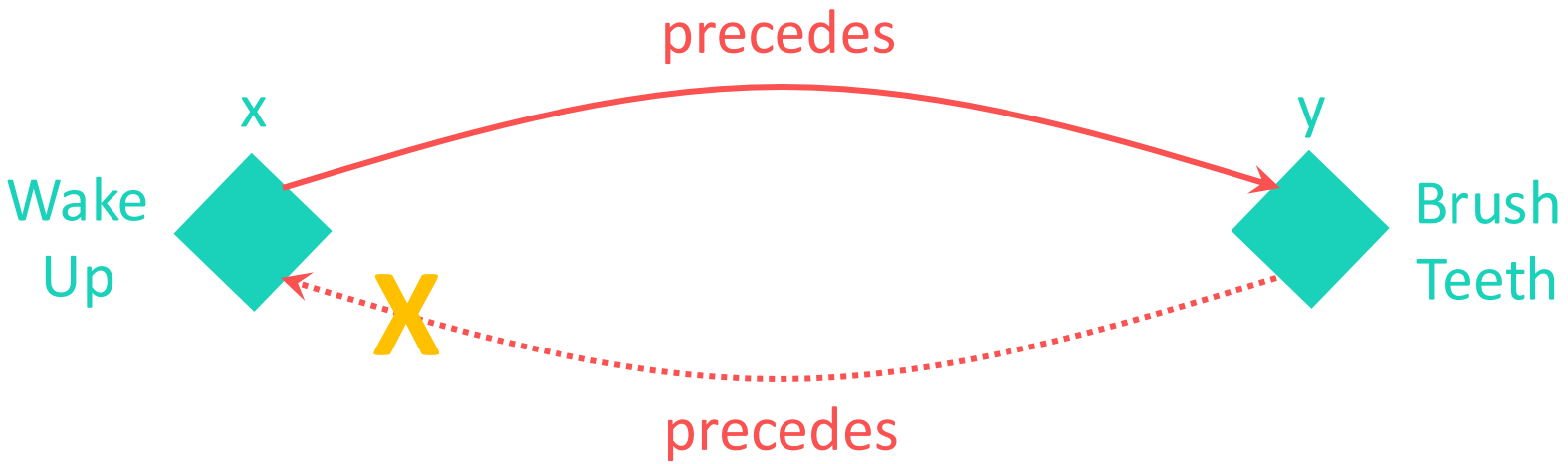
Transitive Properties



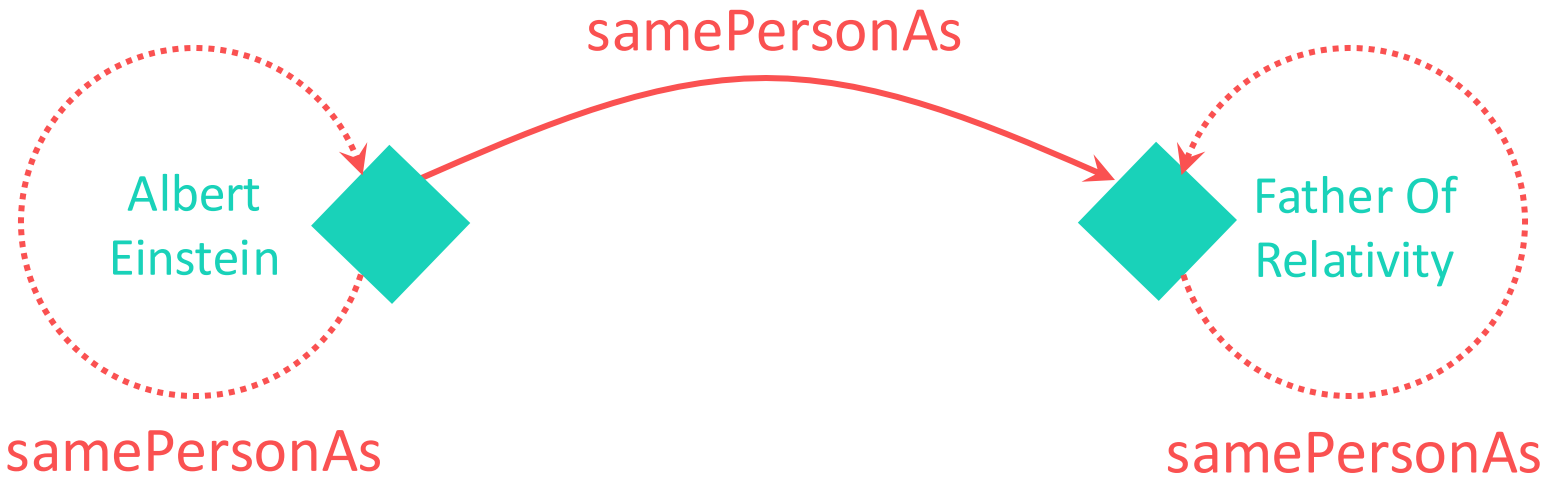
Symmetric Properties



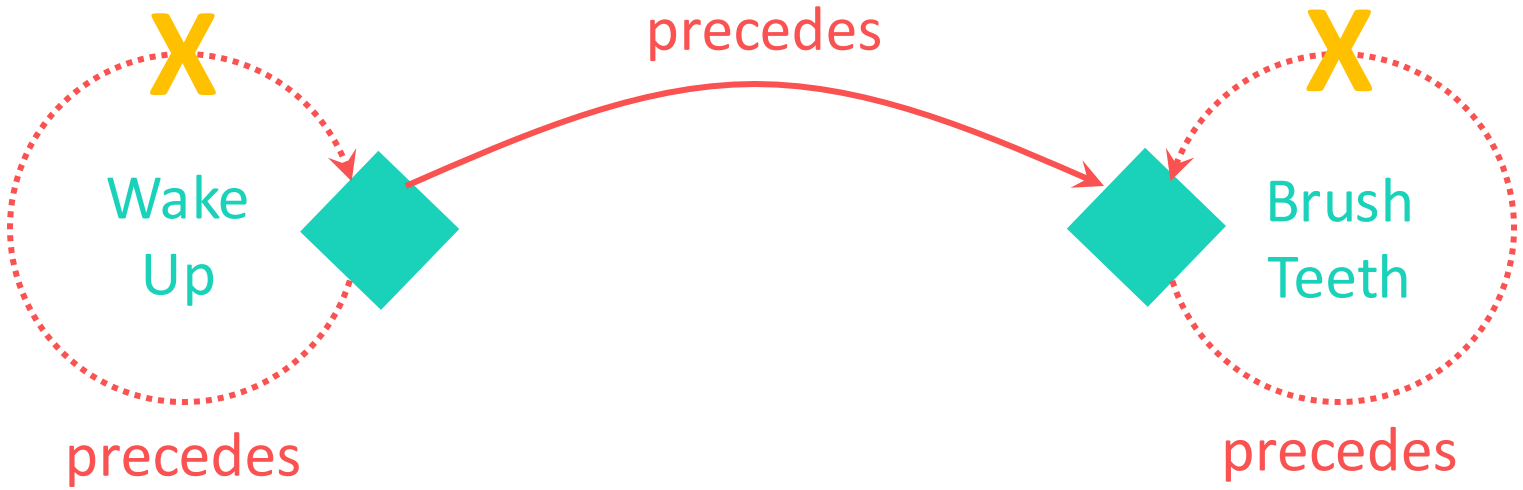
Asymmetric Properties



Reflexive Properties



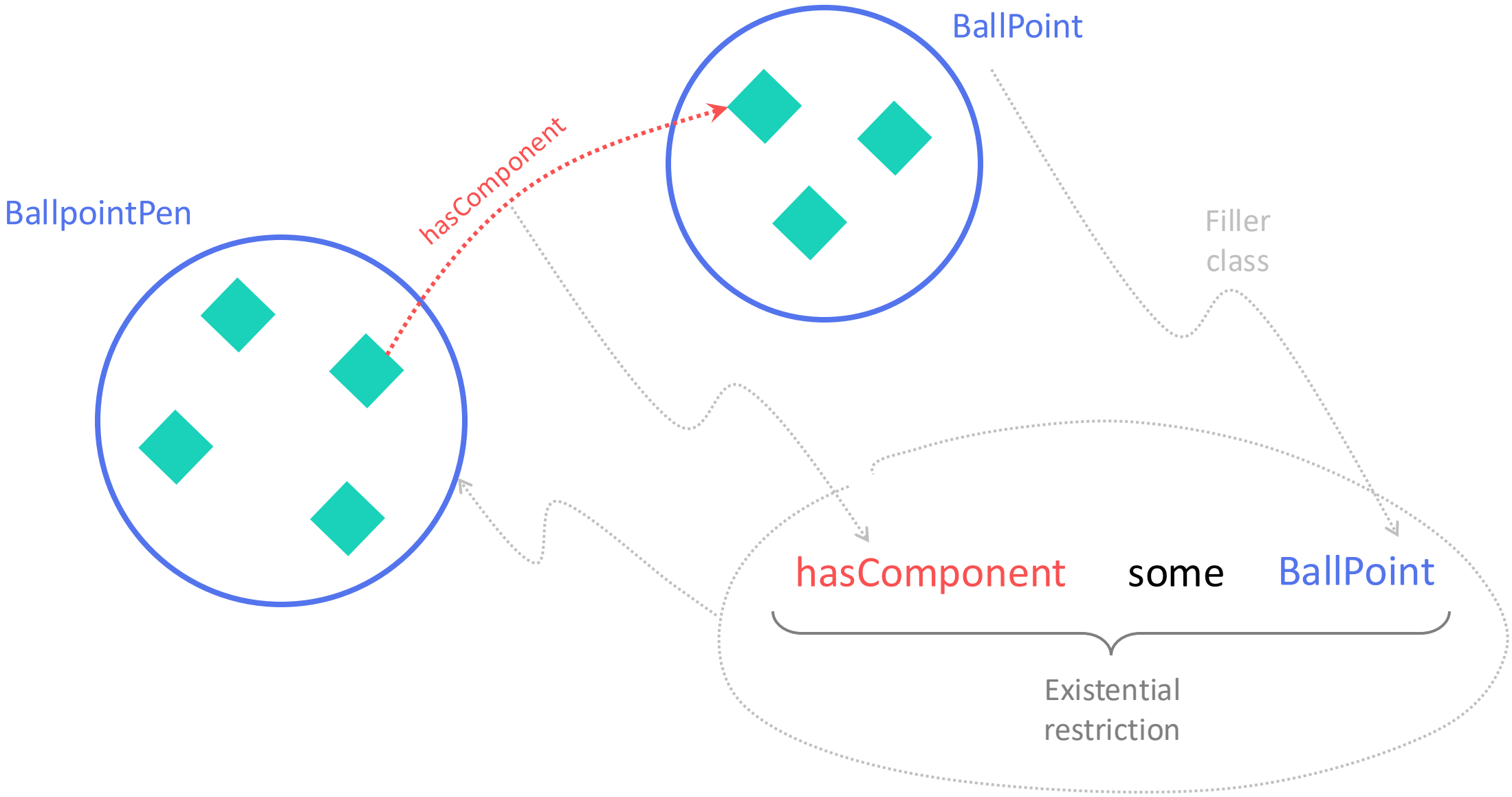
Irreflexive Properties

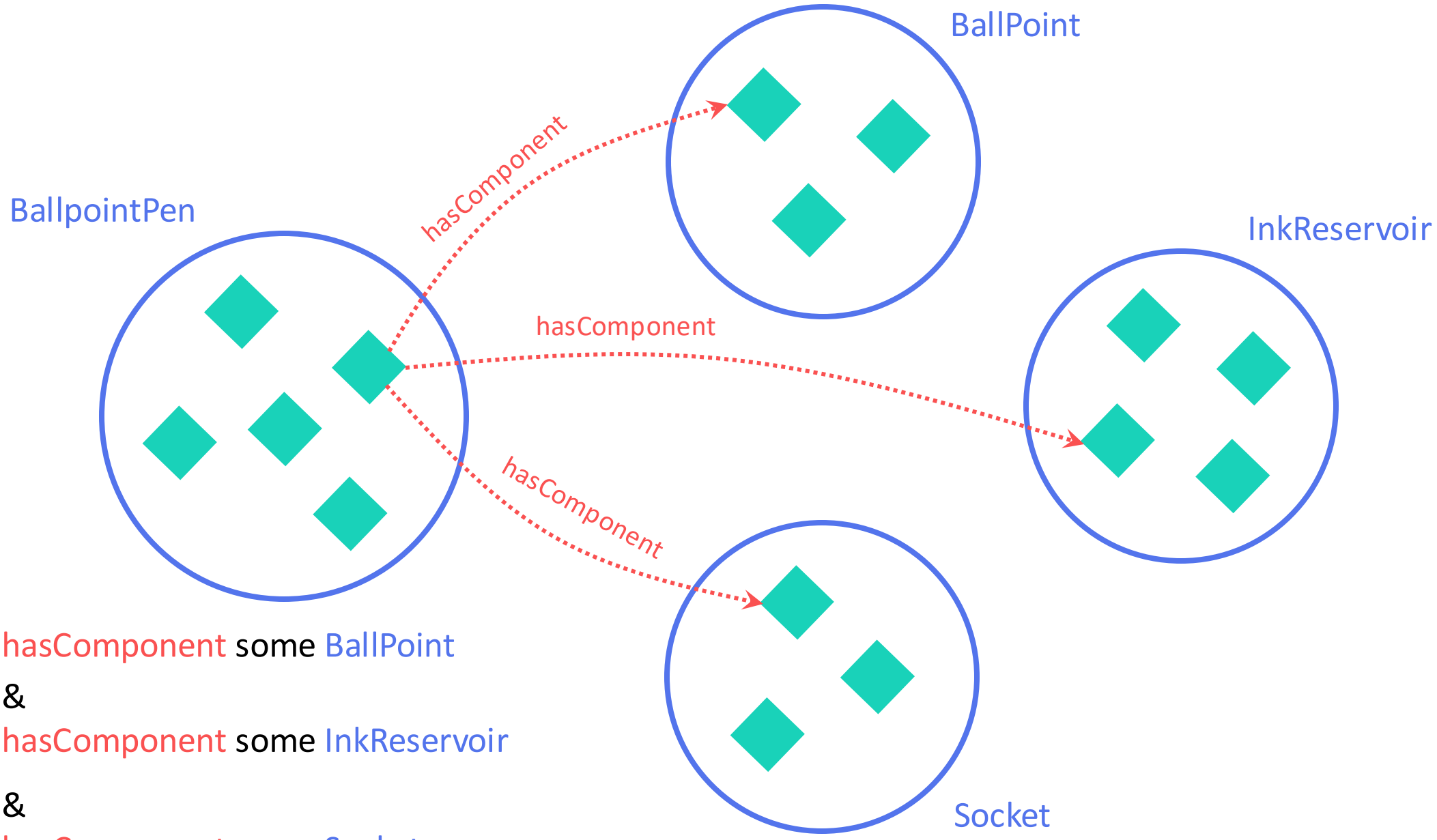


OWL Restrictions

- Existential restriction
- Universal restriction

Existential Restrictions





hasComponent some BallPoint

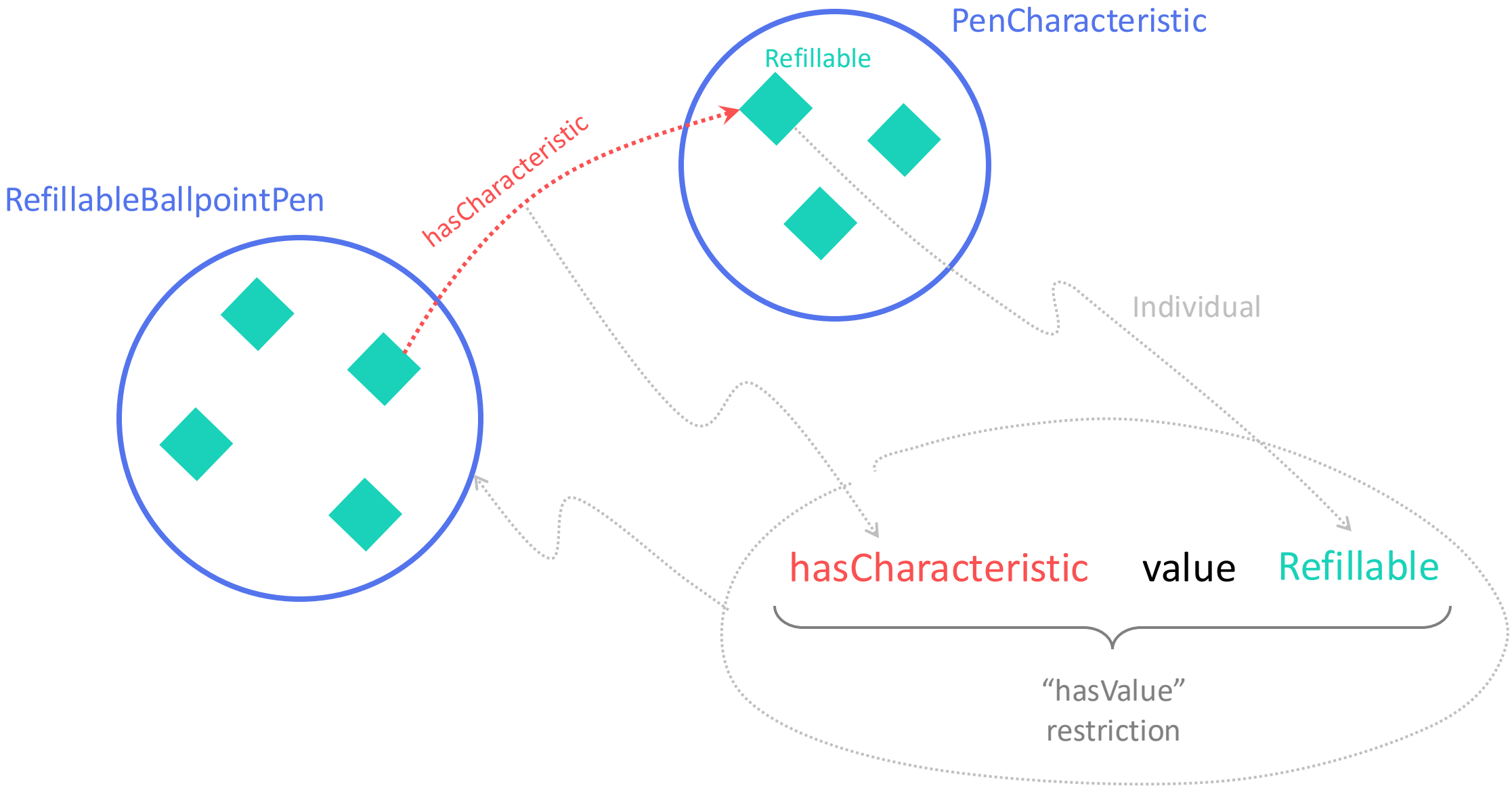
&

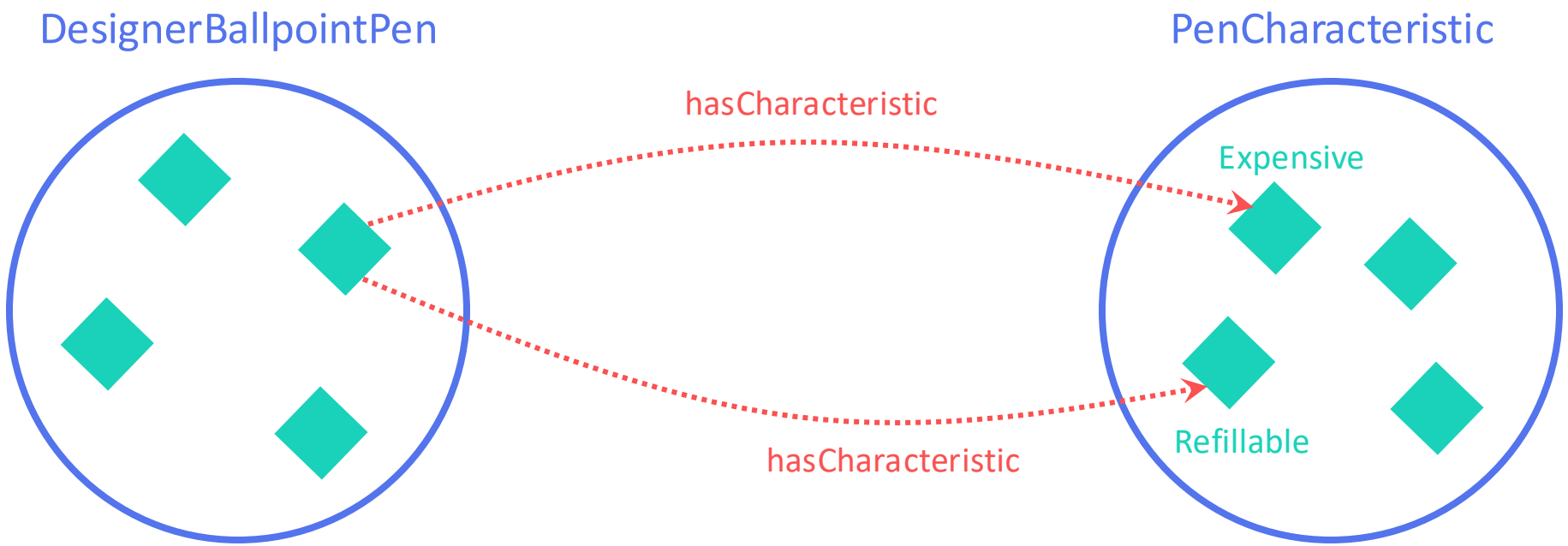
hasComponent some InkReservoir

&

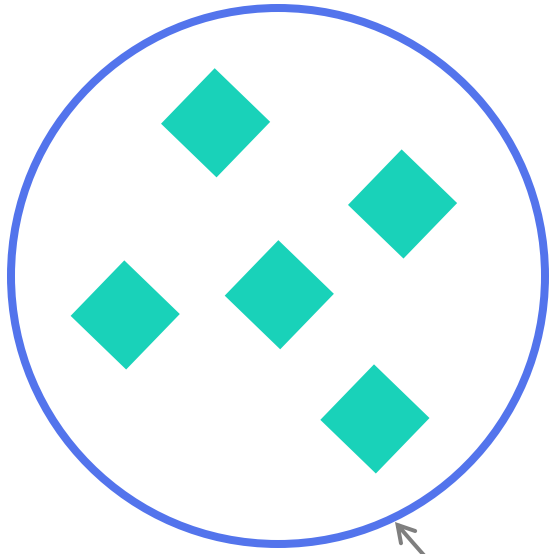
hasComponent some Socket

hasValue Restrictions




DesignerBallpointPen
PenCharacteristic
hasCharacteristic
Expensive
hasCharacteristic
Refillable

BallpointPen



Primitive Class

Implies

Necessary Conditions

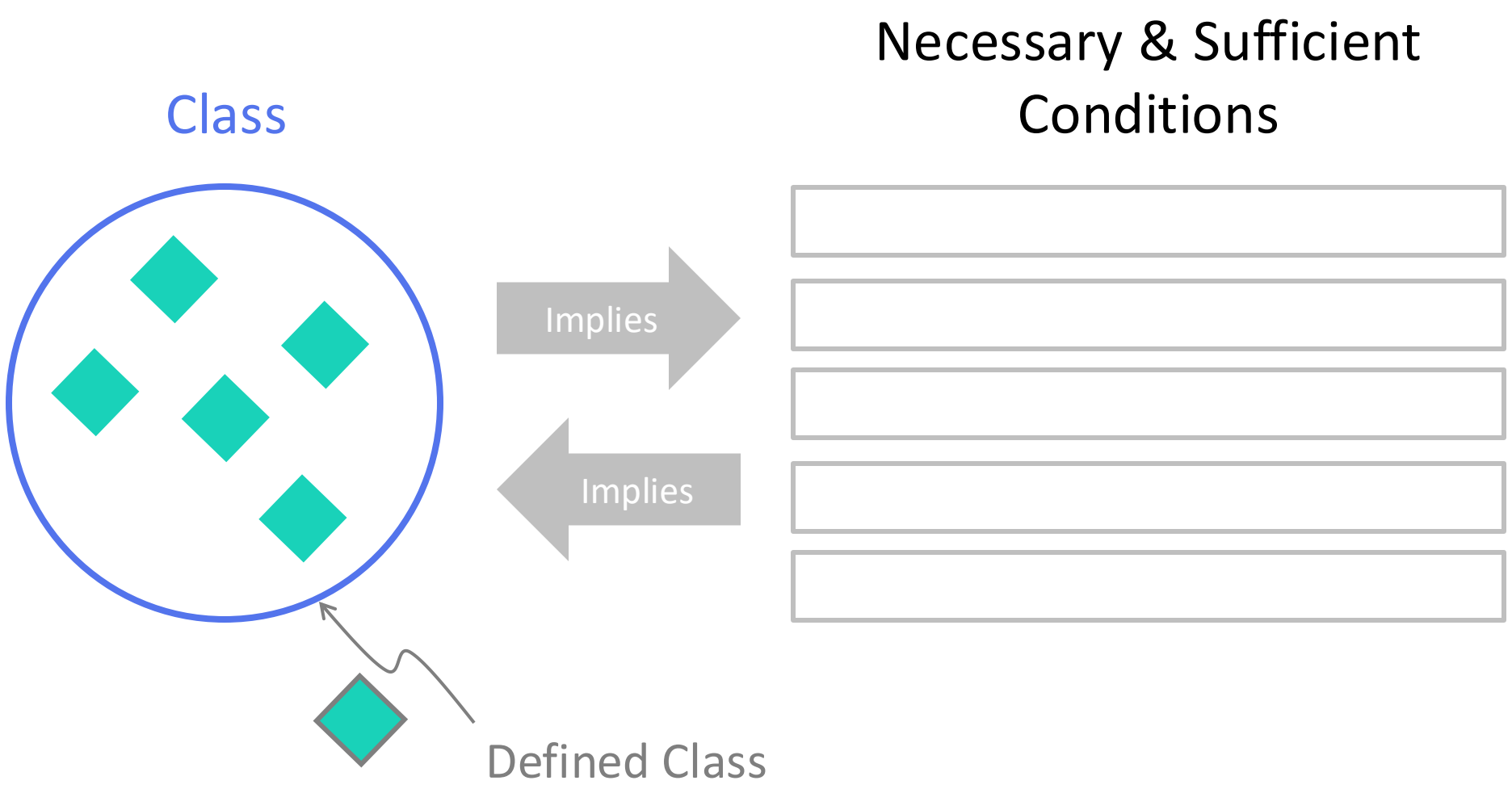
hasComponent some BallPoint

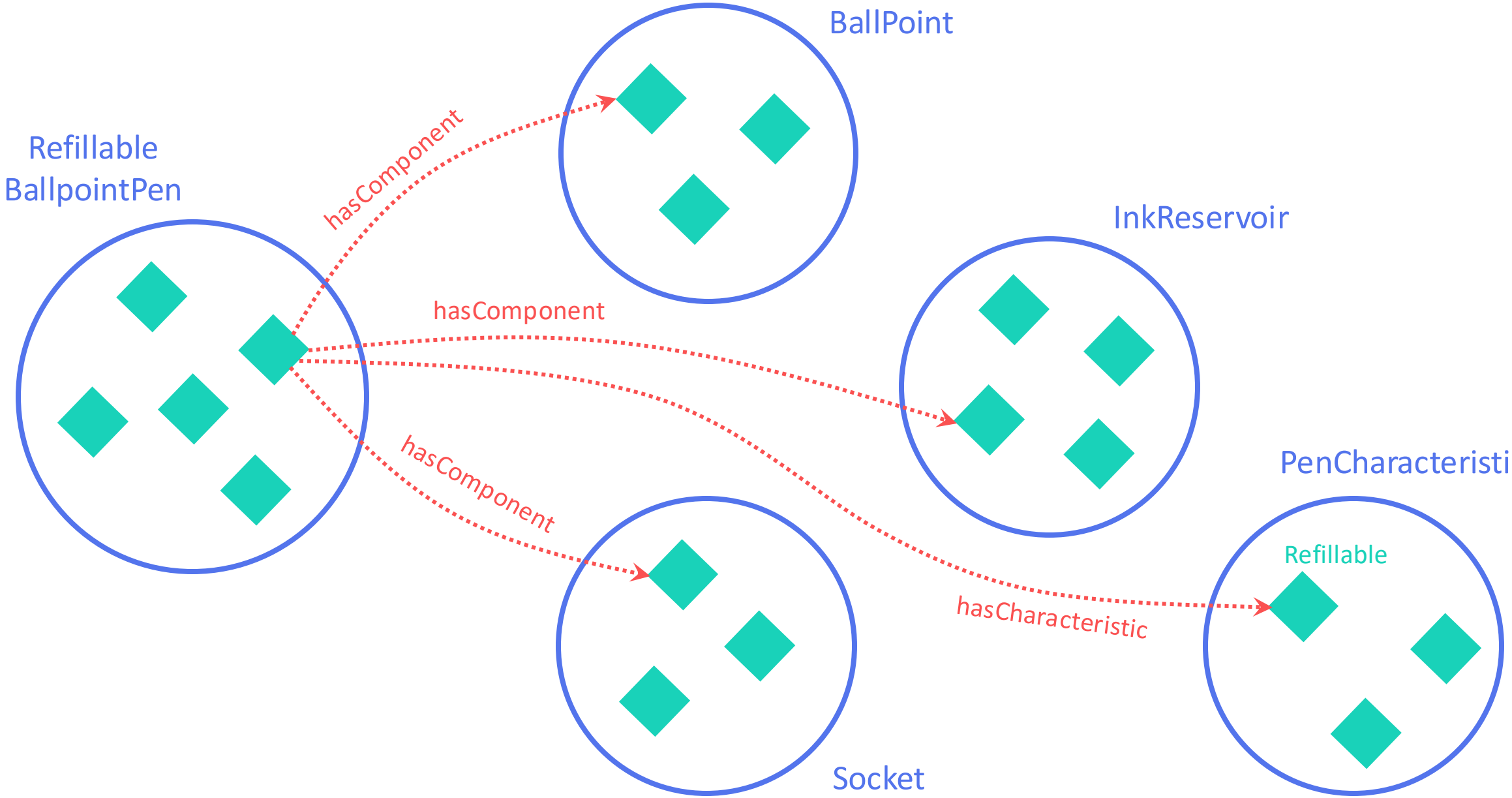
&

hasComponent some InkReservoir

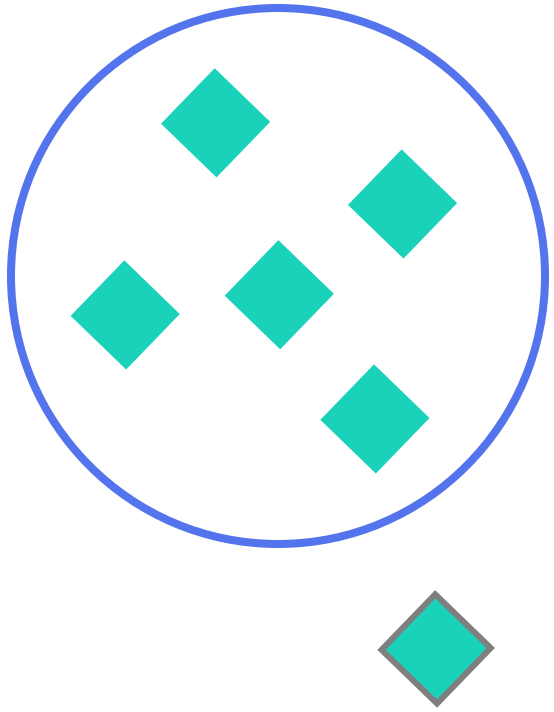
&

hasComponent some Socket


Class
Necessary & Sufficient Conditions
Implies
Implies
Defined Class


BallPoint
Refillable
BallpointPen
hasComponent
InkReservoir
hasComponent
hasComponent
PenCharacteristic
Refillable
hasCharacteristic
Socket

RefillableBallpointPen



Implies

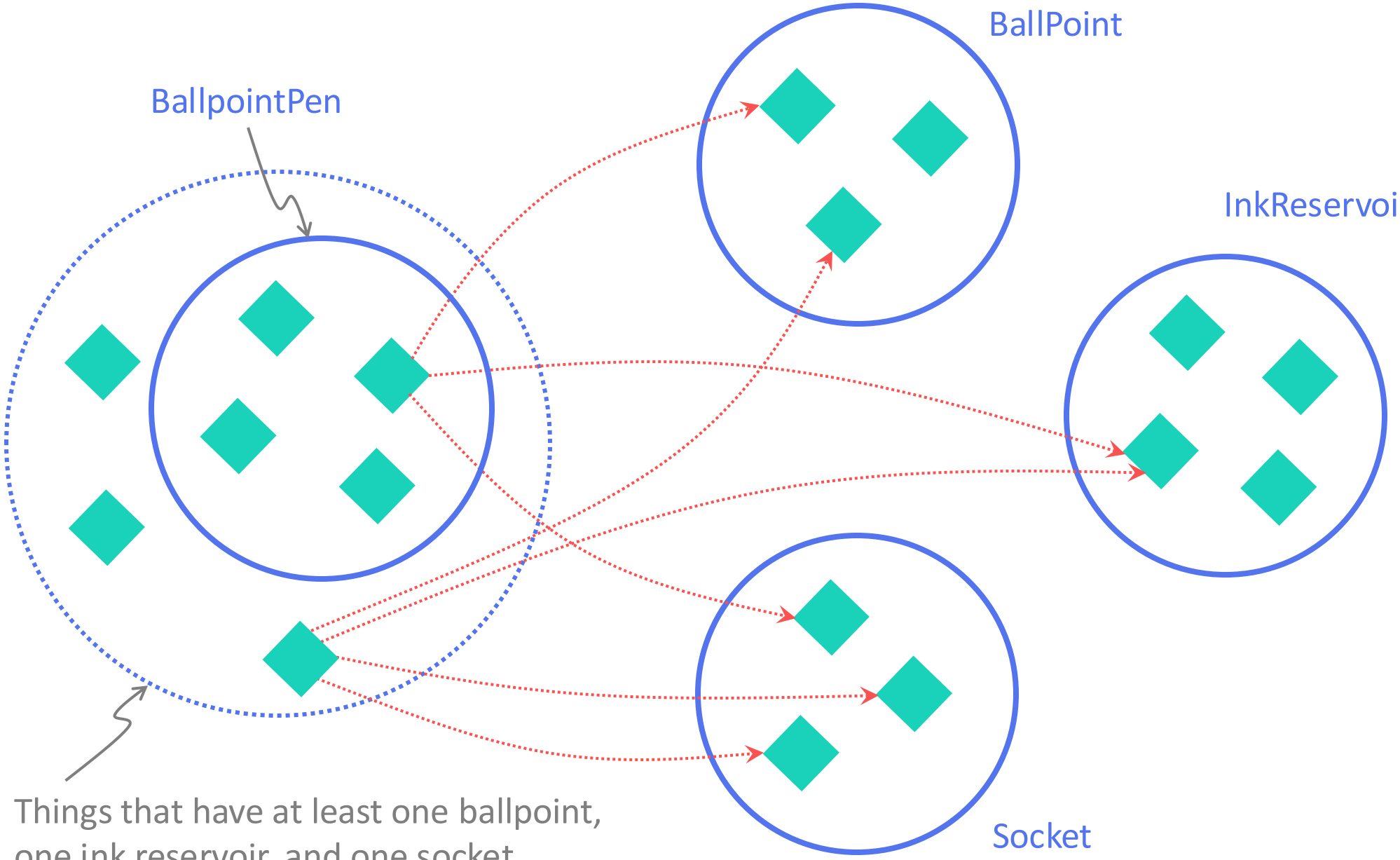
Implies

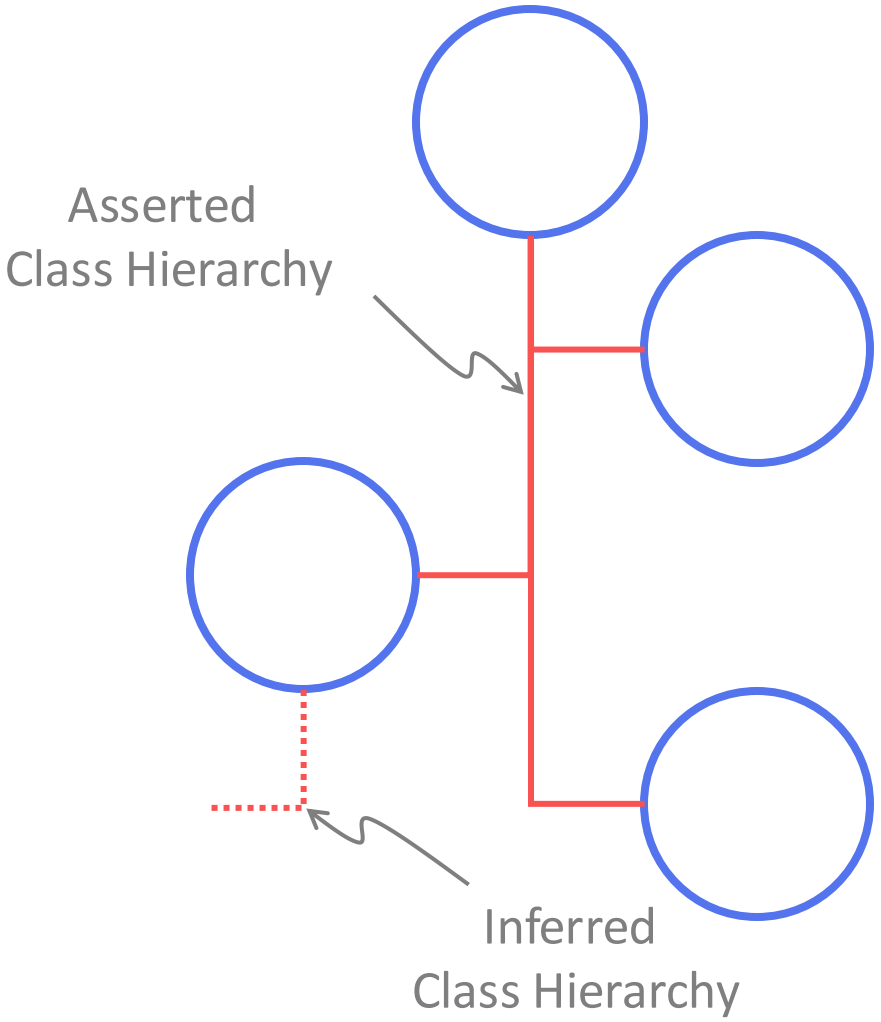
Necessary & Sufficient Conditions

is-a BallpointPen

&

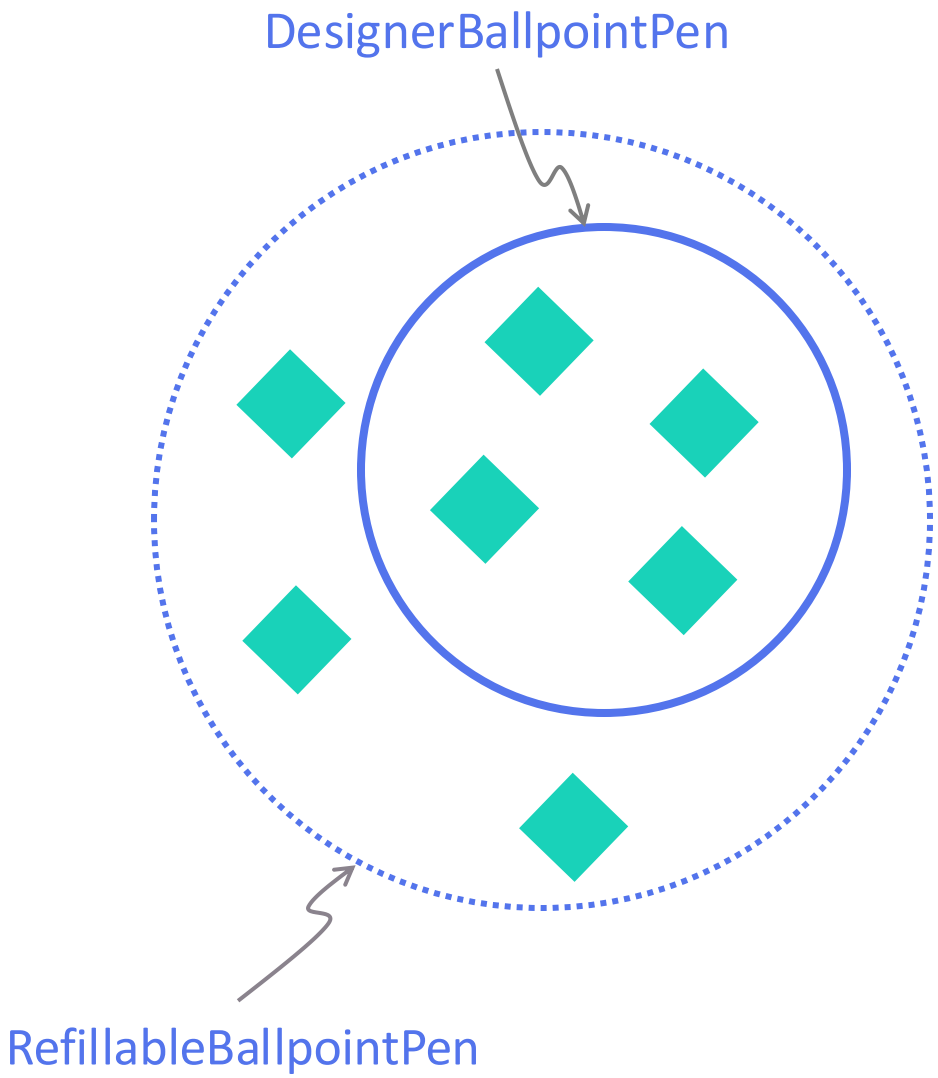
hasCharacteristic value Refillable


BallPoint
BallpointPen
InkReservoir
Things that have at least one ballpoint,
one ink reservoir, and one socket
Socket

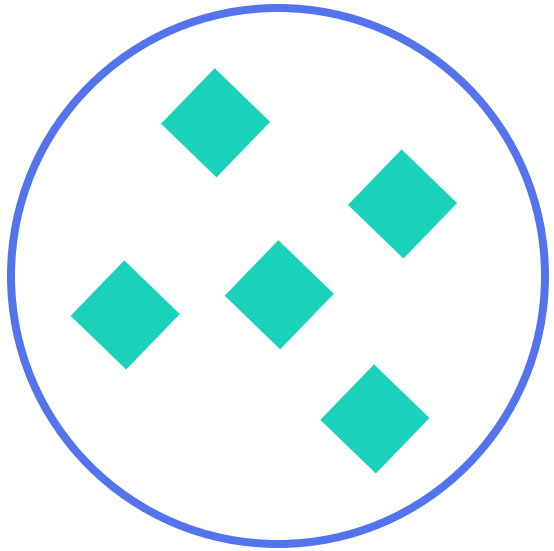


- Automated services to make inferences
- Automated classification
- Useful when building large ontologies
- Verifies correctness and consistency

Reasoning in OWL


DesignerBallpointPen
RefillableBallpointPen

RefillableBallpointPen



Implies

Implies

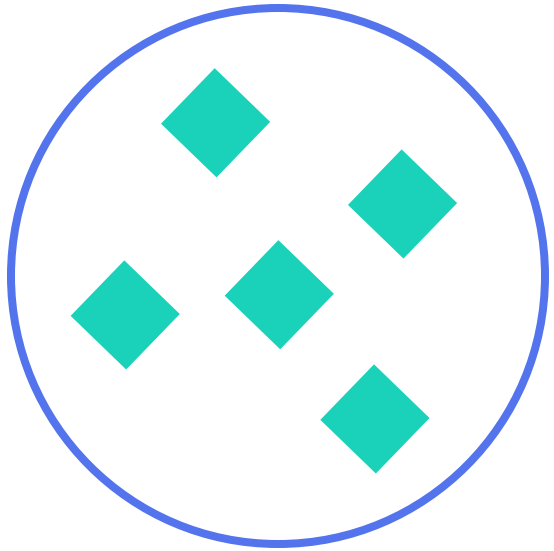
Necessary & Sufficient Conditions

is-a BallpointPen

&

hasCharacteristic value Refillable

DesignerBallpointPen



Implies

Implies

Necessary & Sufficient Conditions

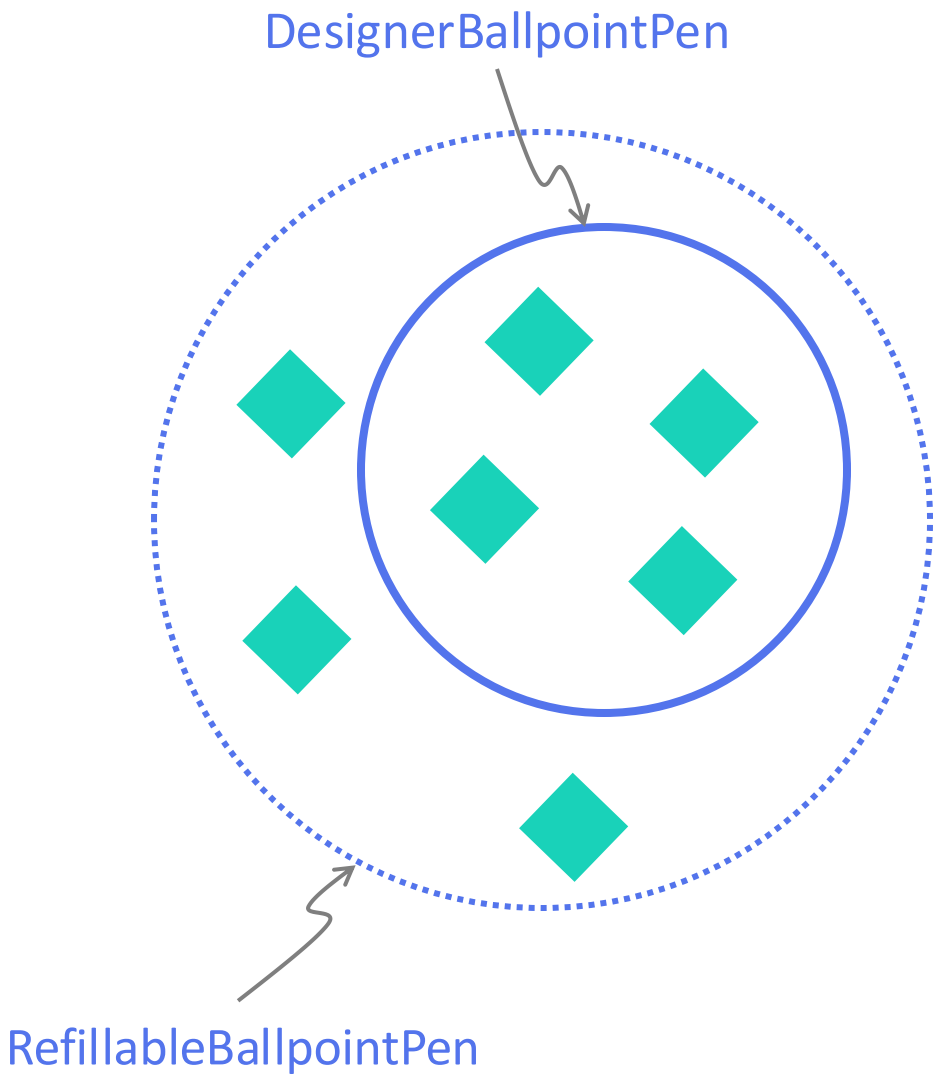
is-a BallpointPen

&

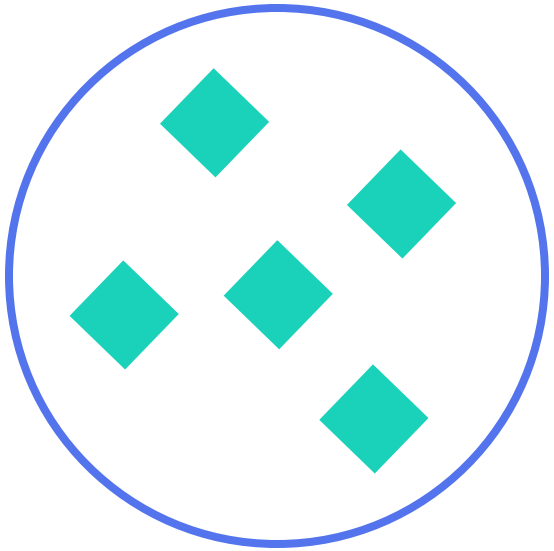
hasCharacteristic value Refillable

&

hasCharacteristic value Expensive


DesignerBallpointPen
RefillableBallpointPen

FisherSpacePen



Implies

Necessary Conditions

- is-a BallpointPen
- hasCharacteristic value UnderWaterUsage
- hasCharacteristic value UpsideDownUsage
- hasCharacteristic value ZeroGravityUsage
- hasMake value Fisher

OWL Restrictions

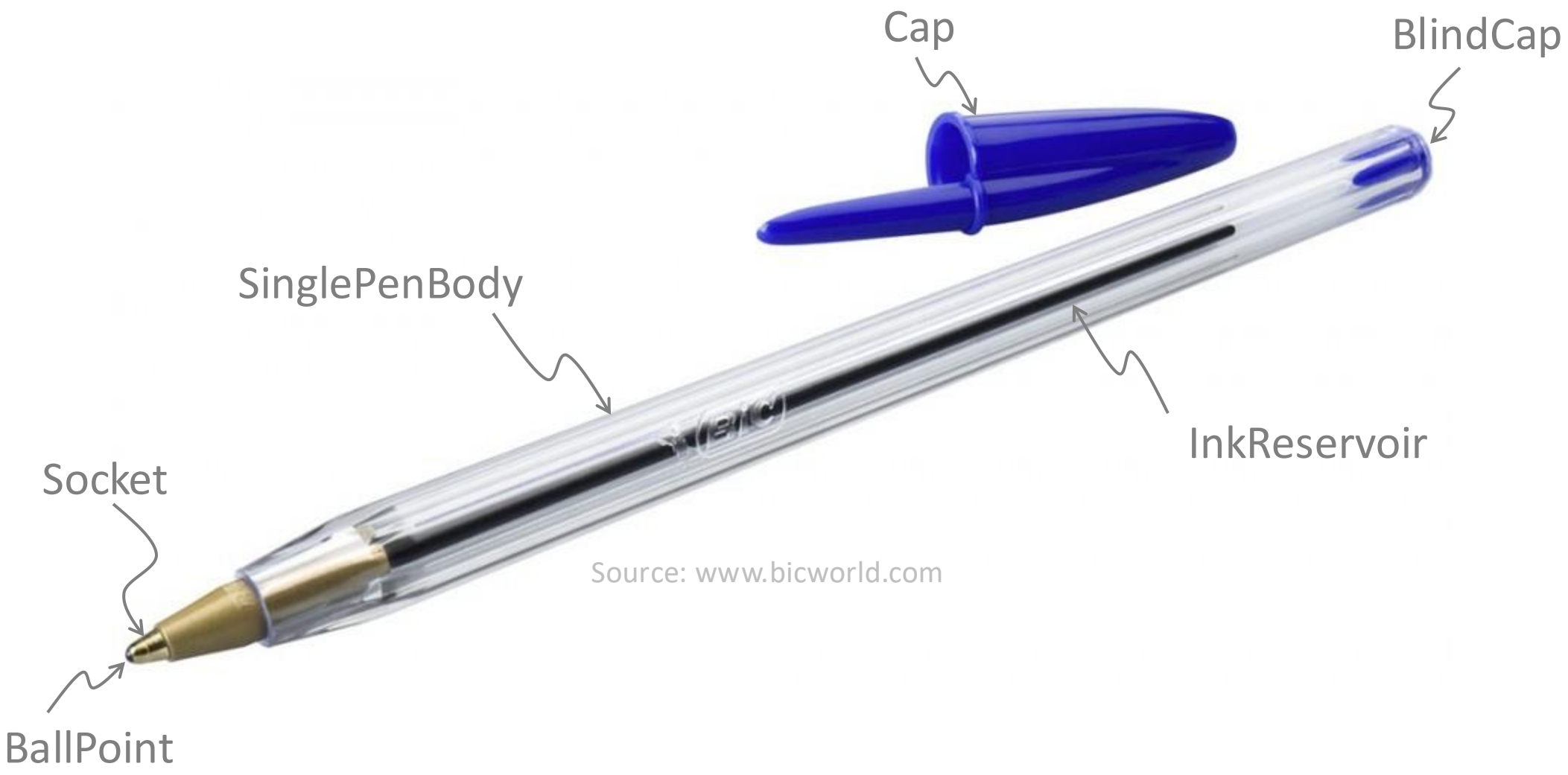
Quantifier Restriction

"hasValue" Restriction

Cardinality Restriction

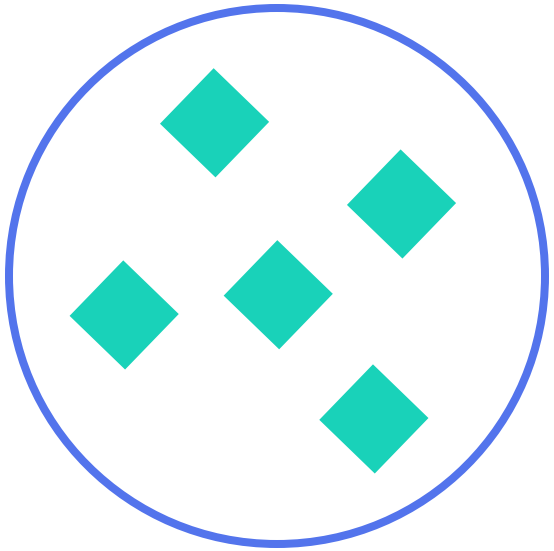
- Existential restriction
- Universal restriction

BicCristal



Source: www.bicworld.com

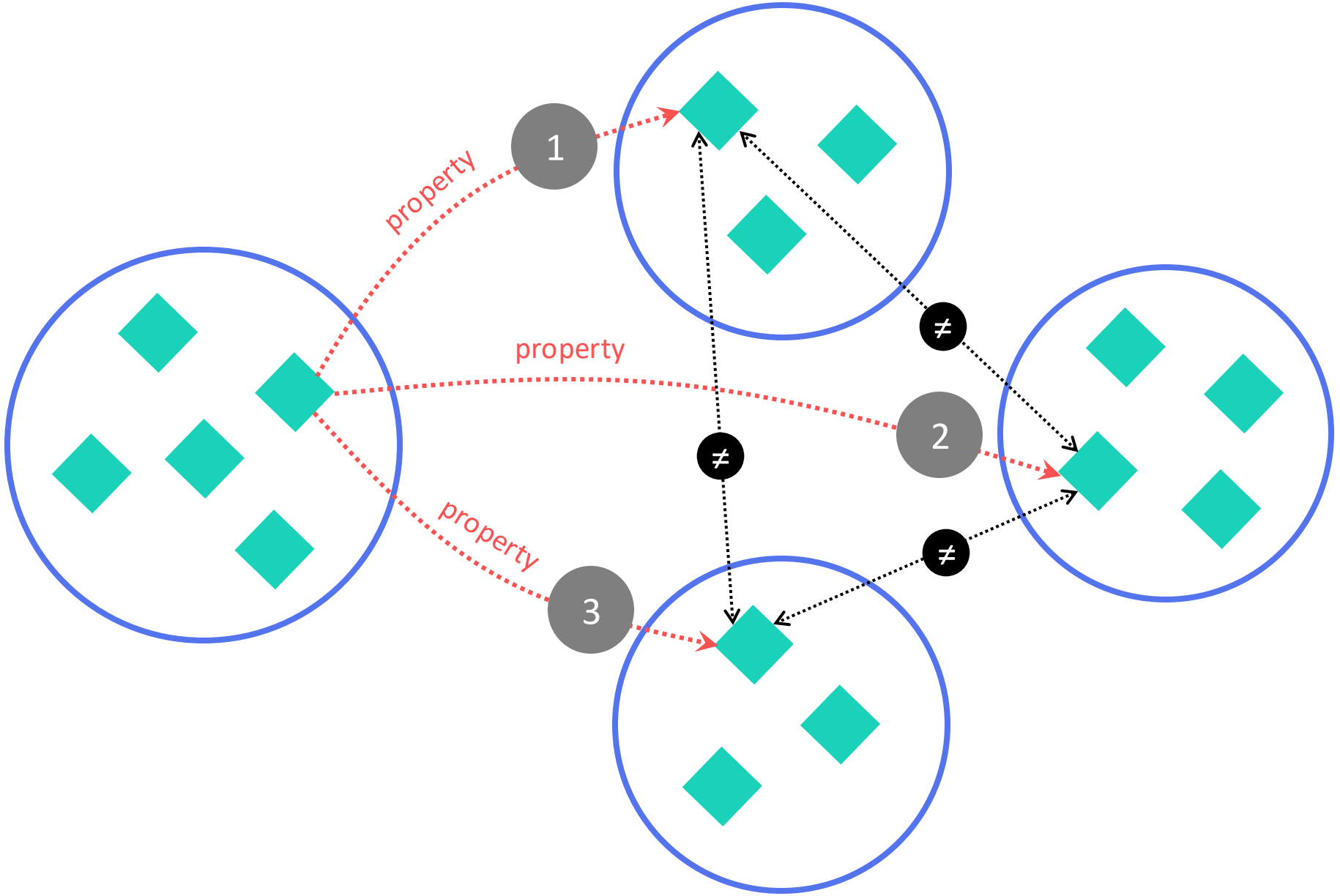
BicCristal



Implies

Necessary Conditions

- is-a BallpointPen
- hasCharacteristic value Disposable
- hasMake value Bic
- hasComponent only (BallPoint or BlindCap or Cap or InkReservoir or SinglePenBody or Socket)

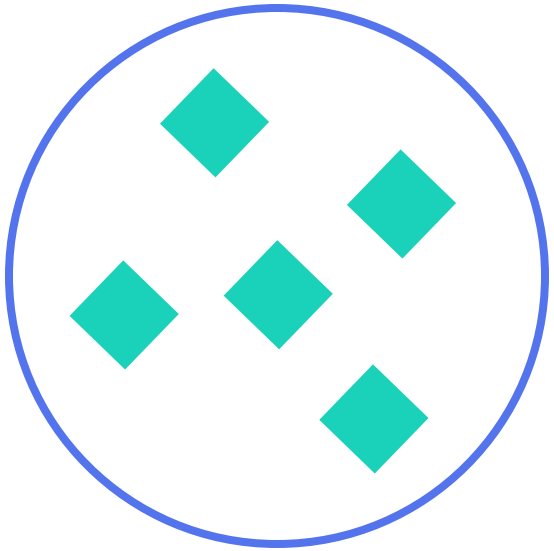

1
property
property
2
property
3
≠
≠
≠

MultiColourBallpointPen

Source: www.bicworld.com

Multicolour

MultiColourBallpointPen



Implies

Necessary Conditions

- is-a BallpointPen
- hasCharacteristic value Multicolour

MultiColourBallpointPen

Source: www.bicworld.com

Minimum of 2 ballpoints,
2 ink reservoirs, 2 sockets

Multicolour

OWL Restrictions

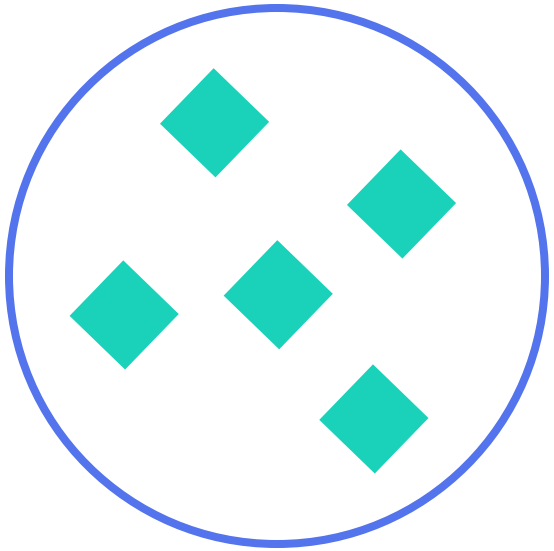
Quantifier Restriction

"hasValue" Restriction

Cardinality Restriction

- Existential restriction
- Universal restriction

MultiColourBallpointPen

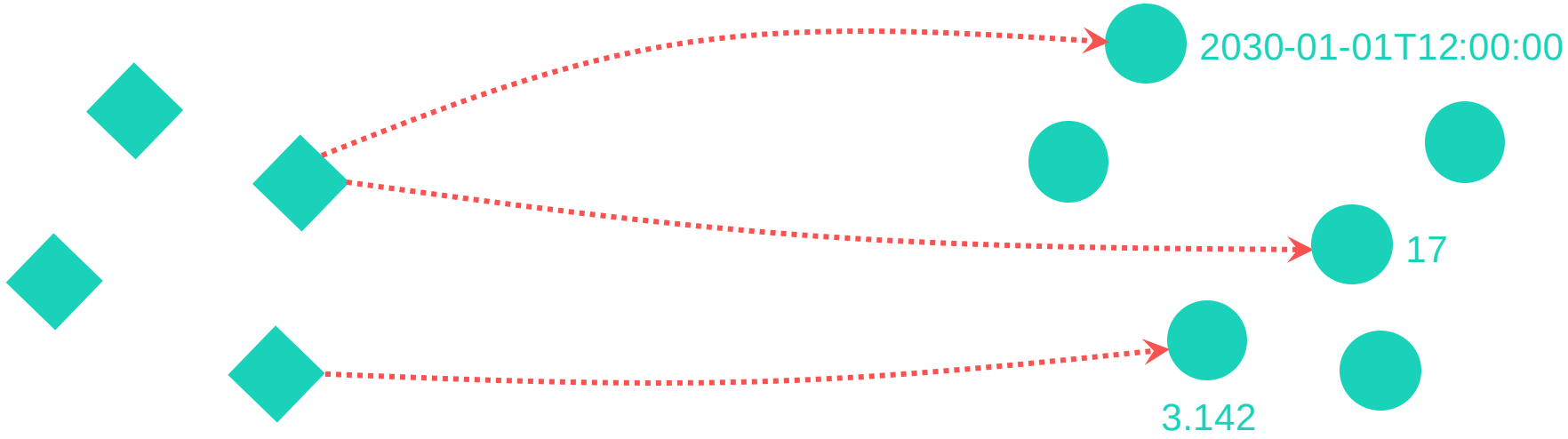


Implies

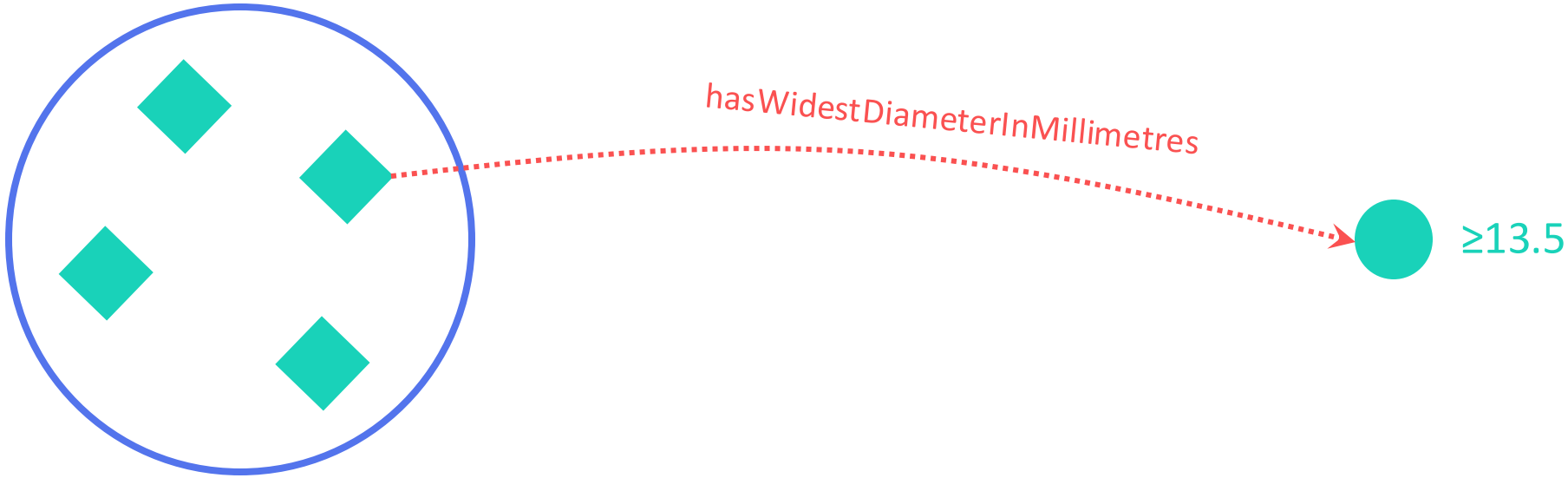
Necessary Conditions

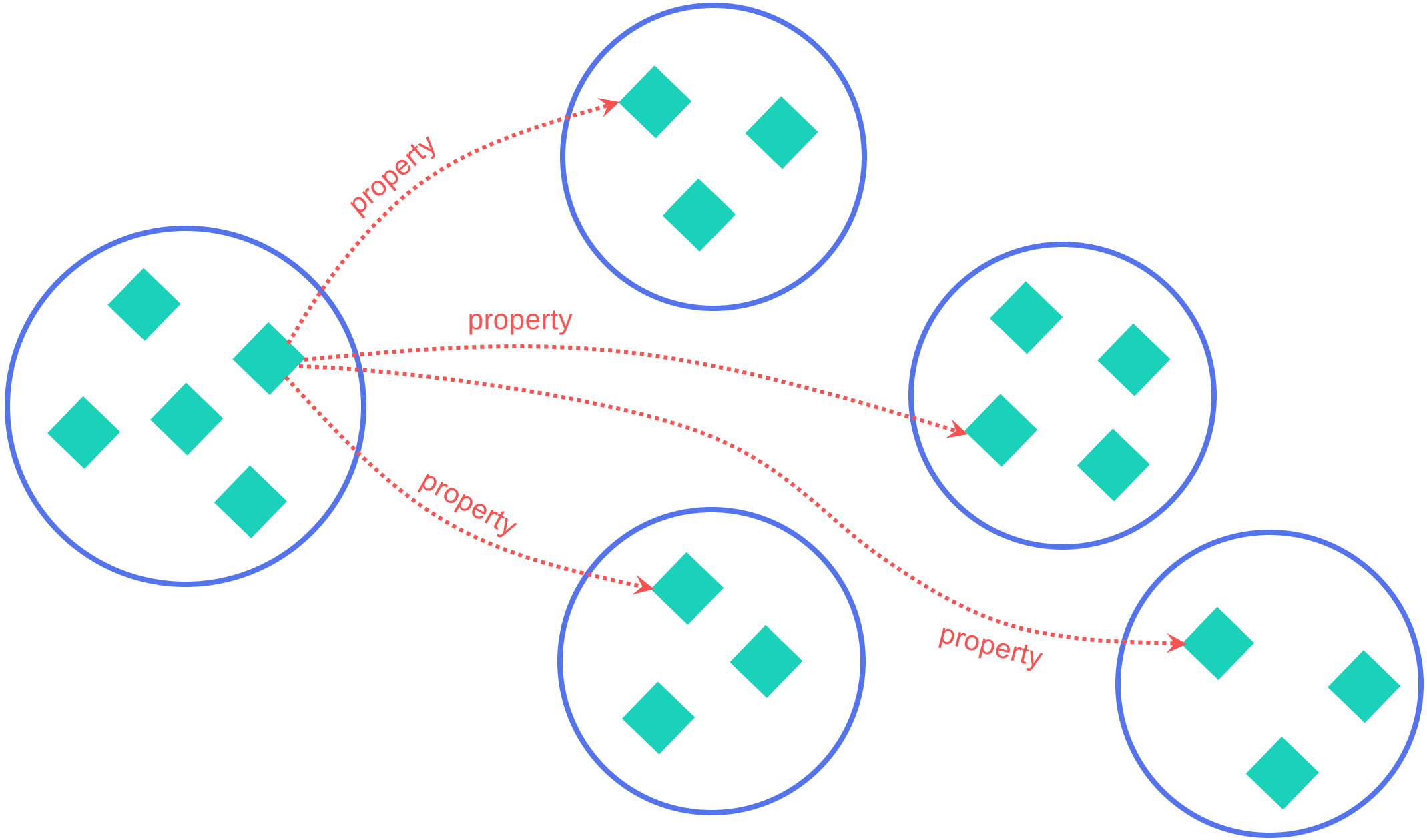
- is-a BallpointPen
- hasCharacteristic value Multicolour
- hasComponent min 2 BallPoint
- hasComponent min 2 InkReservoir
- hasComponent min 2 Socket

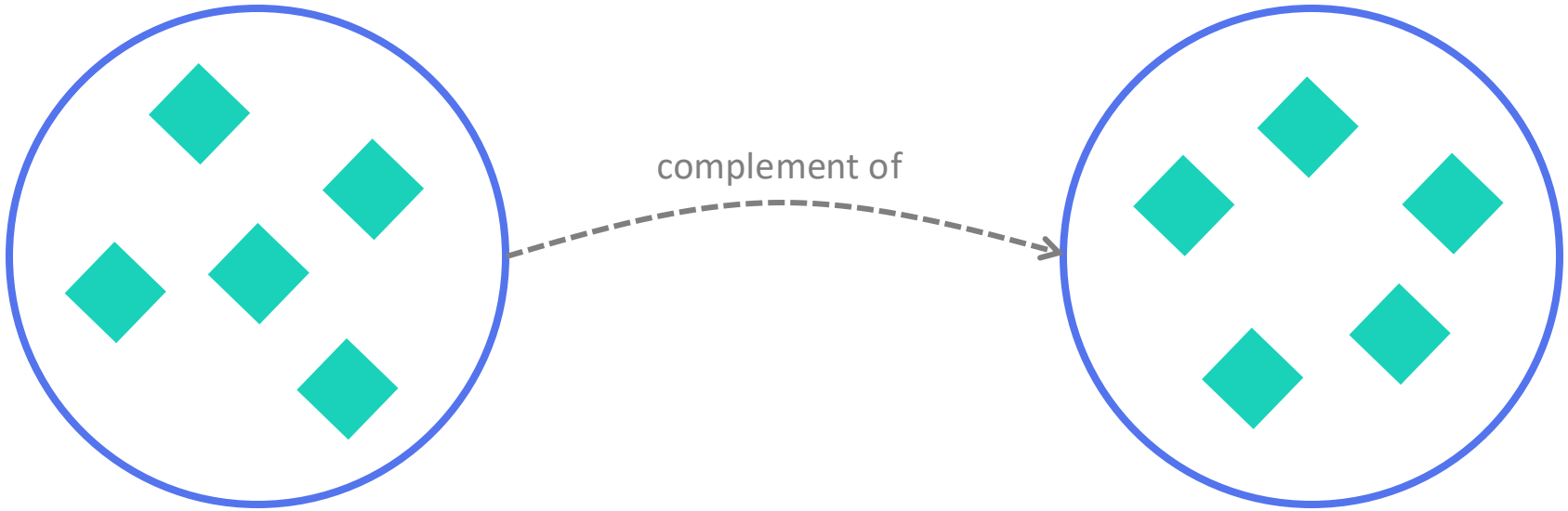
OWL Properties


2030-01-01T12:00:00
17
3.142

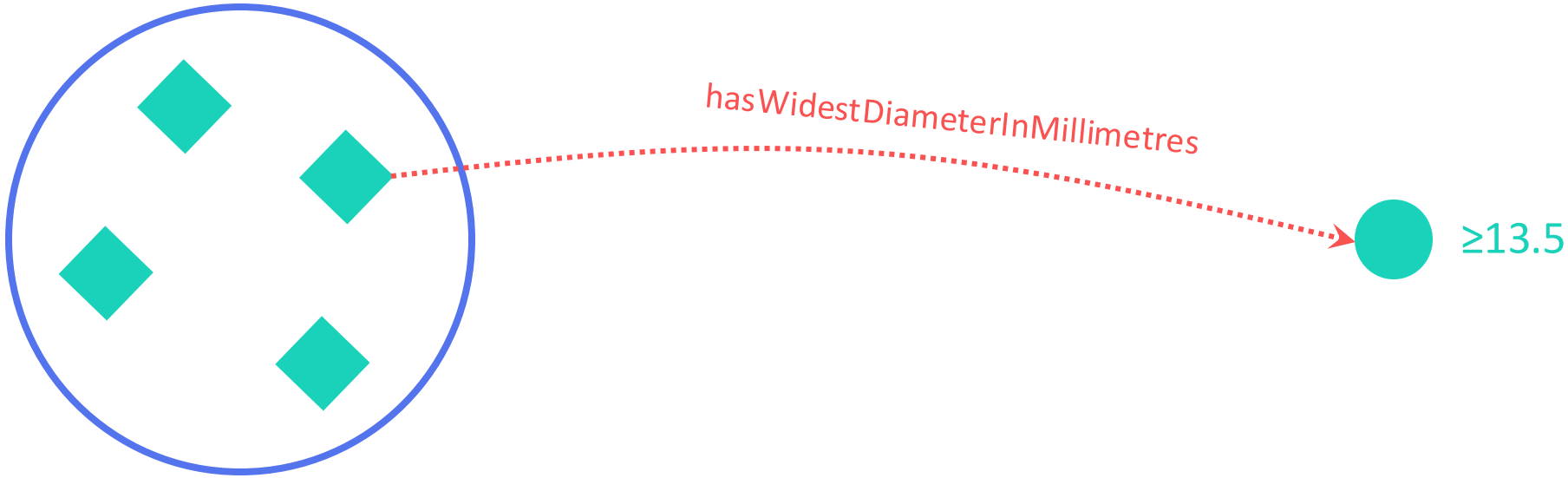
ChunkyBallpointPen



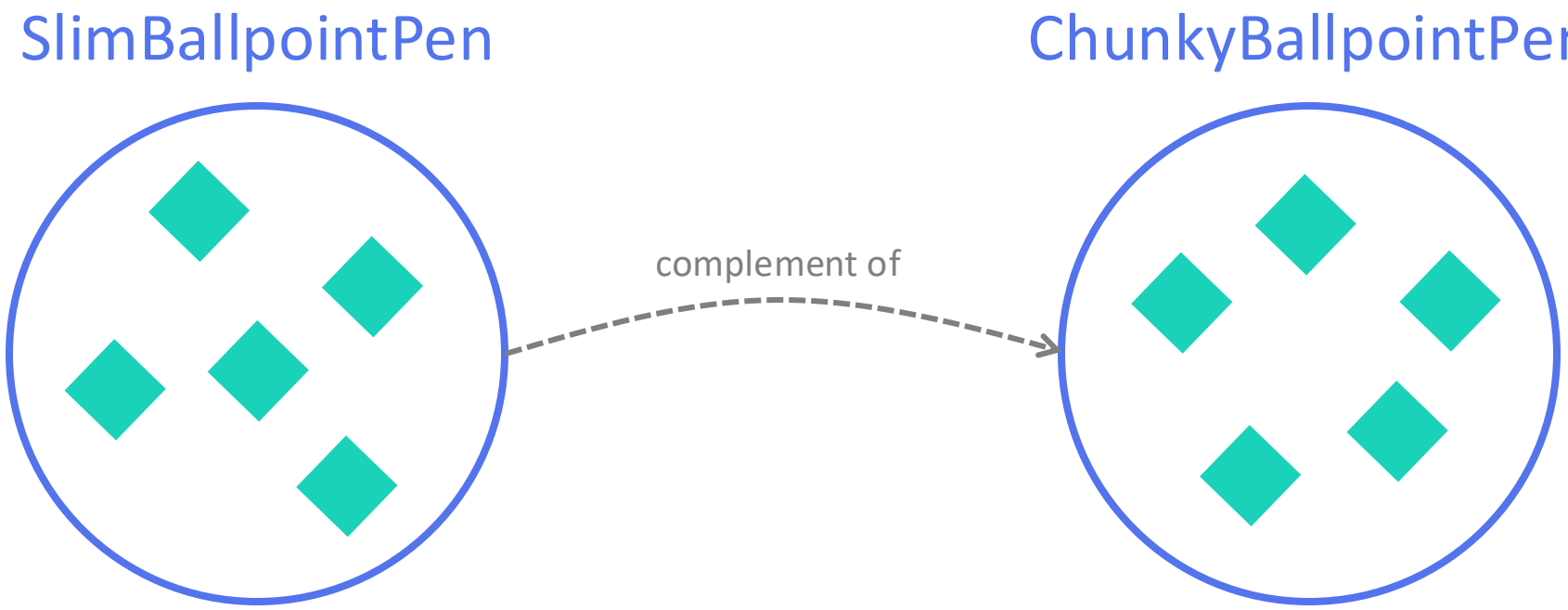

property
property
property
property


complement of

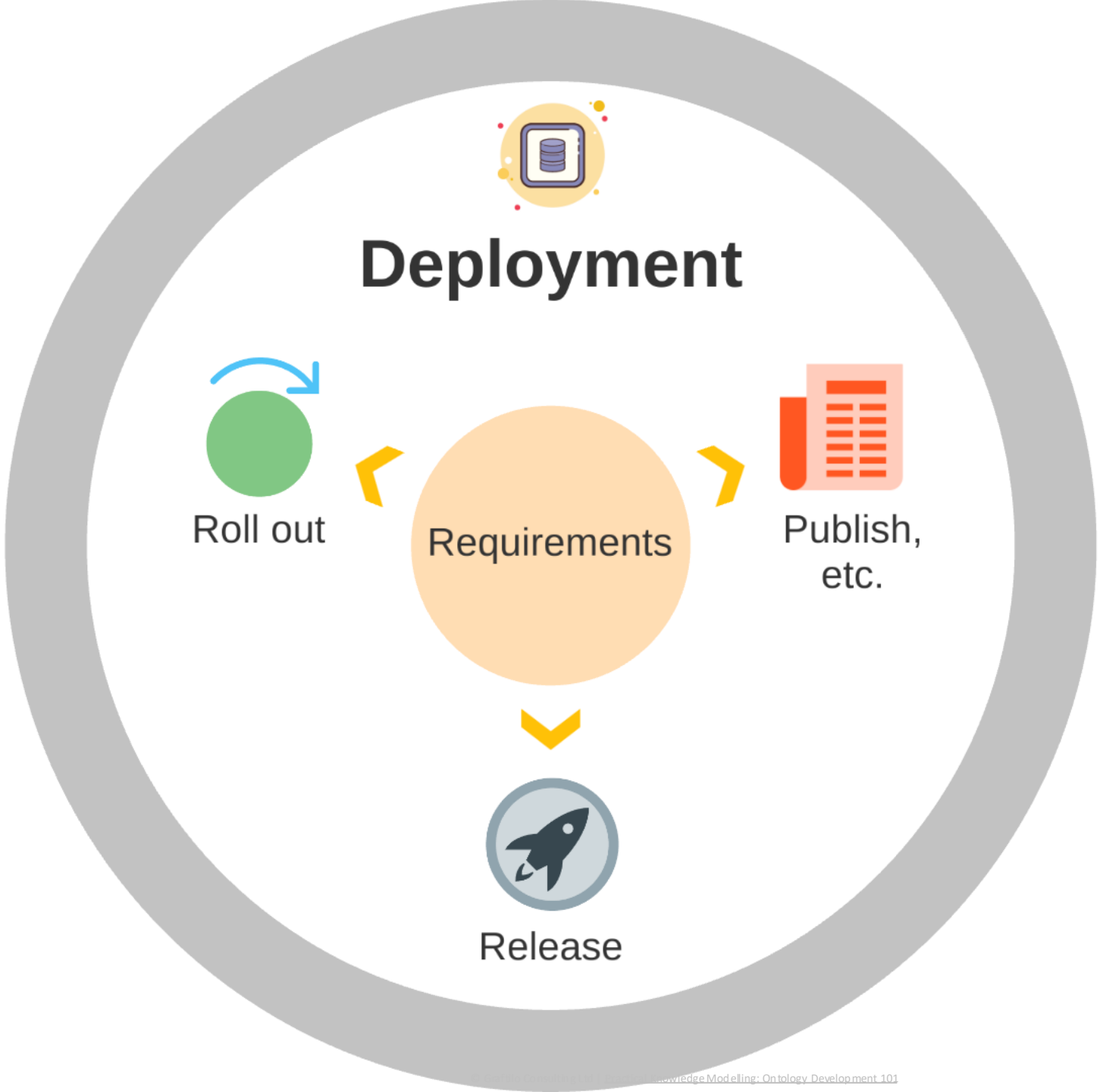
ChunkyBallpointPen




SlimBallpointPen
hasWidestDiameterInMillimetres
<13.5


SlimBallpointPen
ChunkyBallpointPen
complement of

06 Deployment


Deployment
Roll out
Requirements
Publish, etc.
Release

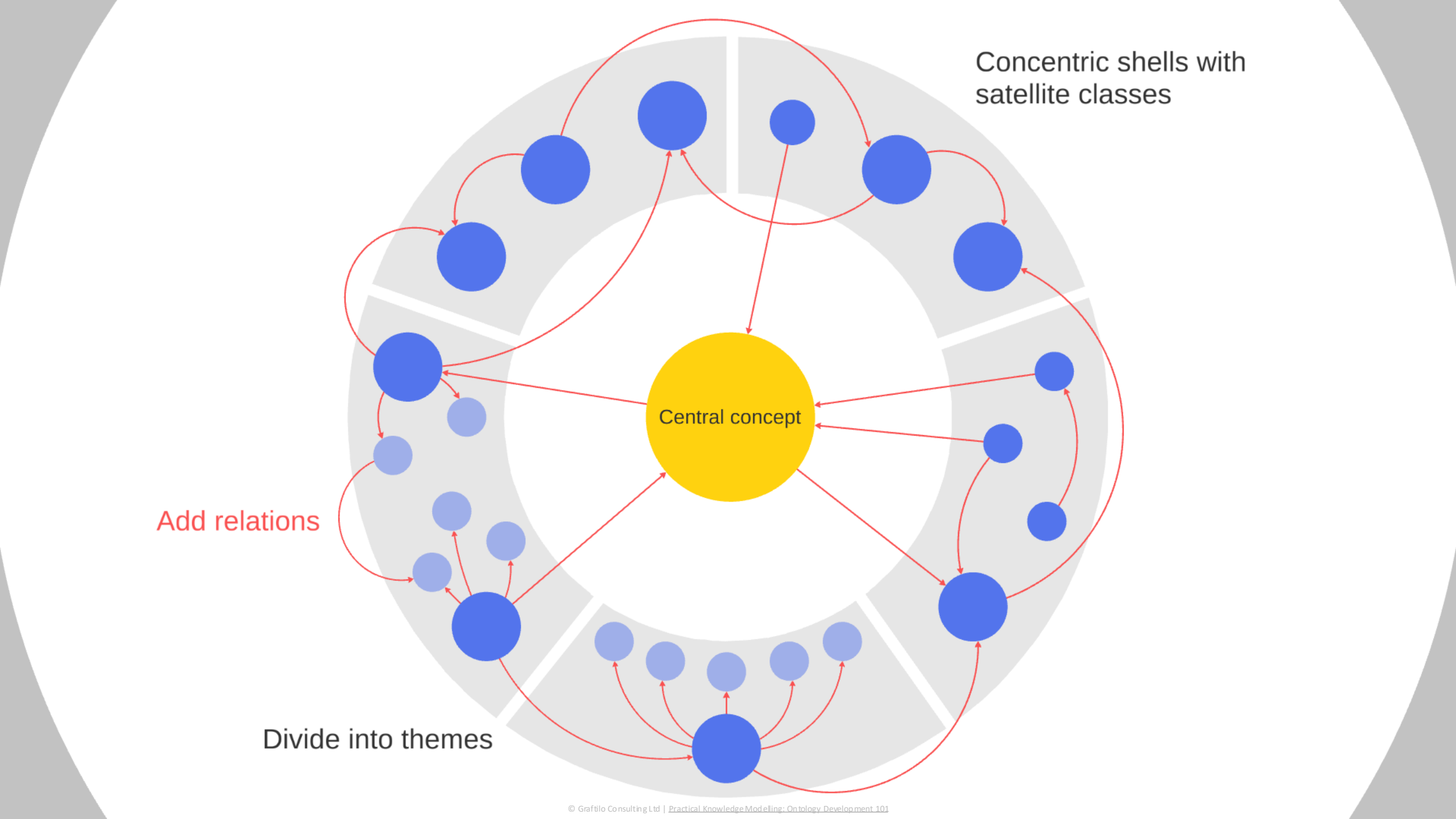
Example

Ballpoint Pen Ontology

HTML

Company Intranet

Radial diagrams


Concentric shells with satellite classes
Central concept
Add relations
Divide into themes

Ontology editor

Ontology deployment environment

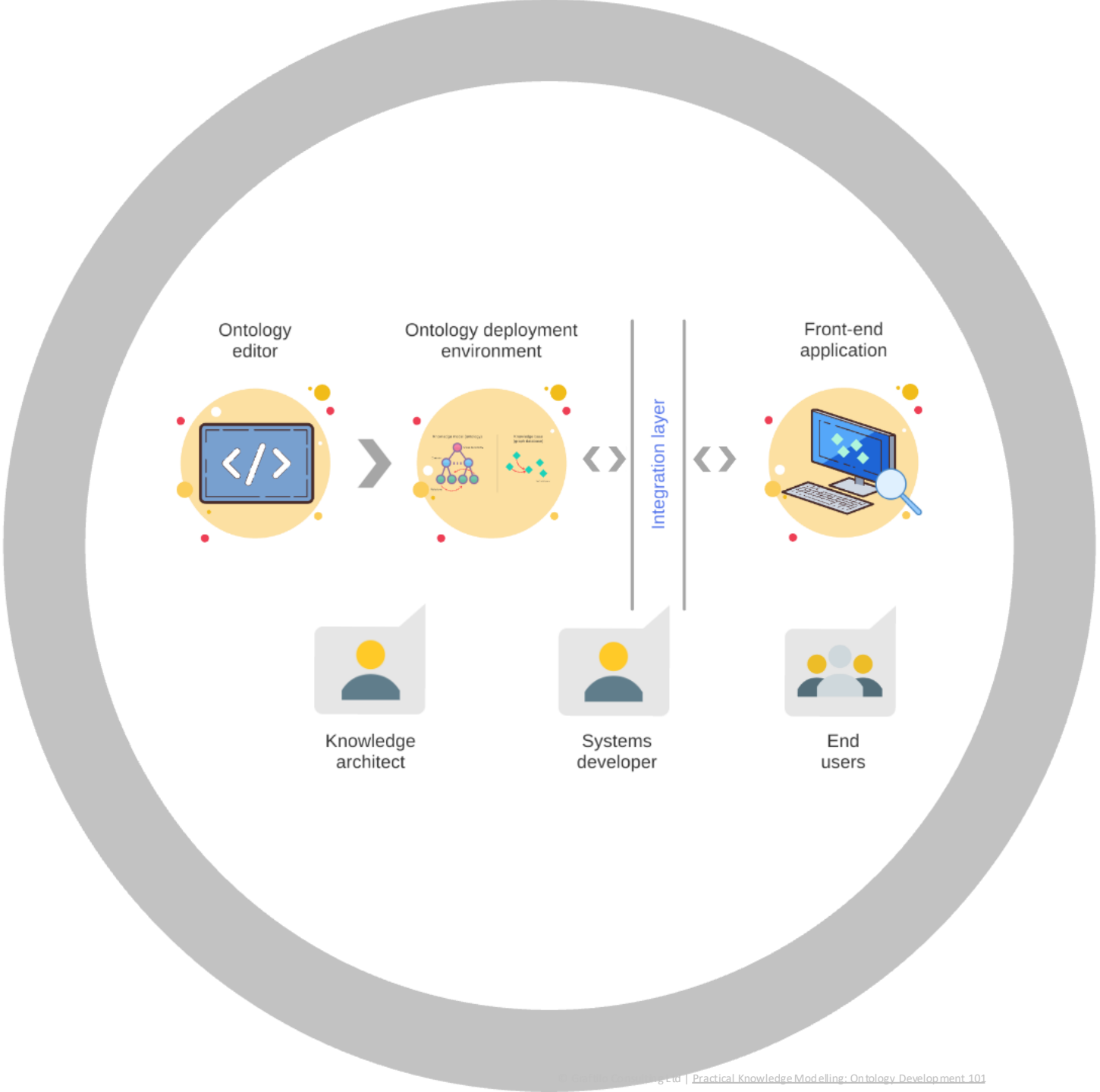
Integration layer

Front-end application

Knowledge architect

Systems developer

End users



- Infrastructure
- Resources
- Testing, etc.

07 Evaluation

Evaluation

Technical & specialist methods

- Ontology alignment
- Ontology comparison, etc.

hnical &
ecialist
ethods

Answering
business
questions

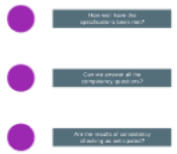
How well have the specifications been met?

Can we answer all the competency questions?

Are the results of consistency checking as anticipated?

Evaluation

Technical & specialist methods



Answering business questions

Knowledge management

Lessons learnt

Continuous improvement

Lessons learnt

Contin improv

Method	Depth of information	Cost	Time efficiency	Ease of following up	Level of structure supported
Brainstorming					
Survey					
Interview					
Focus group					
Content analysis					

Lessons learnt

- Understand roles & participants
- Knowledge maintenance & reuse
- Set up necessary workflows, systems, processes, etc.

- Failure to use an ontology development methodology
- Transformations across models are not fully understood 
- Inconsistent naming conventions
- Typos

UML

Generalisation

OWL

rdfs:subClassOf