Ontology-based Enterprise Modeling for Human and Machine Interpretation

Prof. Knut Hinkelmann

Models

Semantics

Models

not what we deal with
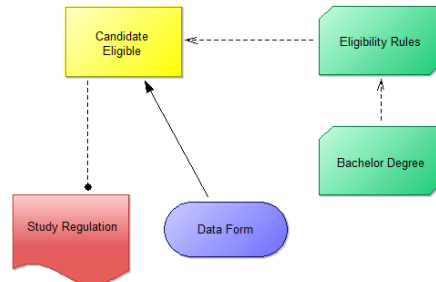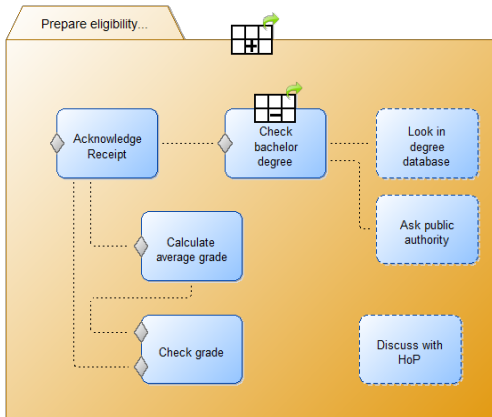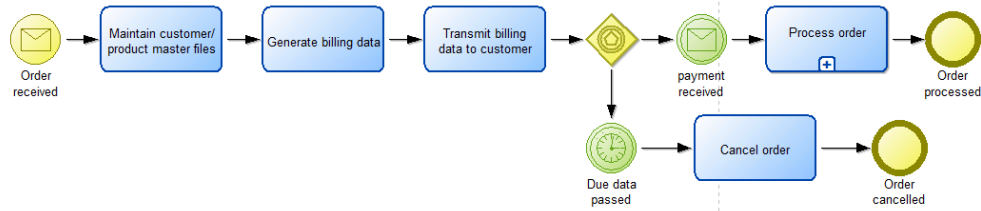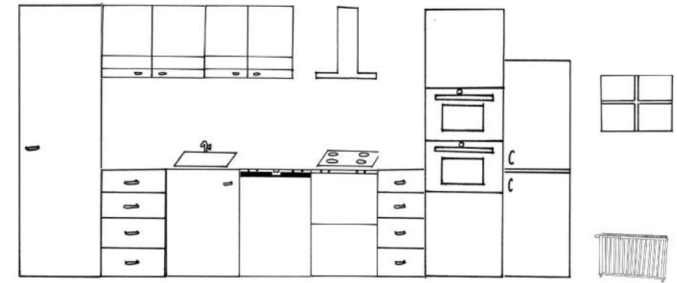
Enterprise Models

Upps! Still not what we deal with

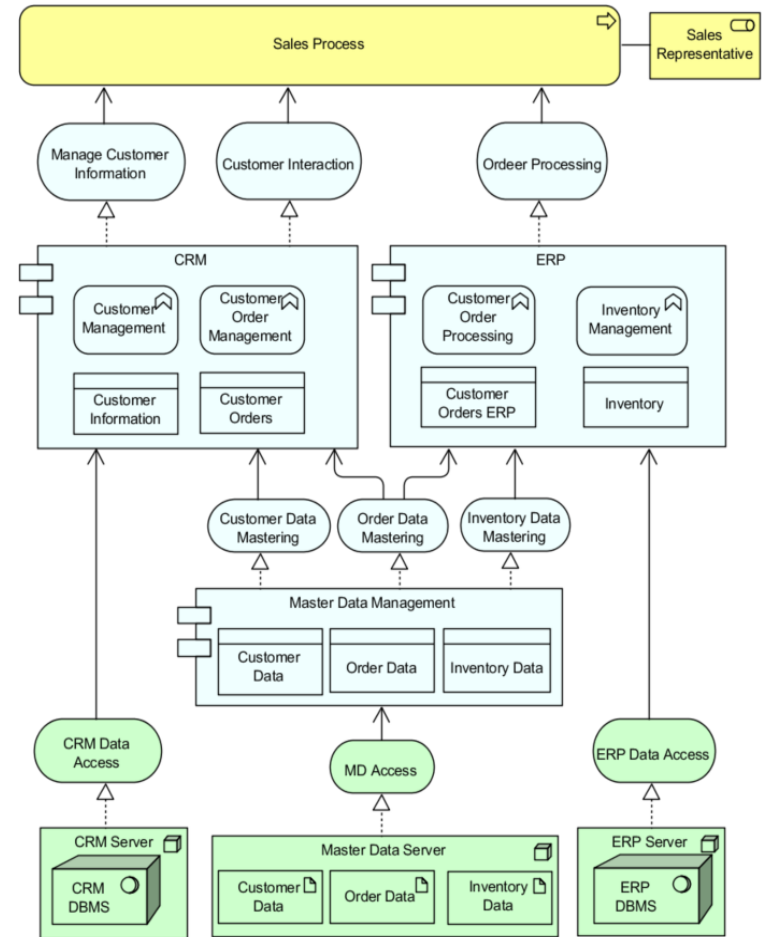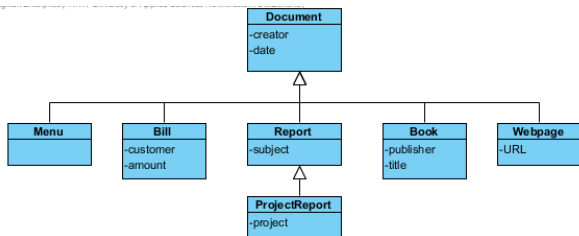# *Enterprise Modeling*

Modelling: Reproduction of Relevant Part of Reality

Communication/
Analysis/
Decision Making

Models
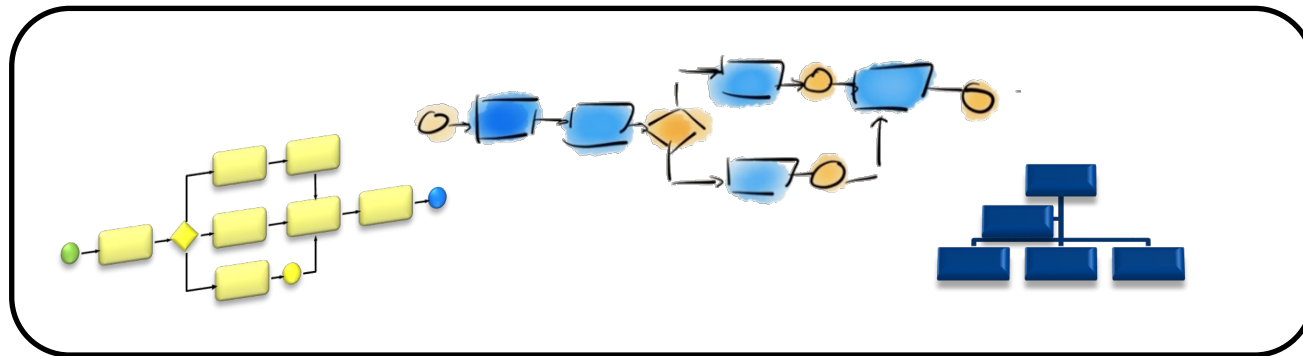
Reality

Collaboration

Shared understanding the situation, agree on a common goal, determine a way to achieve a goal

Models in Collaboration: Make Information Explicit
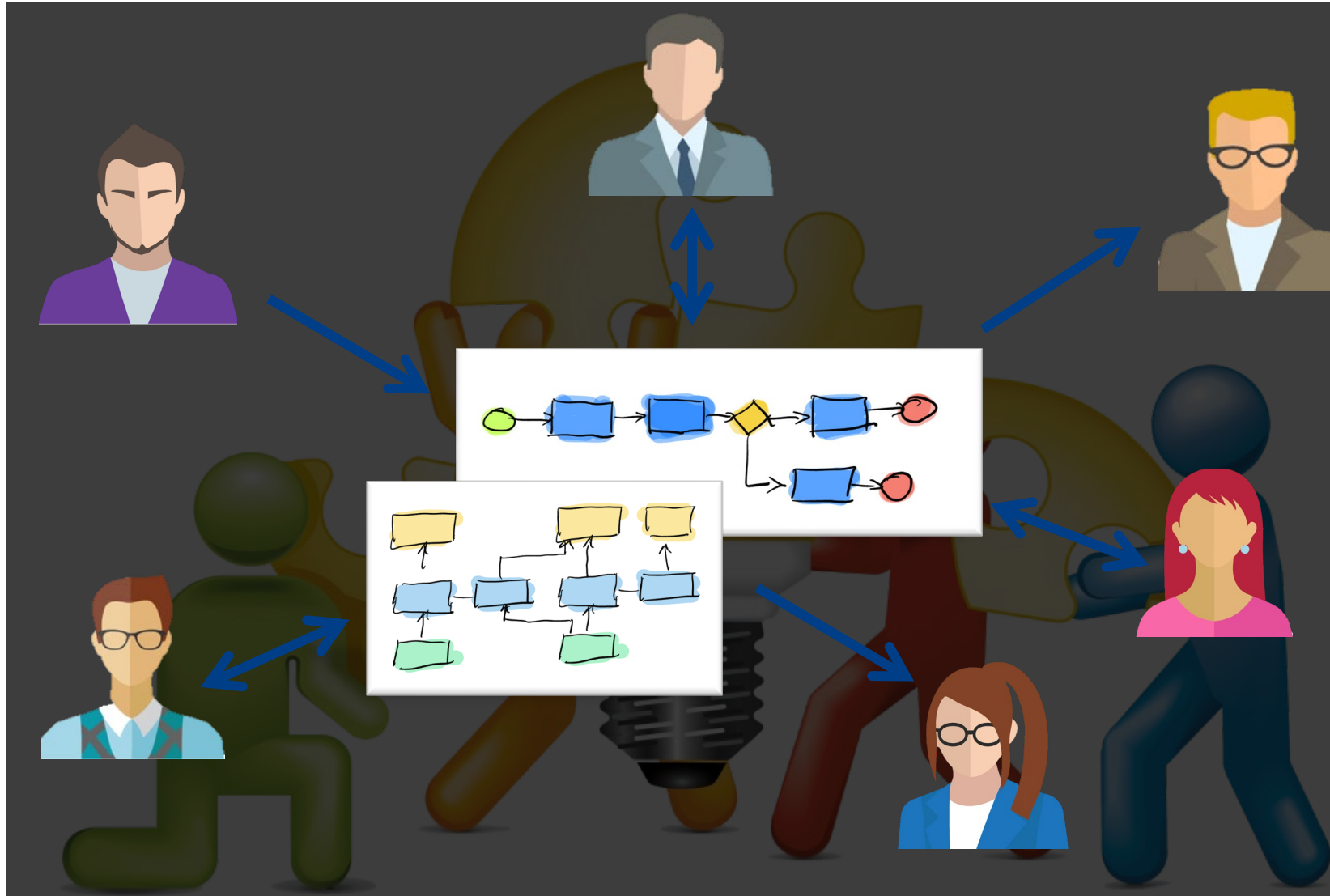
"What is not made explicit is **implicit**, which allows everyone to make **assumptions**.
Incorrect assumptions are sources of miscommunication and **misunderstanding**"

John Zachmann

Meaning of Models

Cartoon adapted by K. Hinkelmann
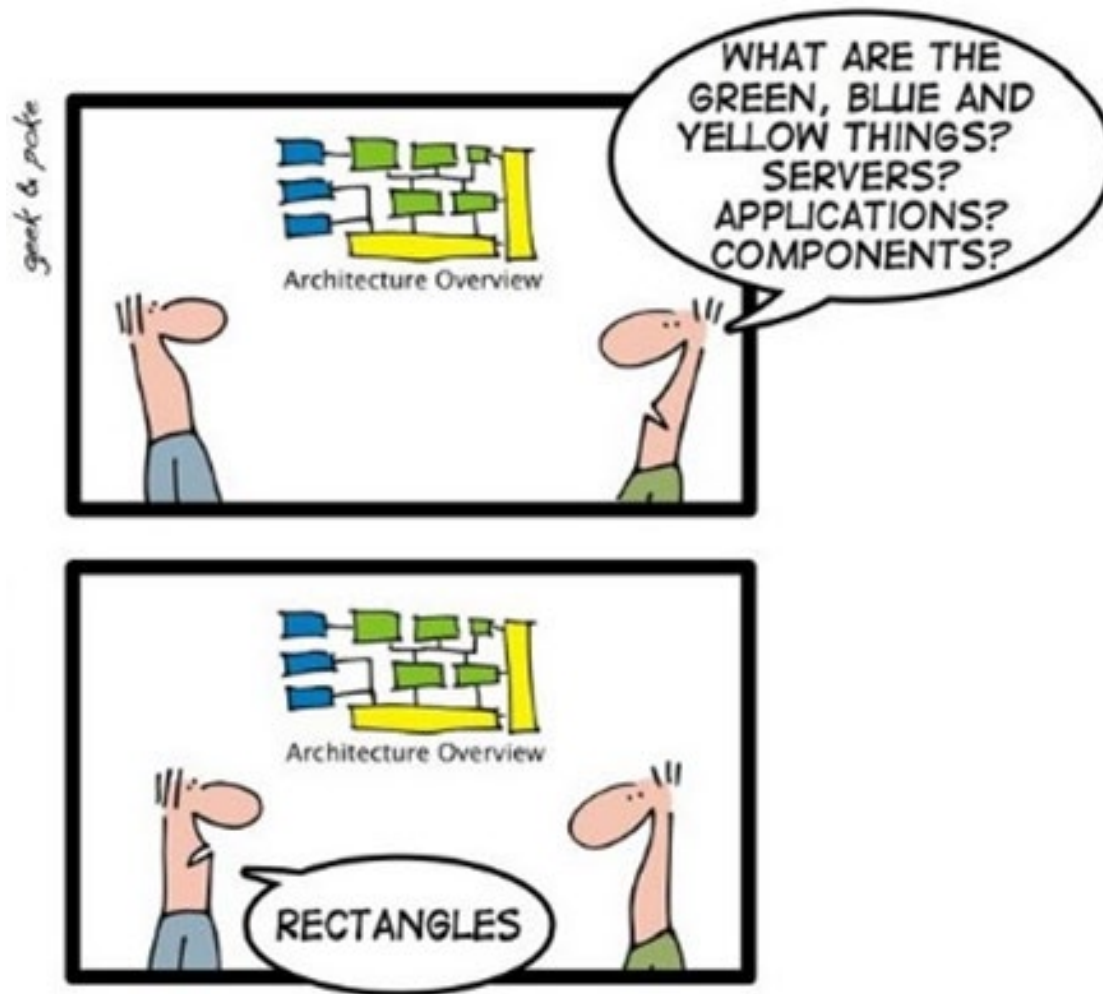
Models

■ Models are not mere pictures; rather, they

♦ provide a precise, meaningful description that can be visualized in different ways for different stakeholders;

♦ can also be used to analyze the impact of changes, cost, risk, security, compliance and other relevant KPIs.

http://blog.bizzdesign.com/how-to-not-fail-when-implementing-strategy

Interpretation of Models
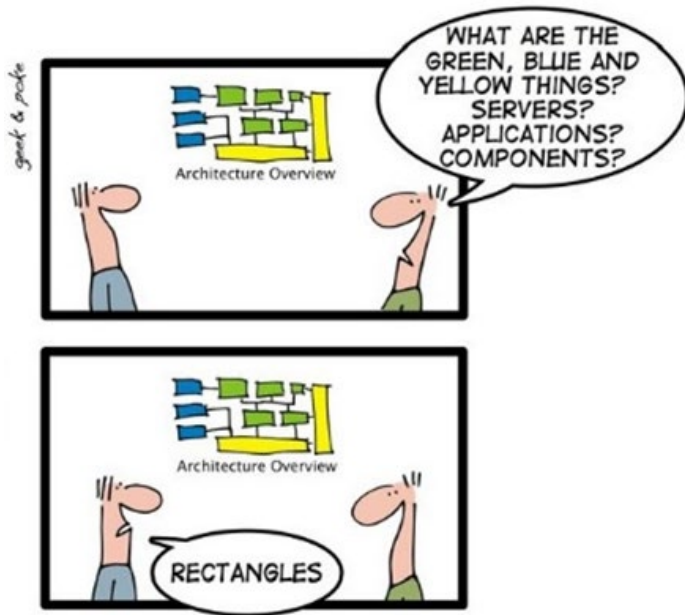
- Models are data

- Models become information by interpretation

- Interpretation requires knowledge

Interpretation of Models requires Knowledge

caption: Communication/ Analysis/ Decision Making — Knowledge

Models — human-interpretable models

Reality

What knowledge is needed to interpret this Model?
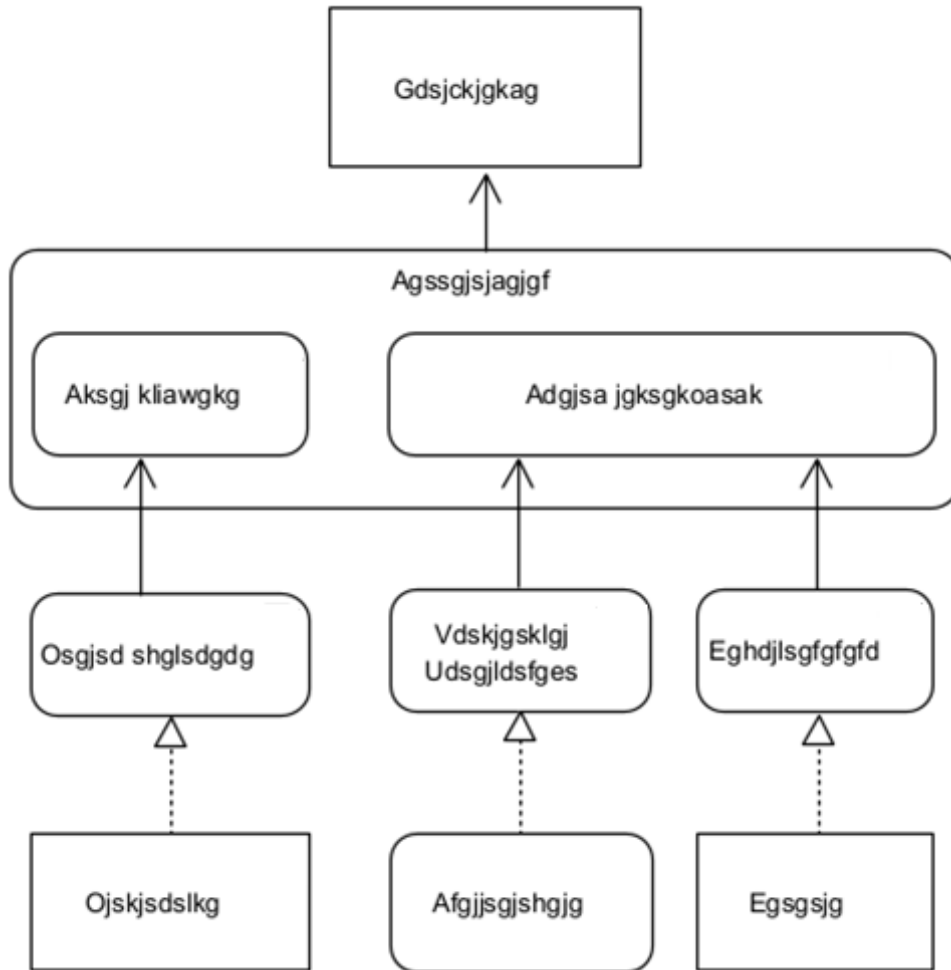
What knowledge is needed to interpret this Model?

Knowledge about modelling concepts (= modelling language)

What knowledge is needed to interpret this Model?

Knowledge about modelling concepts (= modelling language)

Knowledge about application domain
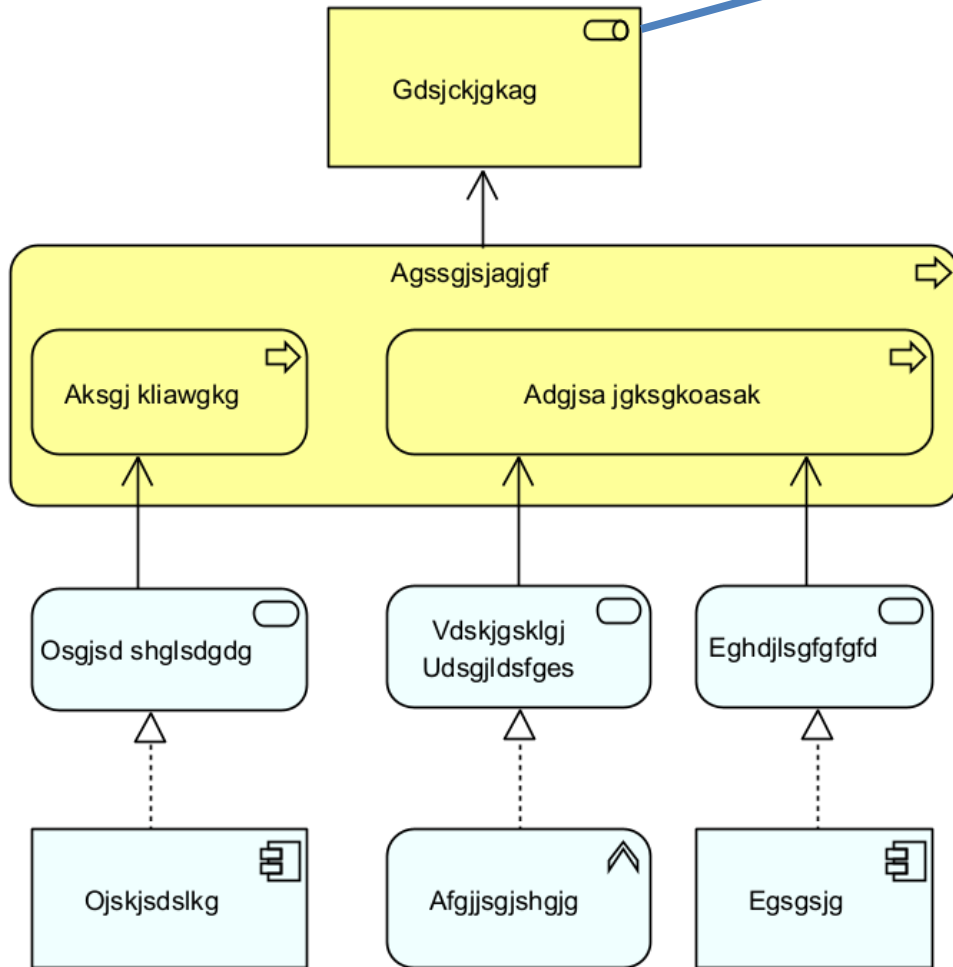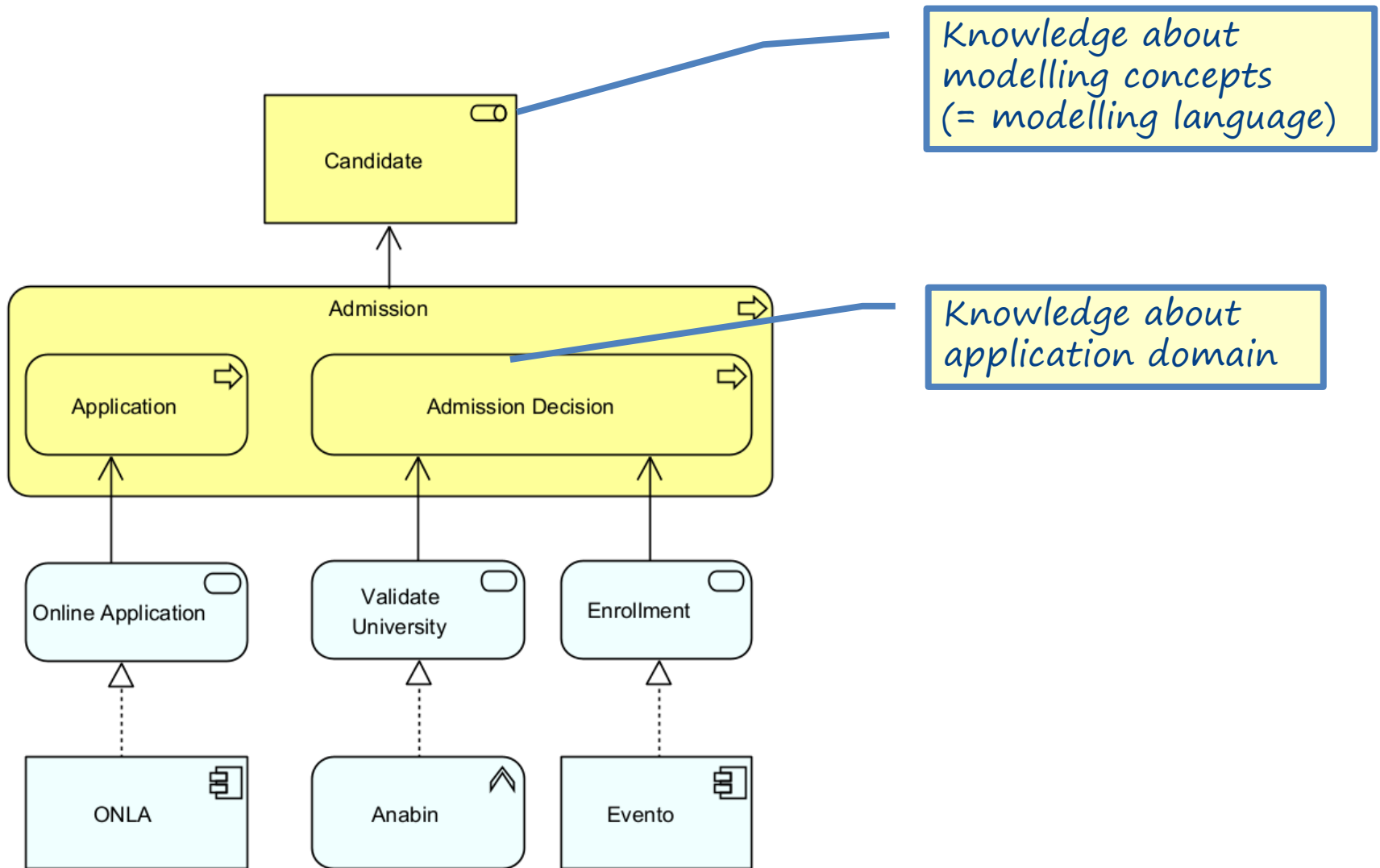
Modelers and Users must have same Interpretation

- Interpretation requires knowledge
- Modelers and users must have the same knowlege
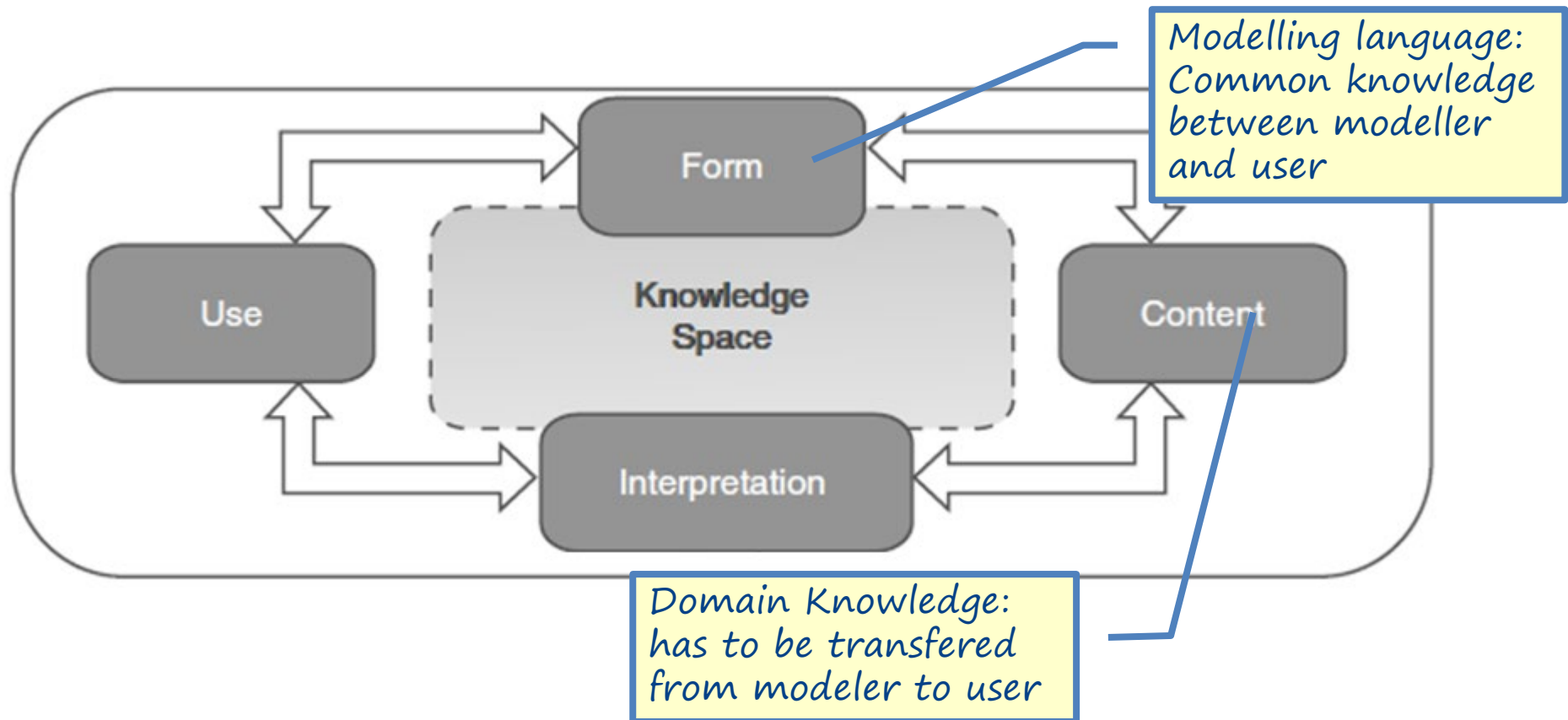
Modelling language: Common knowledge between modeller and user

Domain Knowledge: has to be transfered from modeler to user

Karagiannis, D., & Woitsch, R. (2010). Knowledge Engineering in Business Process Management. In *Handbook on Business Process Management 2* (pp. 463–485). Springer.

Making Knowledge explicit

■ Remember:

"What is not made explicit is implicit, which allows everyone to make assumptions.
Incorrect assumptions are sources of miscommunication and misunderstanding"

■ Make knowledge required for interpretation explicit

♦ Knowledge about modeling concepts
➢ modeling language

♦ Knowledge about the domain
➢ ???

Prepare Fettuccine

Evento

Student administration system
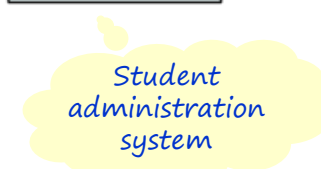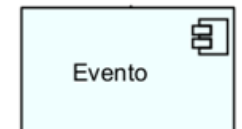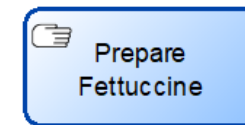
Human Interpretation of Models

Modelling

1. Acquiring information about application domain

2. Representing the information in models

→ 3. Managing knowledge required for model interpretation
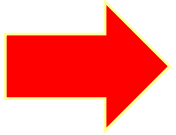
Human Interpretation requires Knowledge Management
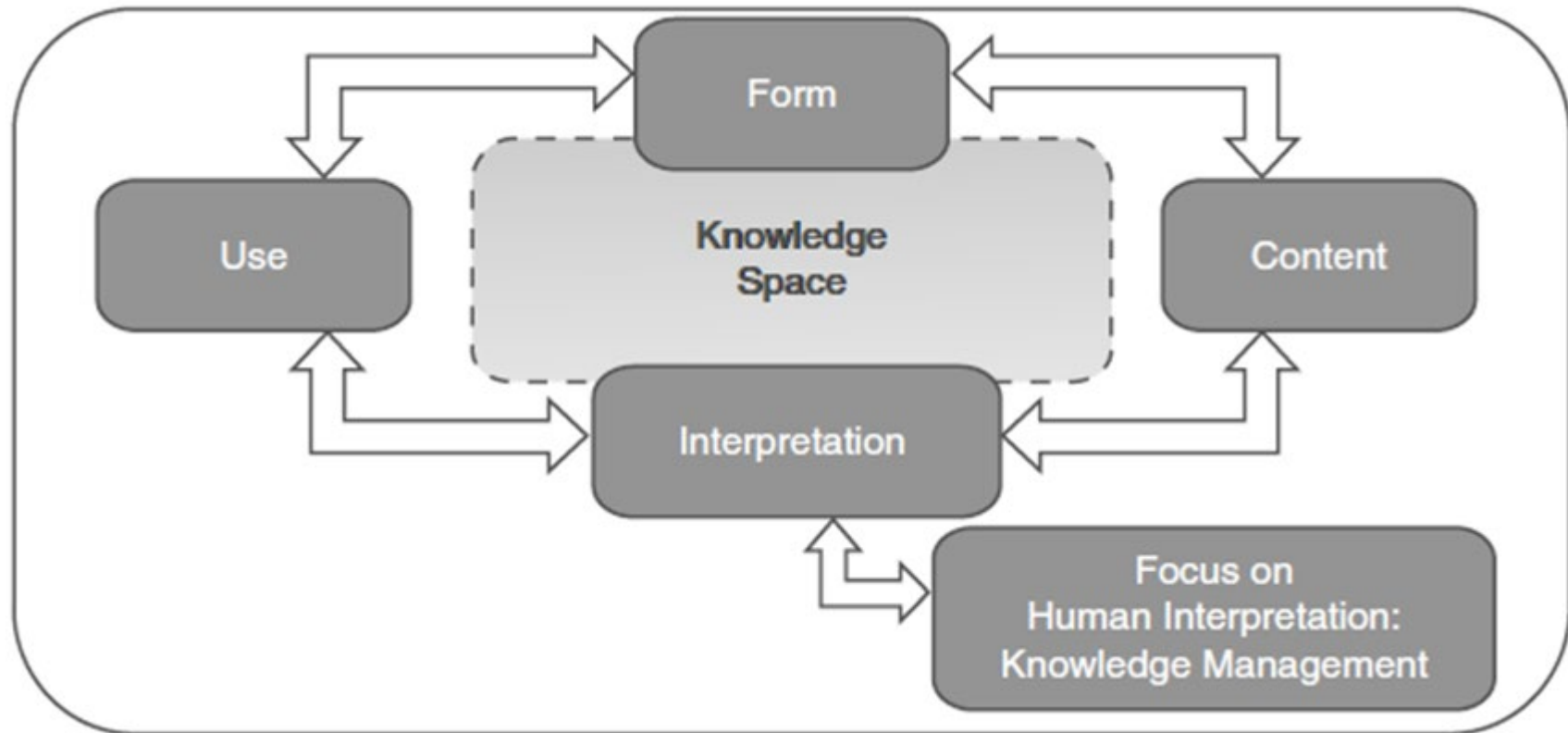
Karagiannis, D., & Woitsch, R. (2010). Knowledge Engineering in Business Process Management.
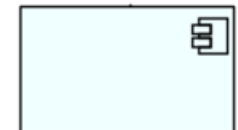In *Handbook on Business Process Management 2* (pp. 463–485). Springer.

Models in Collaboration: Share Information and Knowledge
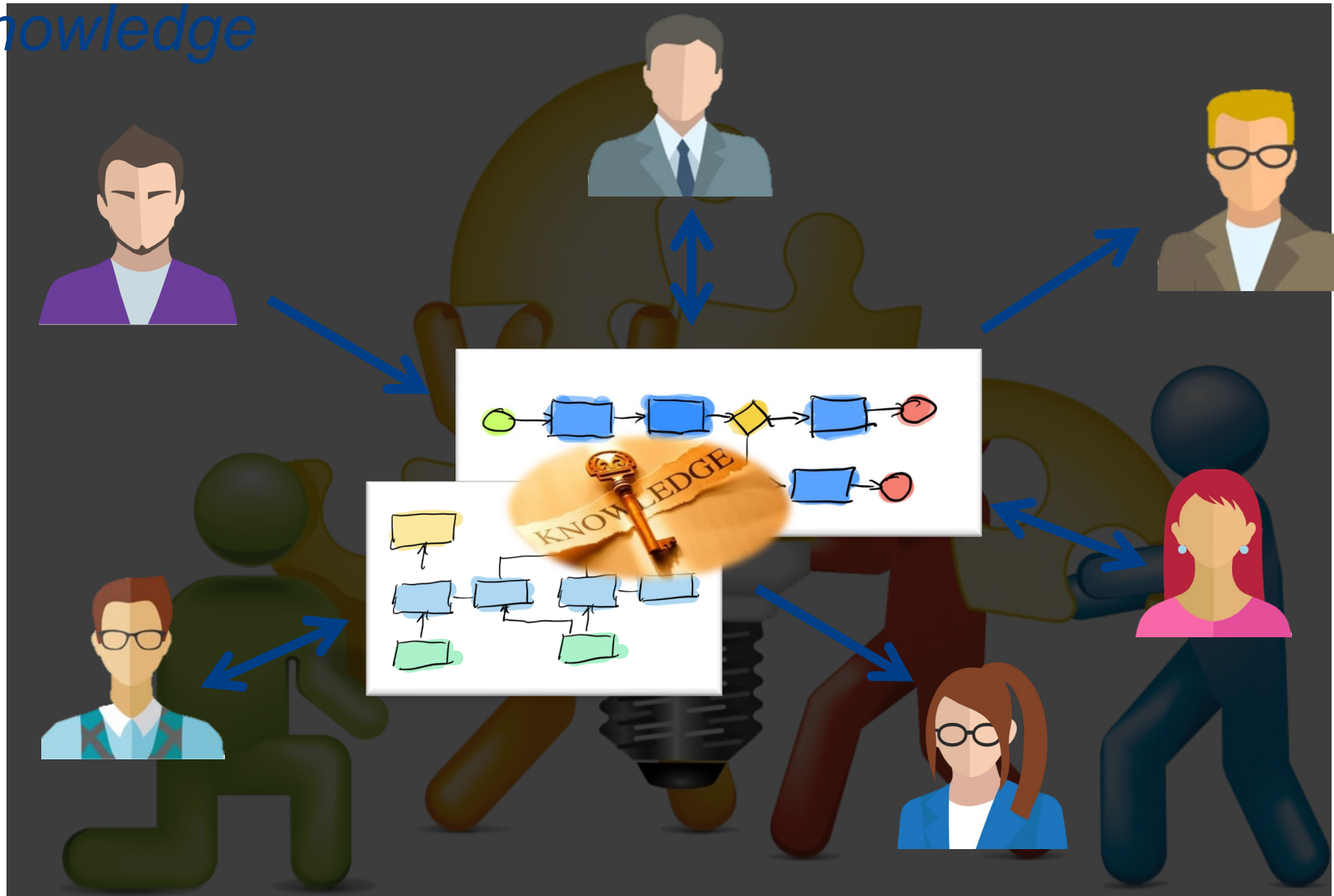
Interpreting Models requires up-to-date Knowledge

Knowledge Management: Sharing Knowledge between Modelers and Model Users

■ Make domain knowledge required for interpretation explicit

■ Example: Glossary

Name	Aliases	Description
ONLA		The tool for online applications of students developed by the FHNW
Anabin		A web application provided the the German Kultusministerkonferenz, which contains information about recognition of universities
Candidate	Prospective student	A person who applied for a study program
Student		A person who is enrolled into a study program at a university
University		An education and research institution on tertiary level.
Admission		The process of deciding, wether a candidate is eligible for a study program.
Eligibility	acceptable	A candidate is eligible, if he/she satisfies all the eligibility criteria
Evento		The student administration system, which contains all information about student, staff and study programs.

Linking Models to Domain Knowledge

Name	Aliases	Description
ONLA		The tool for online applications of students developed by the FHNW
Anabin		A web application provided the the German Kultusministerkonferenz, which contains information about recognition of universities
Candidate	Prospective student	A person who applied for a study program
Student		A person who is enrolled into a study program at a university
University		An education and research institution on tertiary level.
Admission		The process of deciding, wether a candidate is eligible for a study program.
Eligibility	acceptable	A candidate is eligible, if he/she satisfies all the eligibility criteria
Evento		The student administration system, which contains all information about student, staff and study programs.

Machine-Interpretation of Knowledge

Objective

Represent the knowledge about models so that it can be interpreted by a system for decision making and problem solving

Objective

> Models should allow automated analysis, decision making and digitalization.

Approach

> Represent the models and knowledge so that it can be interpreted by a system for decision making and problem solving

Dimensions of a Knowledge Space
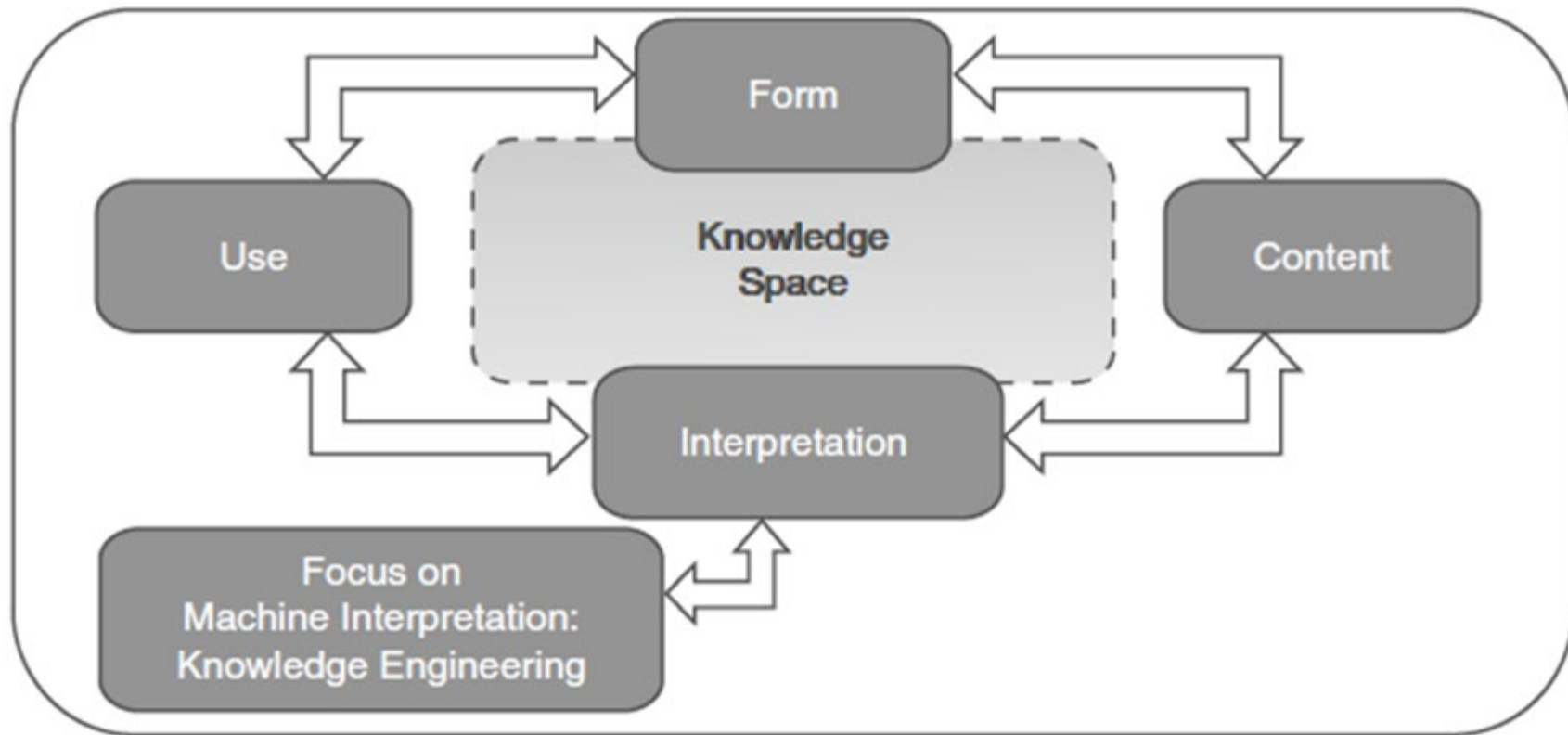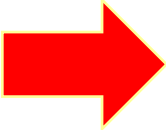
Karagiannis, D., & Woitsch, R. (2010). Knowledge Engineering in Business Process Management.
In *Handbook on Business Process Management 2* (pp. 463–485). Springer.

Objective

Models should allow automated analysis, decision making and digitalization.

- Examples:
    - ♦ Validate architecture principles
    - ♦ Select appropriate cloud services
    - ♦ Plausibility checks

Modelling

1. Acquiring information about application domain

2. Representing the information in models

3. Representing knowledge required for model interpretation

Semantic Lifting

Semantic Lifting: Map Models into an Ontology
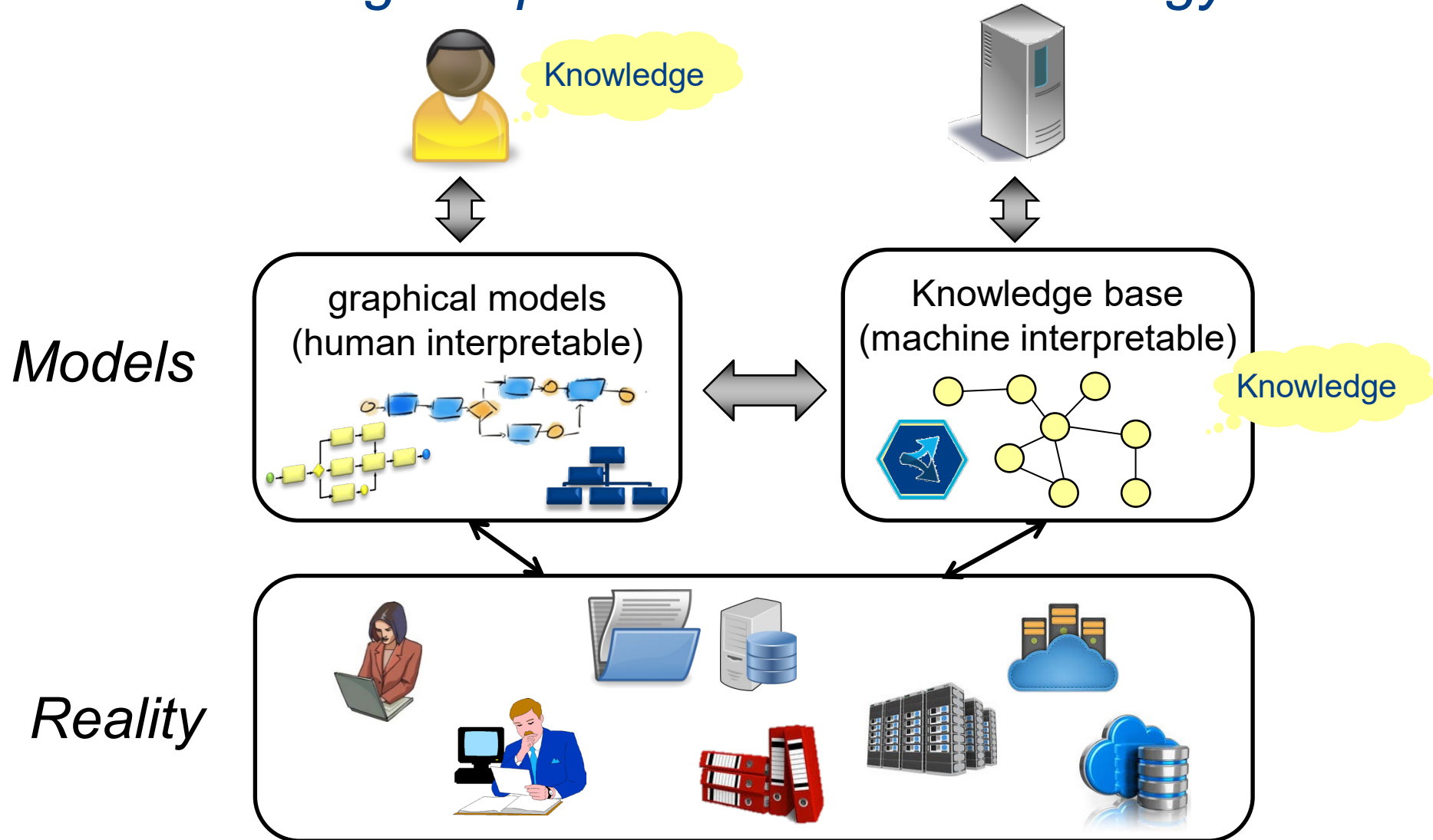
Modelling Environment

Ontology

meta²-model layer

Definition of syntax and (type) semantics

Semantics definition: commonly accepted ontology

metamodel layer

notations   classes

model layer
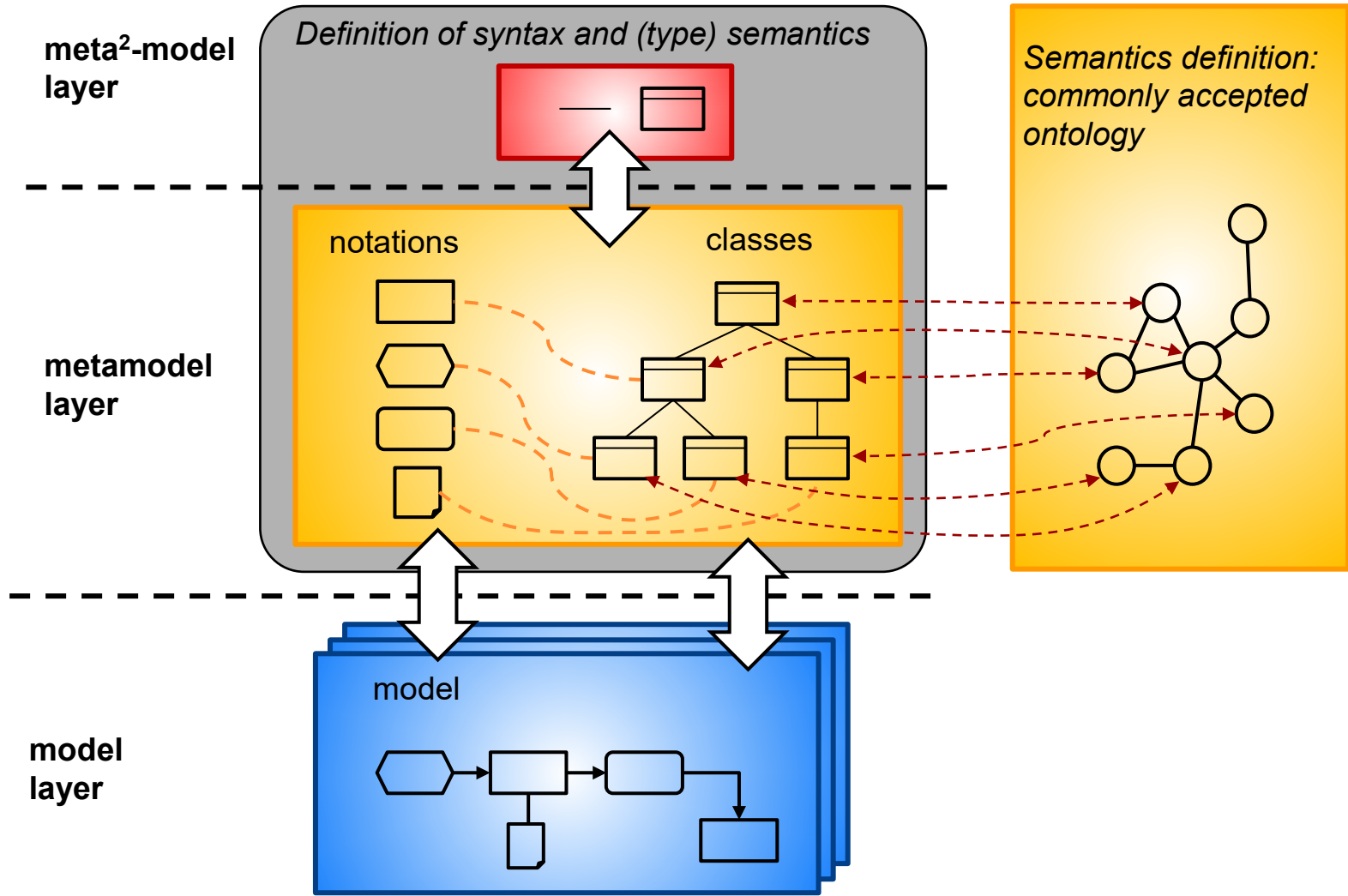
model

⬌ **linguistic metamodelling:** *syntax, structure, type semantics*

⟵---⟶ **ontological metamodelling (lifting):** *explication of type semantics*

Example: Business Process as a Service

What are appropriate services to run this process in the cloud?
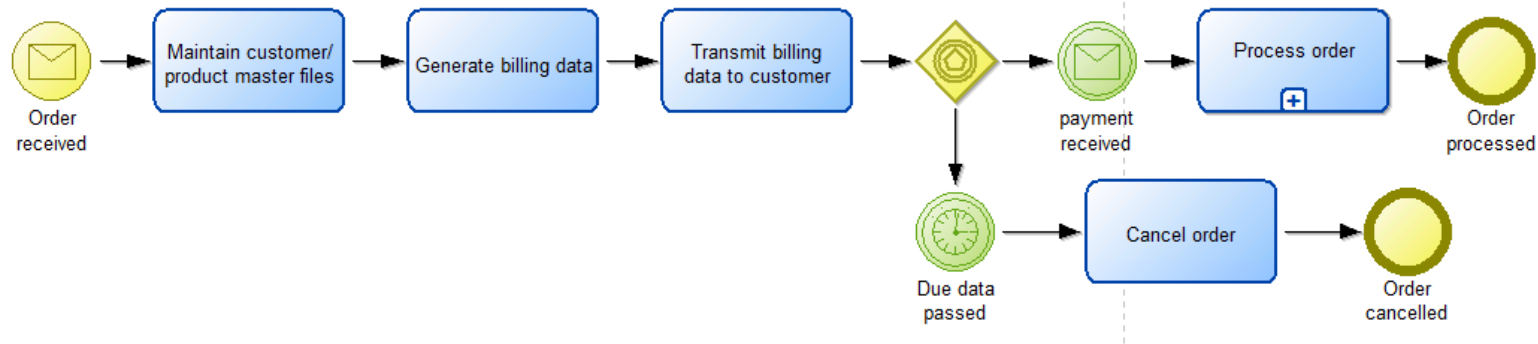
From: CoudSocket Project

Example: Business Process as a Service

human interpretation
informal and semi-formal

machine interpretation
formal
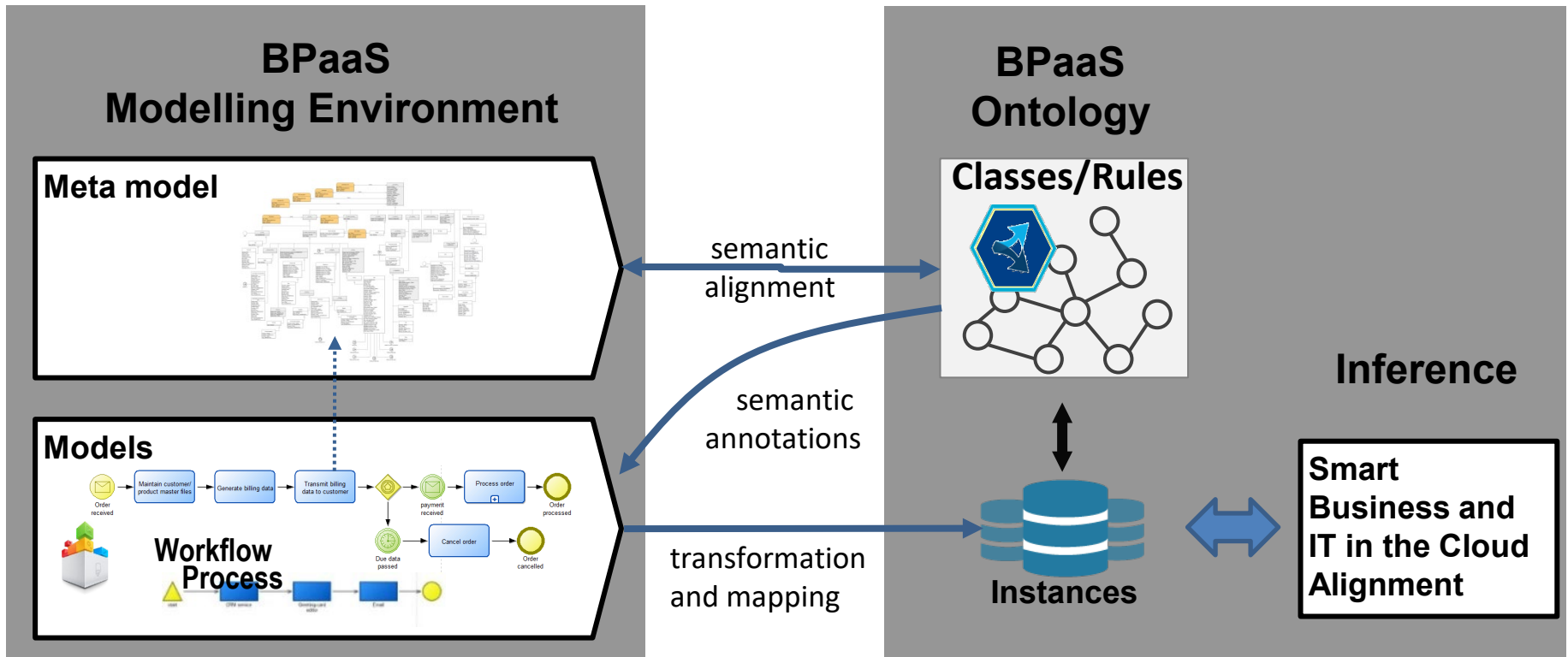
From: CoudSocket Project

Semantic Alignment

human interpretation
informal and semi-formal

machine interpretation
formal

The semantics of the meta-model elements is defined in the BPaaS Ontology

BPaaS Modelling Environment

BPaaS Ontology

Meta model

Classes/Rules

semantic alignment

semantic annotations

Models

Workflow Process

Inference

transformation and mapping

Instances

Smart Business and IT in the Cloud Alignment
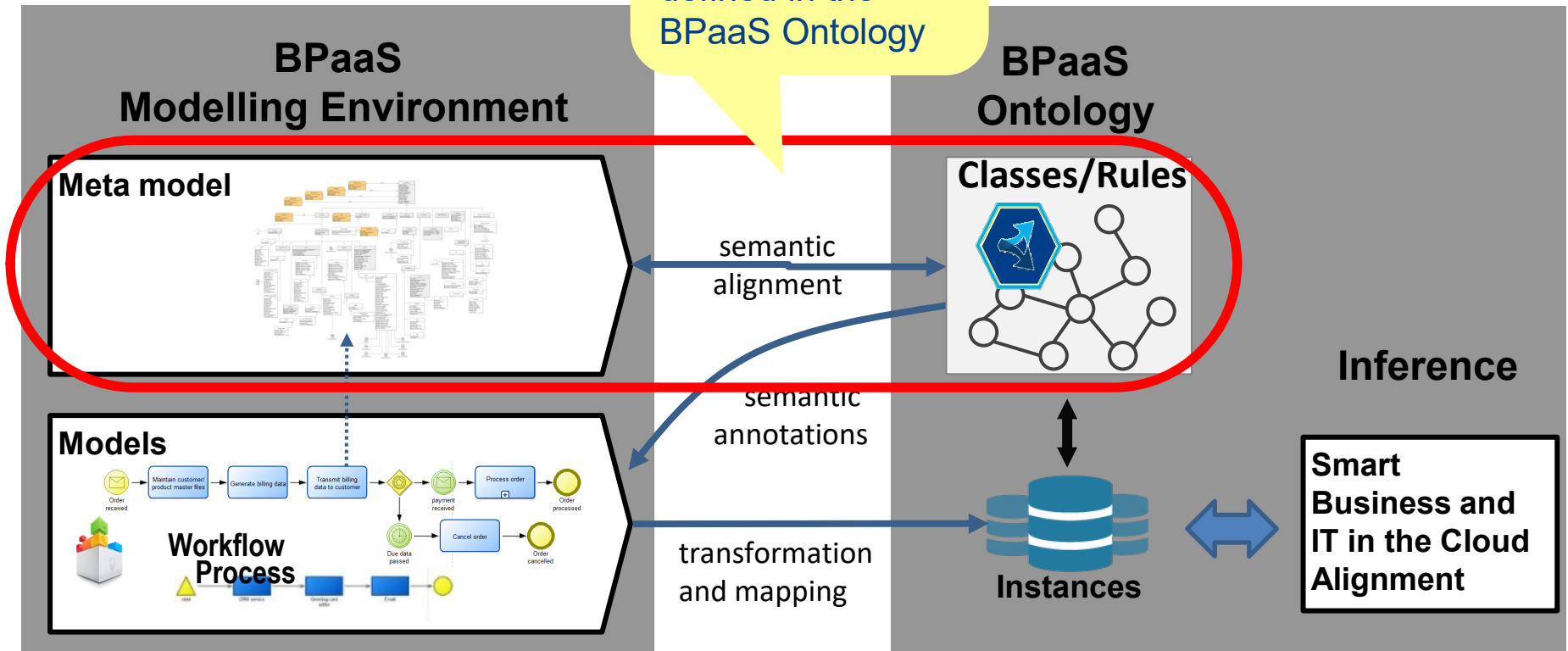
From: CoudSocket Project

Ontology with BPMN Modeling Concept

BPMN Ontology:
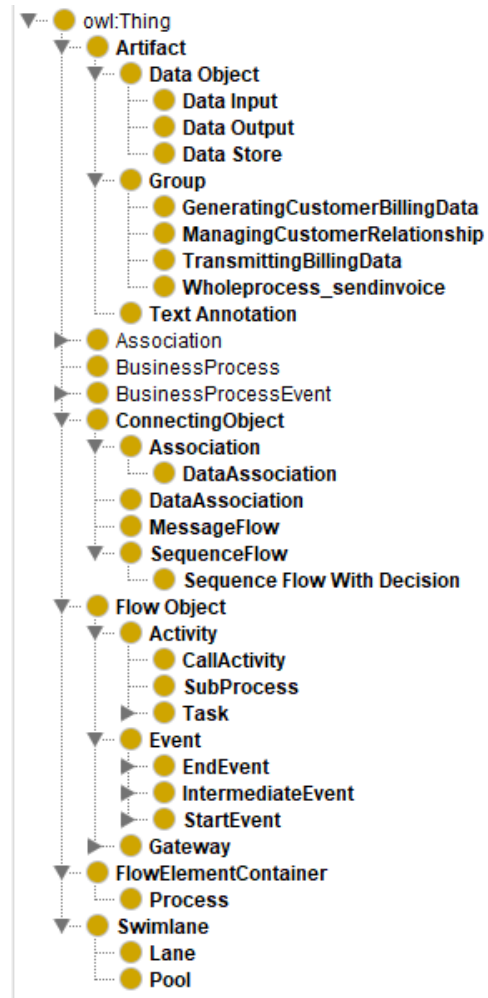
Example: Business Process as a Service

Given a BPMN business process, retrieve all the Cloud Services that satisfy the functional and non-functional requirements.

Challenge: BPMN does not accommodate functional nor non-functional req. for Cloud Services.
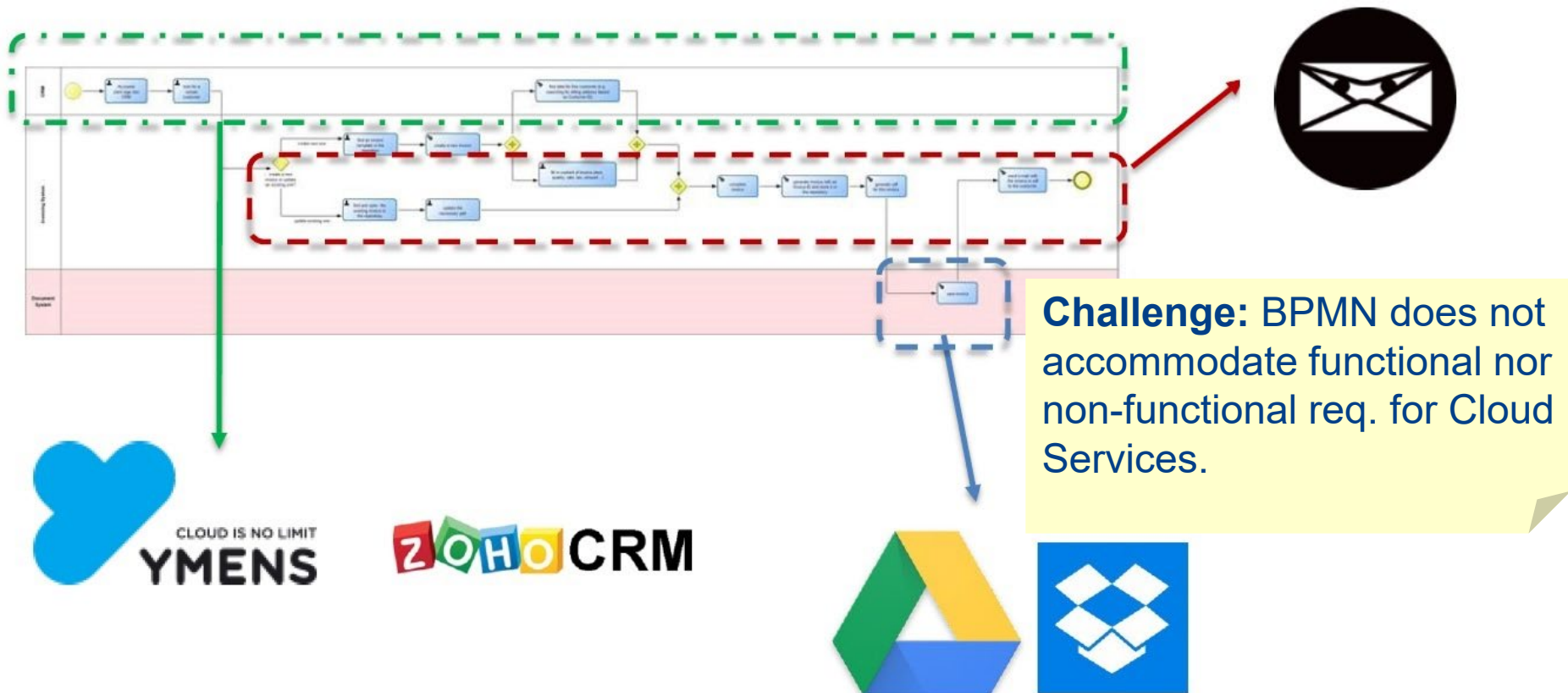
Semantic Annotation

human interpretation
informal and semi-formal

machine interpretation
formal

Define concepts for functionality in ontology and assign to process

BPaaS Modelling Environment

BPaaS Ontology

Meta model

Classes/Rules

semantic alignment

Models

semantic annotations

Workflow Process

transformation and mapping

Inference

Instances

Smart Business and IT in the Cloud Alignment
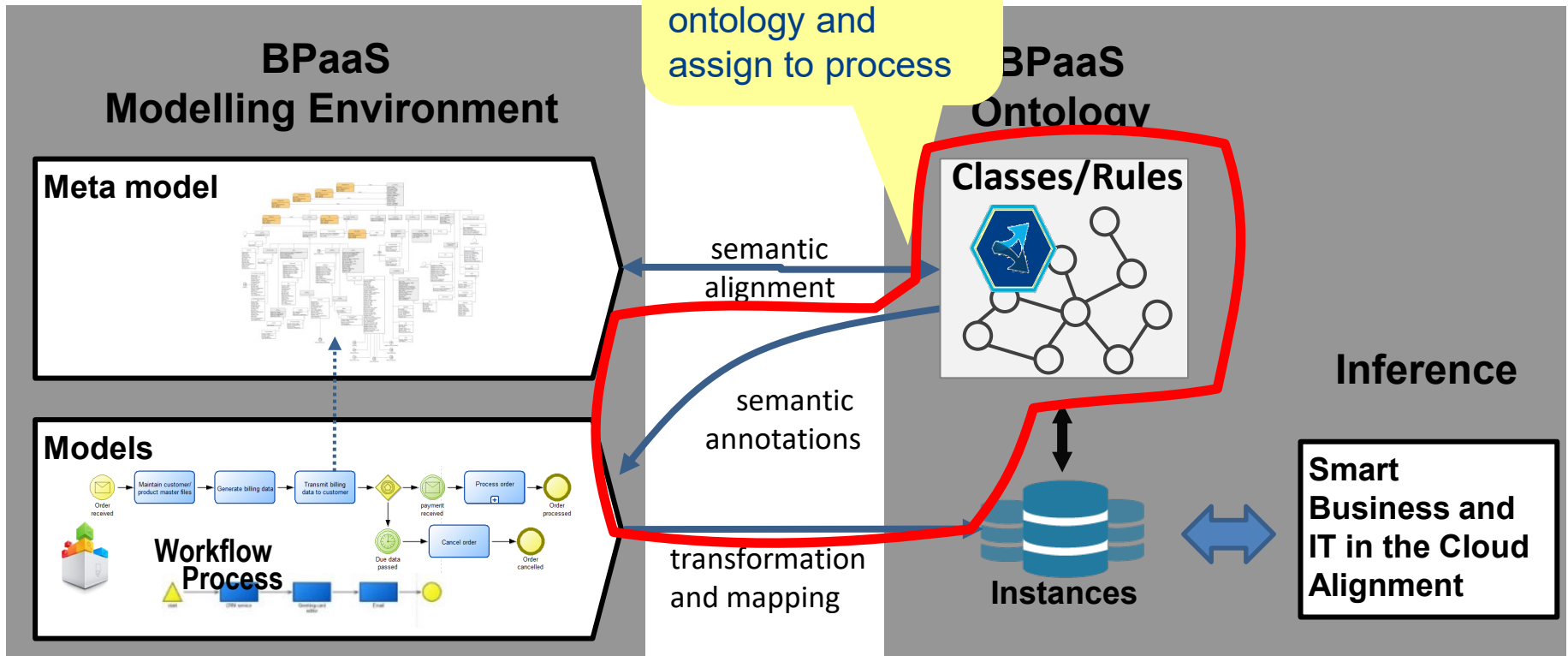
From: CoudSocket Project

Application Knowledge: Semantic Annotations

Annotate modeling elements with classes from the domain ontology

Send Invoice

Example: Functionality of a Service

Domain Ontology:
APQC Process Classification Framework

▼ ● owl:Thing
  ▼ ● American Productivity and Quality Center
    ▶ ● Acquire, Construct, and Manage Assets
    ▶ ● Deliver Physical Products
    ▶ ● Deliver Services
    ▶ ● Develop and Manage Business Capabilities
    ▶ ● Develop and Manage Human Capital
    ▶ ● Develop and Manage Products and Services
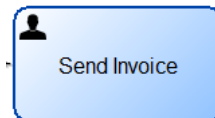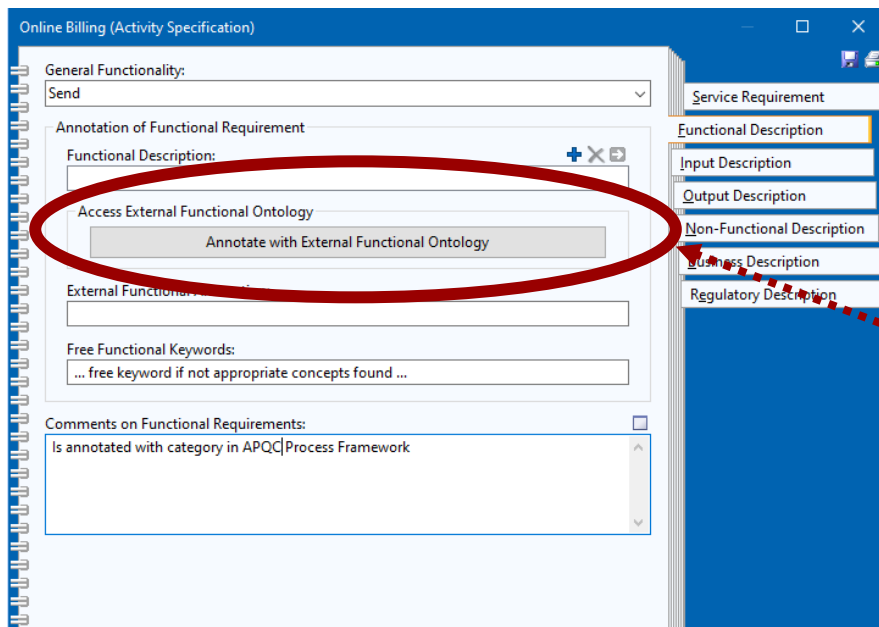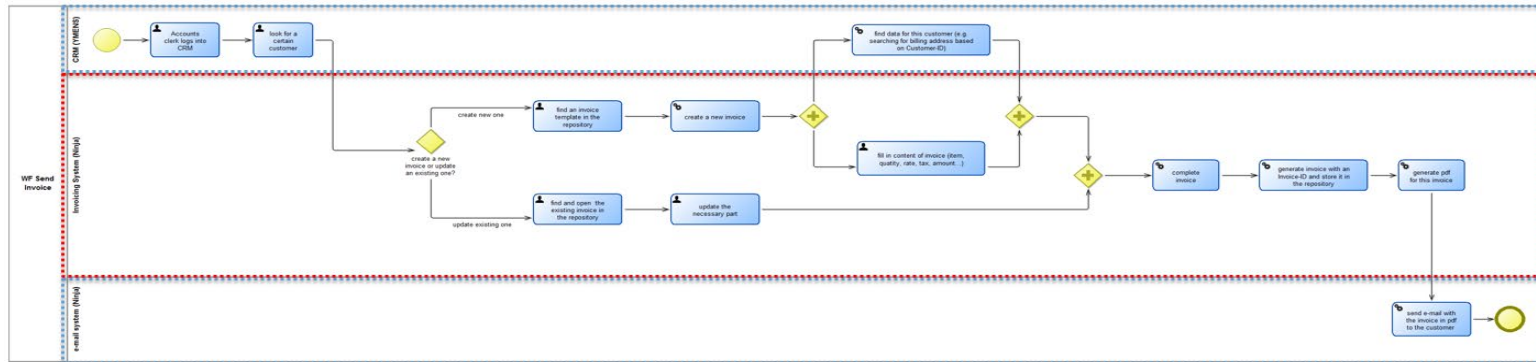   ▶ ● Develop Vision and Strategy
    ▶ ● Manage Customer Service
    ▶ ● Manage Enterprise Risk, Compliance, Remediation, and Resiliency
    ▶ ● Manage External Relationships
    ▼ ● Manage Financial Resources
      ▶ ● Manage fixed-asset project accounting
      ▶ ● Manage internal controls
      ▶ ● Manage international funds/consolidation
      ▶ ● Manage taxes
      ▶ ● Manage treasury operations
      ▶ ● Perform general accounting and reporting
      ▶ ● Perform global trade services
      ▶ ● Perform planning and management accounting
      ▼ ● Perform revenue accounting
        ▼ ● Invoice customer
          ● Generate customer billing data
          ● Maintain customer/product master files
          ● Post receivable entries
          ● Resolve customer billing inquiries
          ● Transmit billing data to customers
      ▶ ● Manage and process adjustments/deductions
      ▶ ● Manage and process collections
      ▶ ● Process accounts receivable (AR)
      ▶ ● Process customer credit
    ▶ ● Process accounts payable and expense reimbursements
    ▶ ● Process payroll
  ▶ ● Manage Information Technology (IT)
  ▶ ● Market and Sell Products and Services

Online Billing (Activity Specification)

General Functionality:
Send

Annotation of Functional Requirement

Functional Description:

Access External Functional Ontology

Annotate with External Functional Ontology

External Functional

Free Functional Keywords:
... free keyword if not appropriate concepts found ...

Comments on Functional Requirements:
Is annotated with category in APQC Process Framework

Service Requirement
Functional Description
Input Description
Output Description
Non-Functional Description
Business Description
Regulatory Description

Example

Functionality

Non-functional requirements

All Concepts are defined in the Ontology

Transformation and Mapping for Inferencing
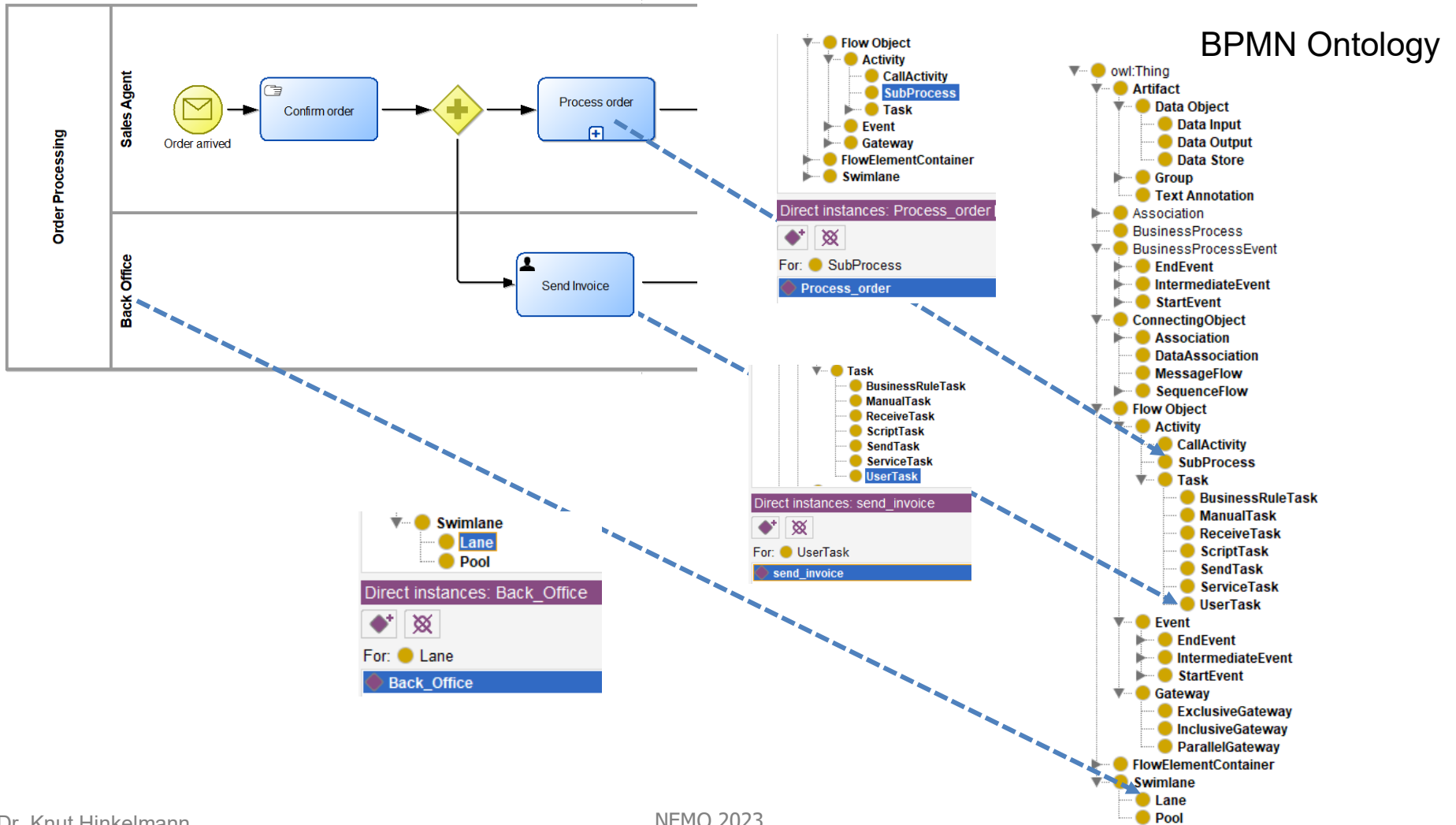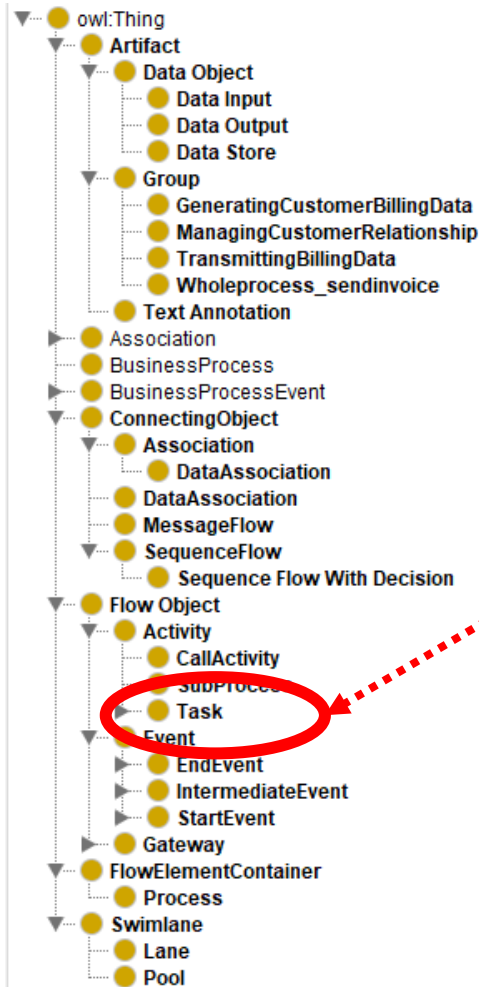

From: CoudSocket Project

Transformation and Mapping

The model elements are exported as instances of ontology classes

BPMN Ontology

Triple Store: Language and Domain Ontologies

BPMN Ontology:

Domain Ontology:
APQC Process Classification Framework
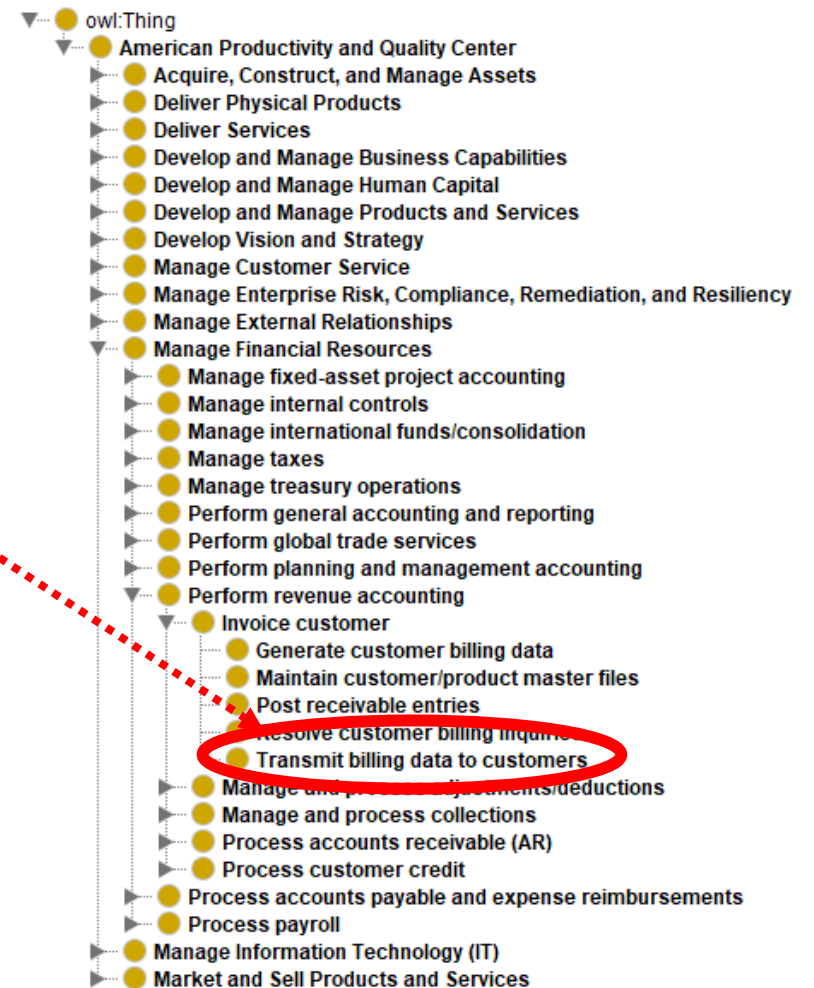
functionality

Inferencing: Cloud Service Selection

Cloud Service Selection

Functionality

Non-functional requirements

Thanks to Emanuele Laurenzi

355 Cloud Services

355 CSs from 4 Marketplaces–with13.098 specsvalues.
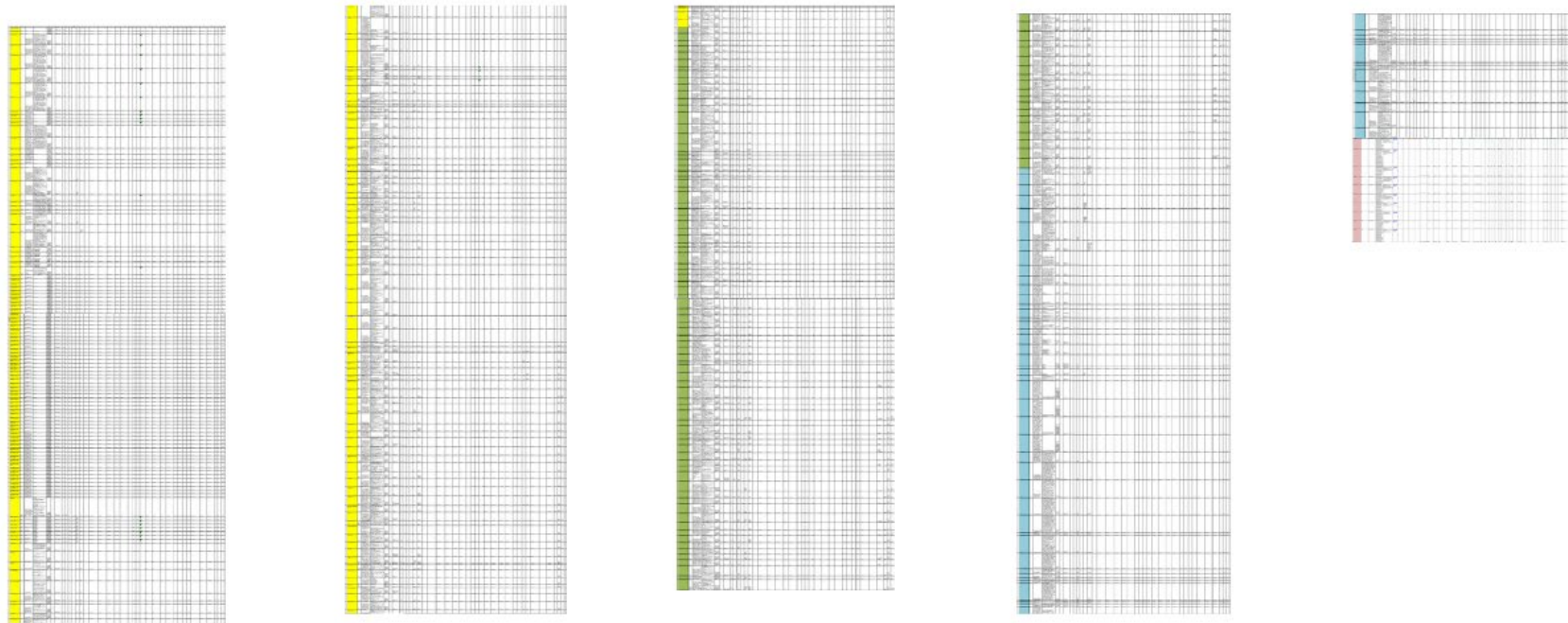
Inference
# *to enable the retrieval of suitable Cloud Services*

- Semantic rules (in SPARQL),

  - For Business-IT mapping

  - To turn implicit knowledge into explicit knowledge

- Example: If a service offers a backup retention time up to one year, implies also

  - Up to six months

  - Up to one month

  - Up to one week

  - Up to one day

```
[CONSTRUCT {
    ?cservice bpaas:cloudServiceHasBackupRetentionTime bpaas:up_to_6_months .
    ?cservice bpaas:cloudServiceHasBackupRetentionTime bpaas:up_to_1_month .
    ?cservice bpaas:cloudServiceHasBackupRetentionTime bpaas:up_to_1_week .
    ?cservice bpaas:cloudServiceHasBackupRetentionTime bpaas:up_to_1_day .
}
WHERE{
    ?cservice rdf:type bpaas:CloudService .
    ?cservice bpaas:cloudServiceHasBackupRetentionTime bpaas:up_to_1_year .
}]
```

Retrieve suitable Cloud Services for the given Business Process Model
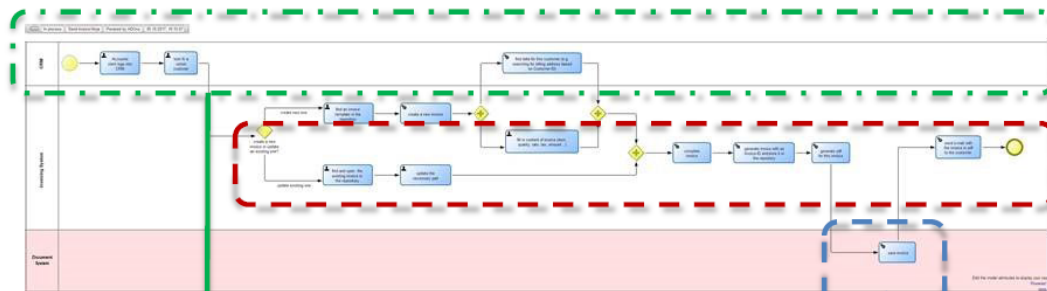
Invoice Management System

Specifications
Action: Generate
Object: Invoice
APQC: 9.2.2.2 Generate Customer Billing Data
Storage Location: Europe
Availability: 99.999%
N. simultaneus users: 100

Customer Management System Specifications
Action: Manage
Object: Customer
APQC: 3.5.2.4 Manage Customer Relationship
Storage Location: Europe
Availability: 99.999%

Action: Manage
Object: Document
APQC: 9.2.2.2 Generate Customer Billing Data
Storage Type: Storage Service
Storage Location: Europa
**Availability:** 99.999%

Drawbacks of Semantic Lifting

- **Separate Environments**

  - ♦ Modelling and Metamodelling

  - ♦ Ontology

- **Inconsistency**

  - ♦ Metamodel and ontology must represent the same semantics but are maintained independently

  - ♦ Each change in metamodel must be reproduced in the ontology and vice versa

- **Effort**

  - ♦ After each change the models must be translated again into the ontology instances
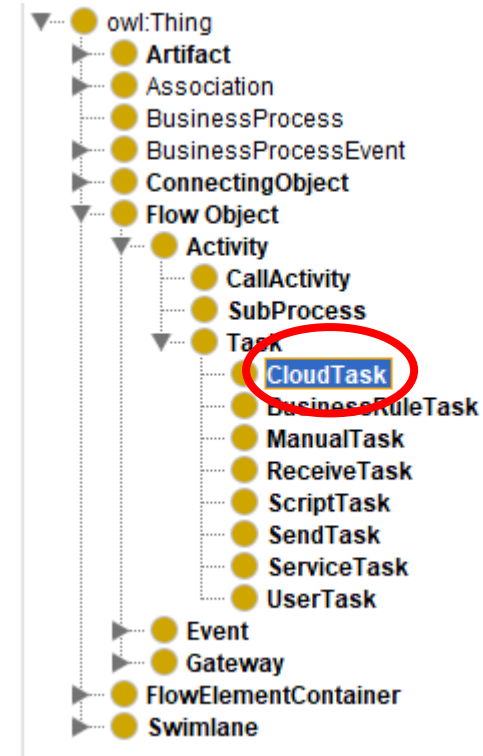
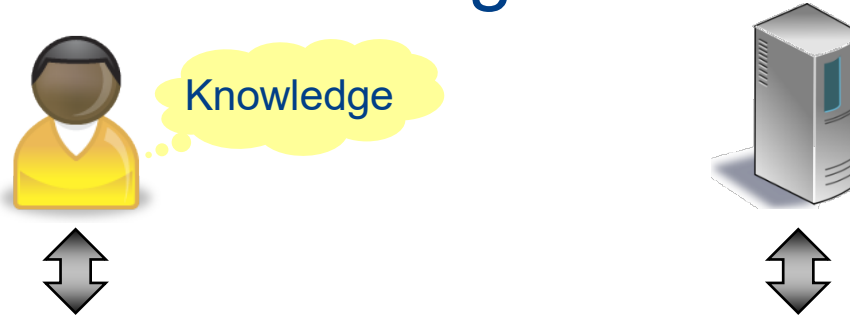# *Example: New Model Element*

■ New task type: Cloud Task
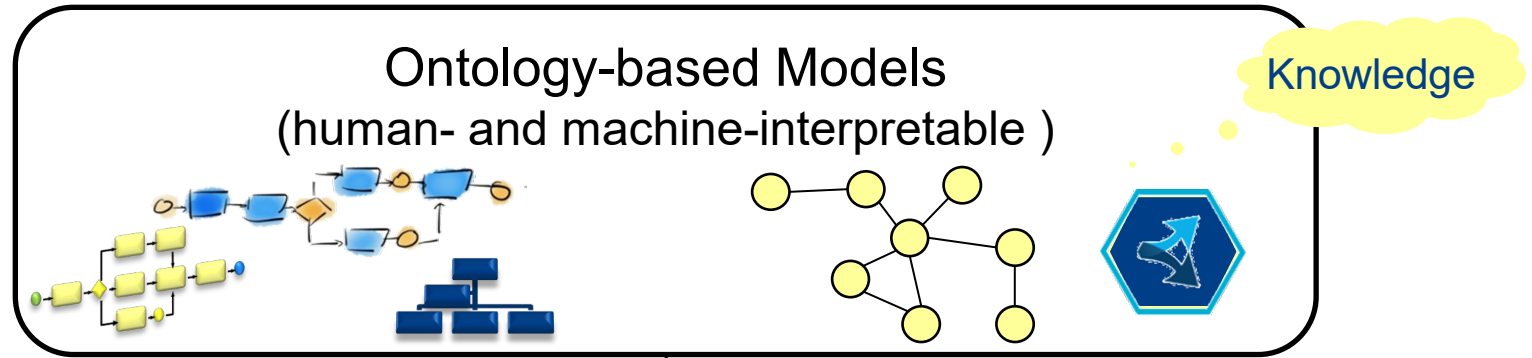
Change in the meta model:
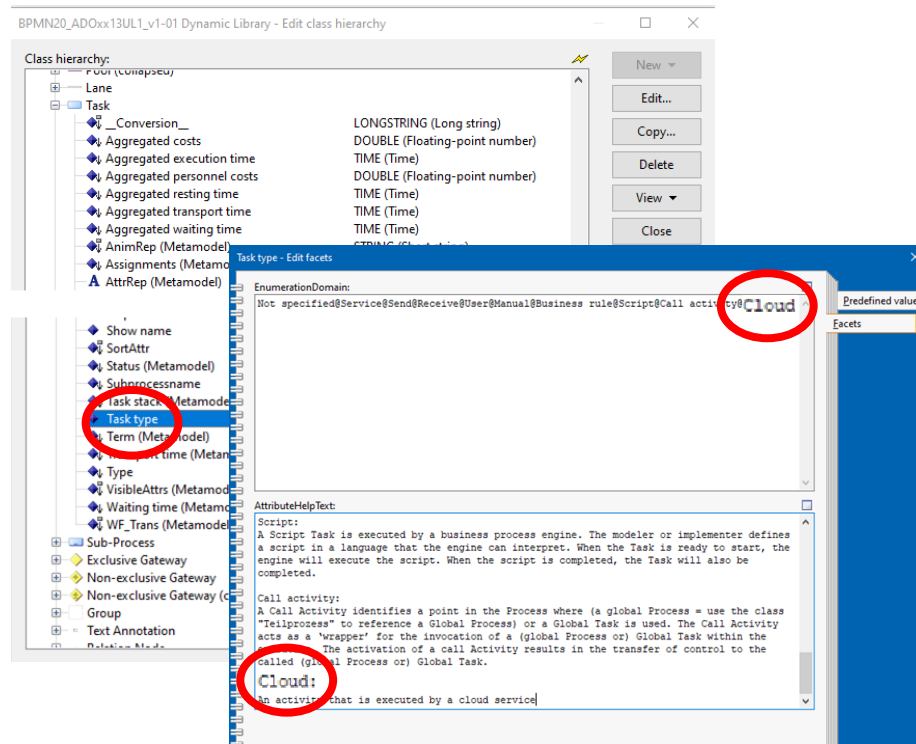
Change in the ontology:

Ontology-based Metamodeling
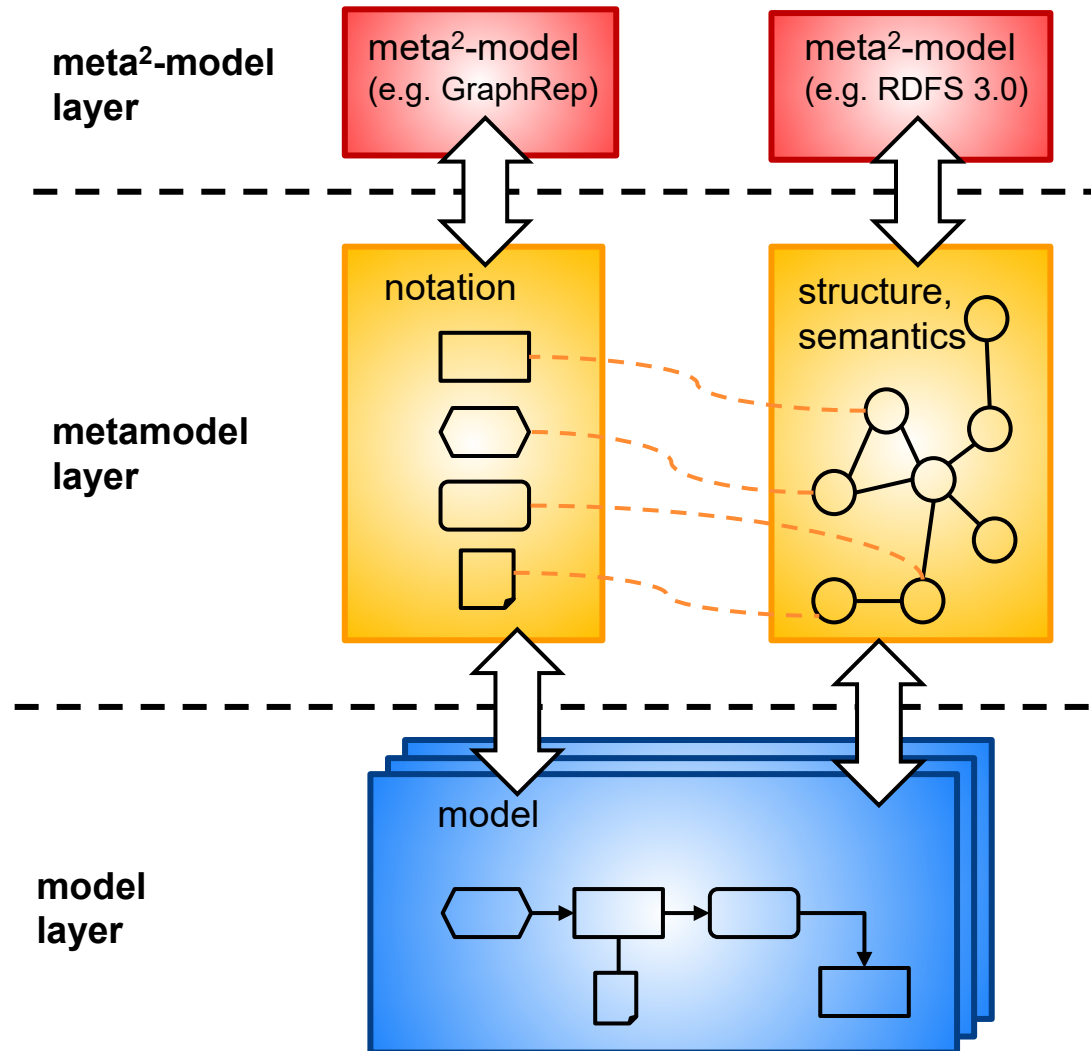
Ontology-based Metamodeling

Knowledge

Knowledge

Knowledge

*Models +
Knowledge*

Ontology-based Models
(human- and machine-interpretable)

Reality

(Laurenzi et al. 2018)

Ontology-based Metamodeling

meta^2-model layer

meta^2-model (e.g. GraphRep)

meta^2-model (e.g. RDFS 3.0)

metamodel layer

notation

structure, semantics

model layer

model

AOAME – A Tool for Ontology-Based Modelling

- Single environment for modelling and ontology
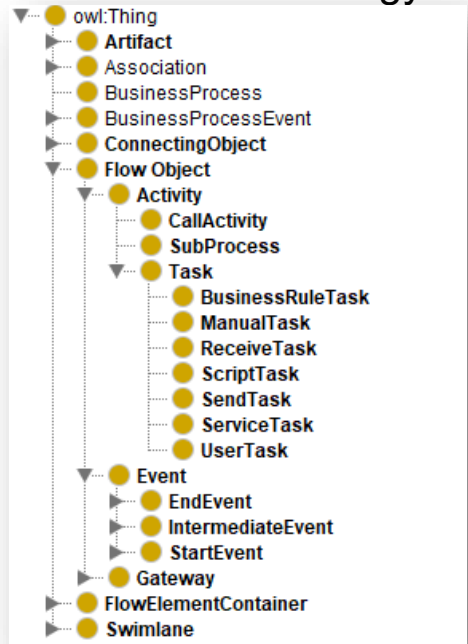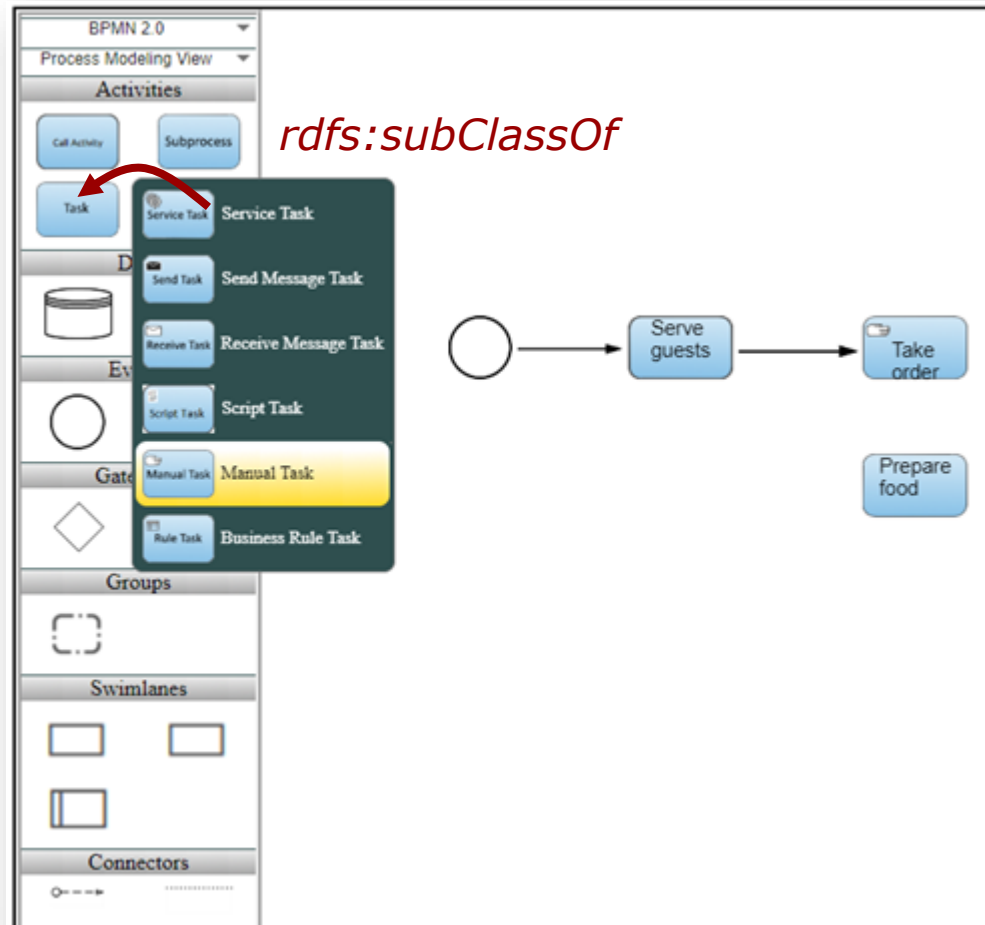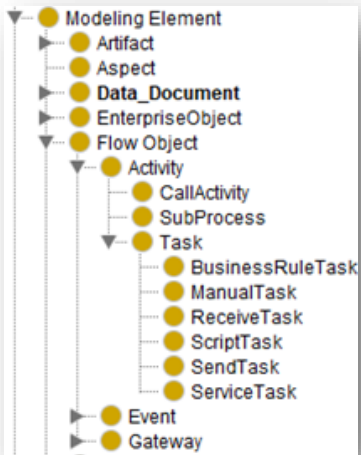- Model elements are directly created as instances in the ontology
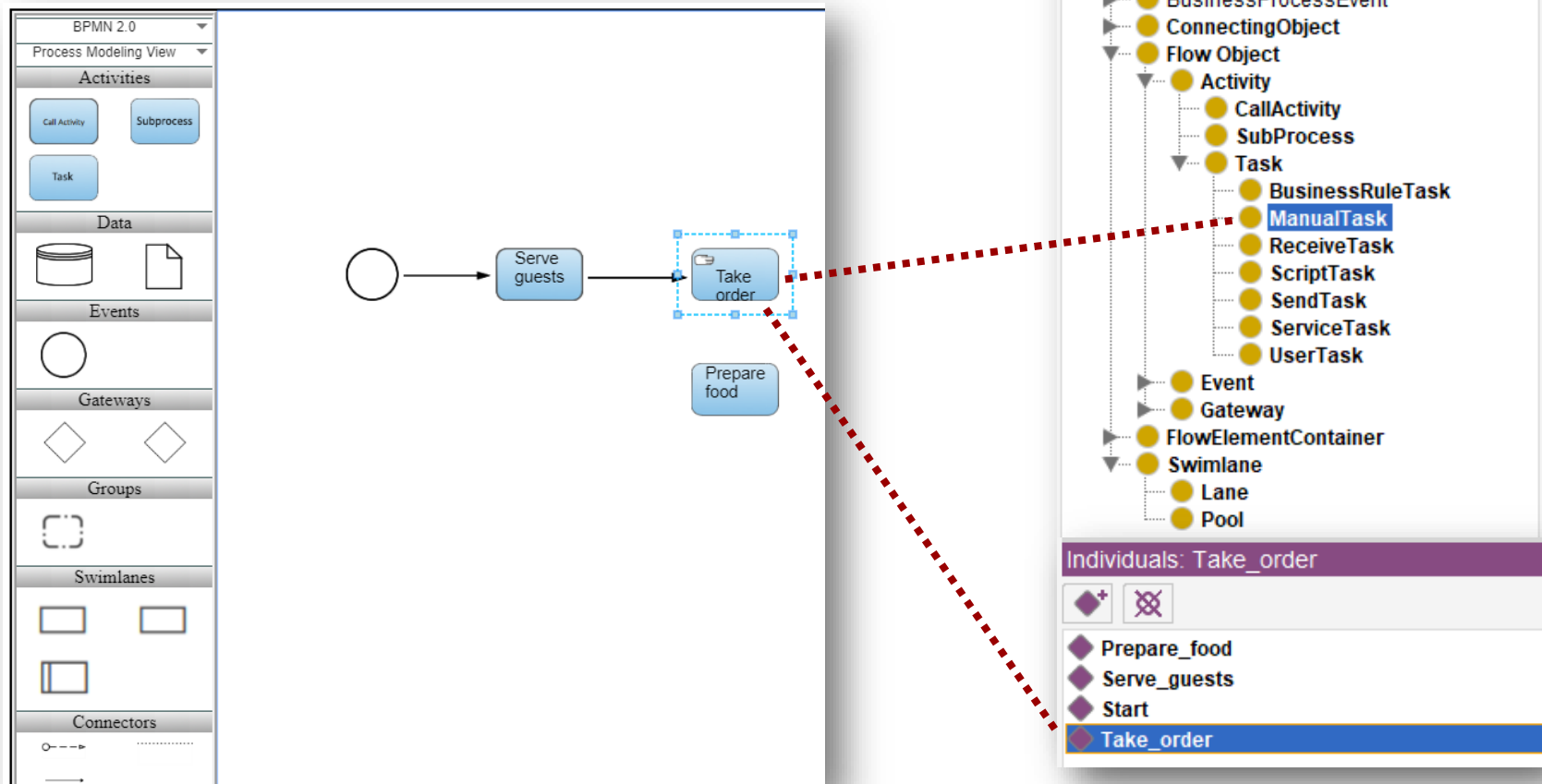
Ontology-Based Modeling in AOAME

Palette

Model Editor

BPMN
Meta model Ontology

Palette Ontology
(excerpt)

rdfs:subClassOf

Example Query

«Which task elements are in the model Serve Guests»?

```
PREFIX owl: <http://www.w3.org/2002/07/owl#>
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX mod: <http://fhnw.ch/modelingEnvironment/ModelOntology#>
PREFIX lo: <http://fhnw.ch/modelingEnvironment/LanguageOntology#>
PREFIX po: <http://fhnw.ch/modelingEnvironment/PaletteOntology#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX bpmn: <http://ikm-group.ch/archiMEO/BPMN#>


SELECT ?model ?shape ?task ?l
WHERE {
    ?model rdfs:label "Serve Guests".
    ?model mod:modelHasShape ?shape.
    ?shape mod:shapeVisualisesConceptualElement ?task.
    ?task rdf:type bpmn:Task .
    ?shape rdfs:label ?l.
}
```
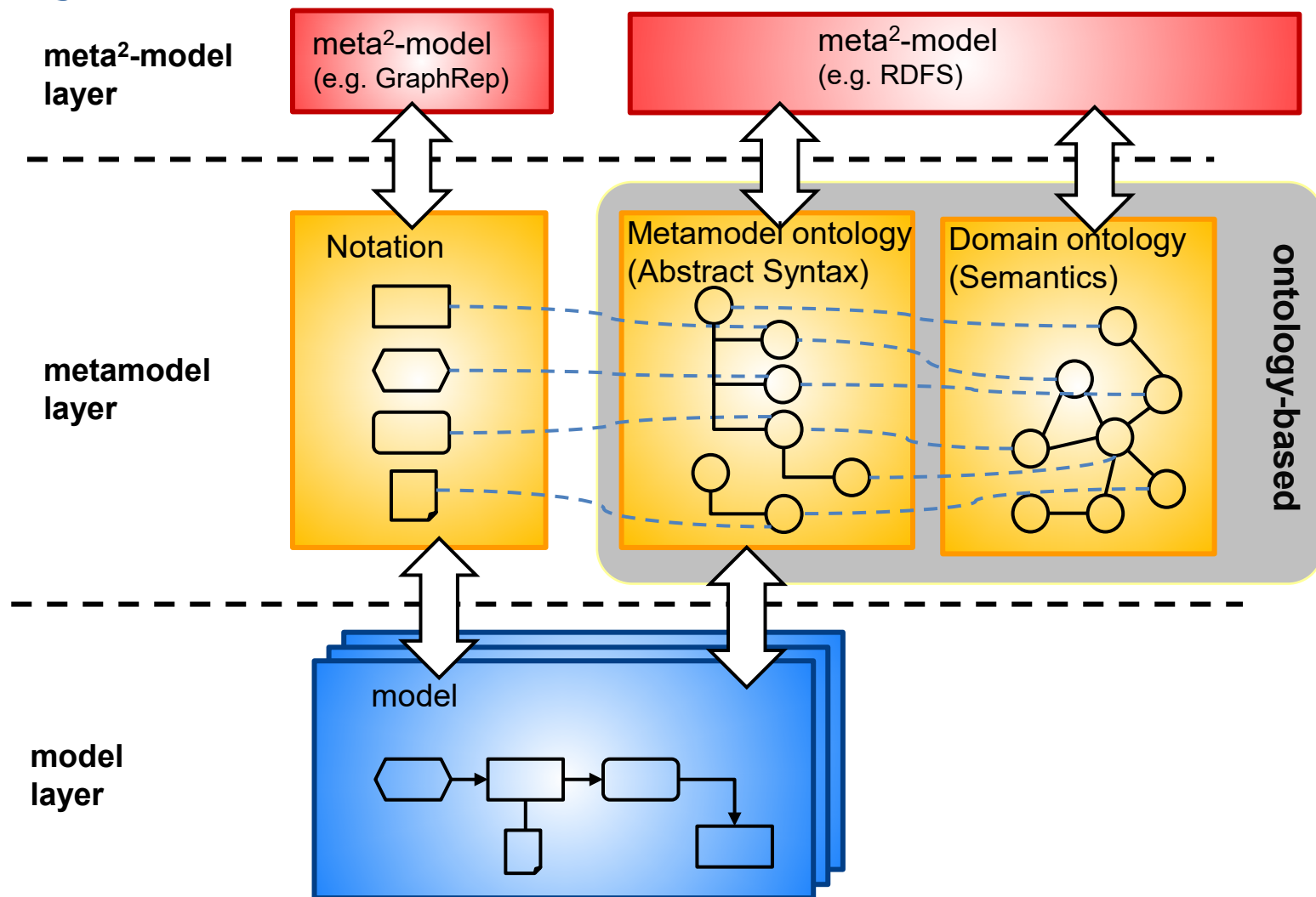
Select the elements (named shapes) in the model

For the shapes find the conceptual elements

Filter the elements for BPMN Tasks and show the labels

Ontology-based Metamodeling (2): Ontologies for Metamodel and Content

meta²-model layer

meta²-model (e.g. GraphRep)

meta²-model (e.g. RDFS)

metamodel layer

Notation

Metamodel ontology (Abstract Syntax)

Domain ontology (Semantics)

ontology-based

model layer

model

Representing Complete Content as Ontology

■ **Meta model Ontology**:

    ♦ Concepts of the meta model are classes in an ontology

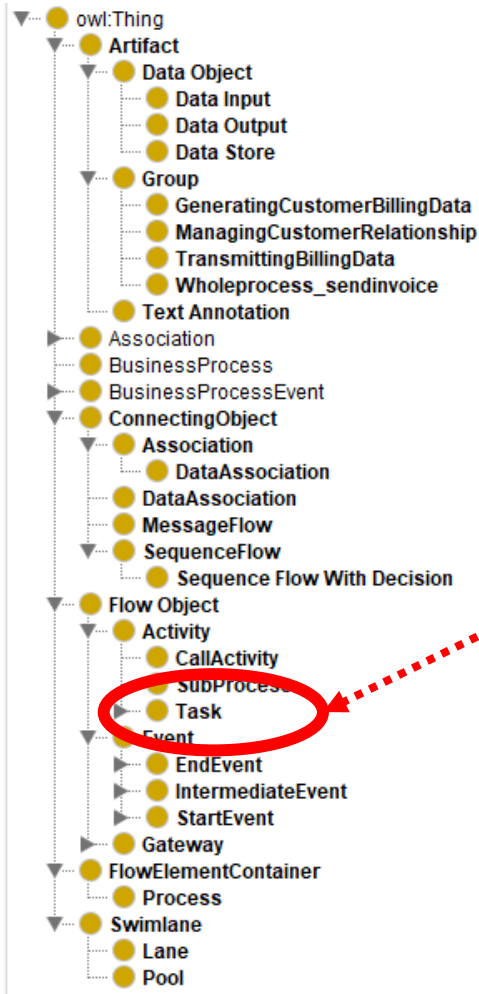   ♦ Modelling = creating instances of classes

■ **Application Domain Ontology**:

    ♦ Model elements are annotated with domain knowledge
      from application domain ontology

ERP

Send Invoice

■ Ontology reasoning can be applied to the complete content
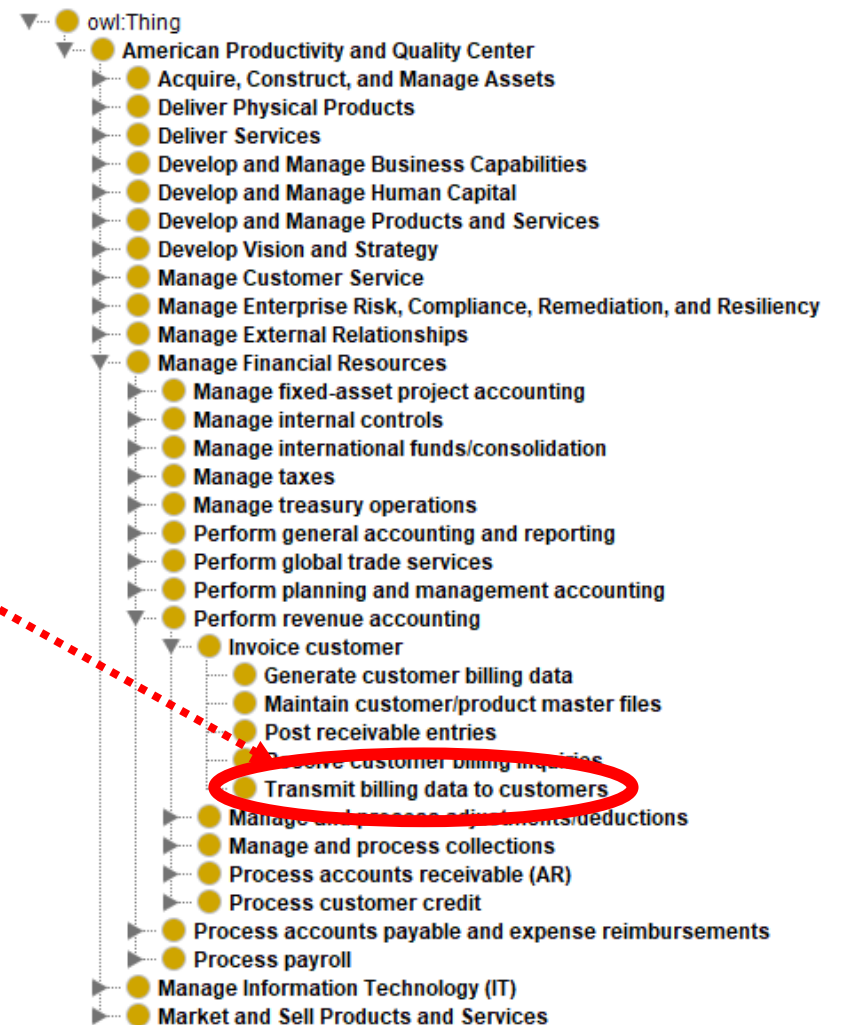knowledge in the models

Language and Domain Ontologies
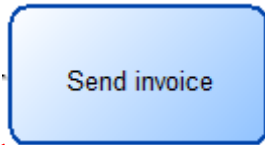
BPMN Ontology:

Domain Ontology:
APQC Process Classification Framework

functionality

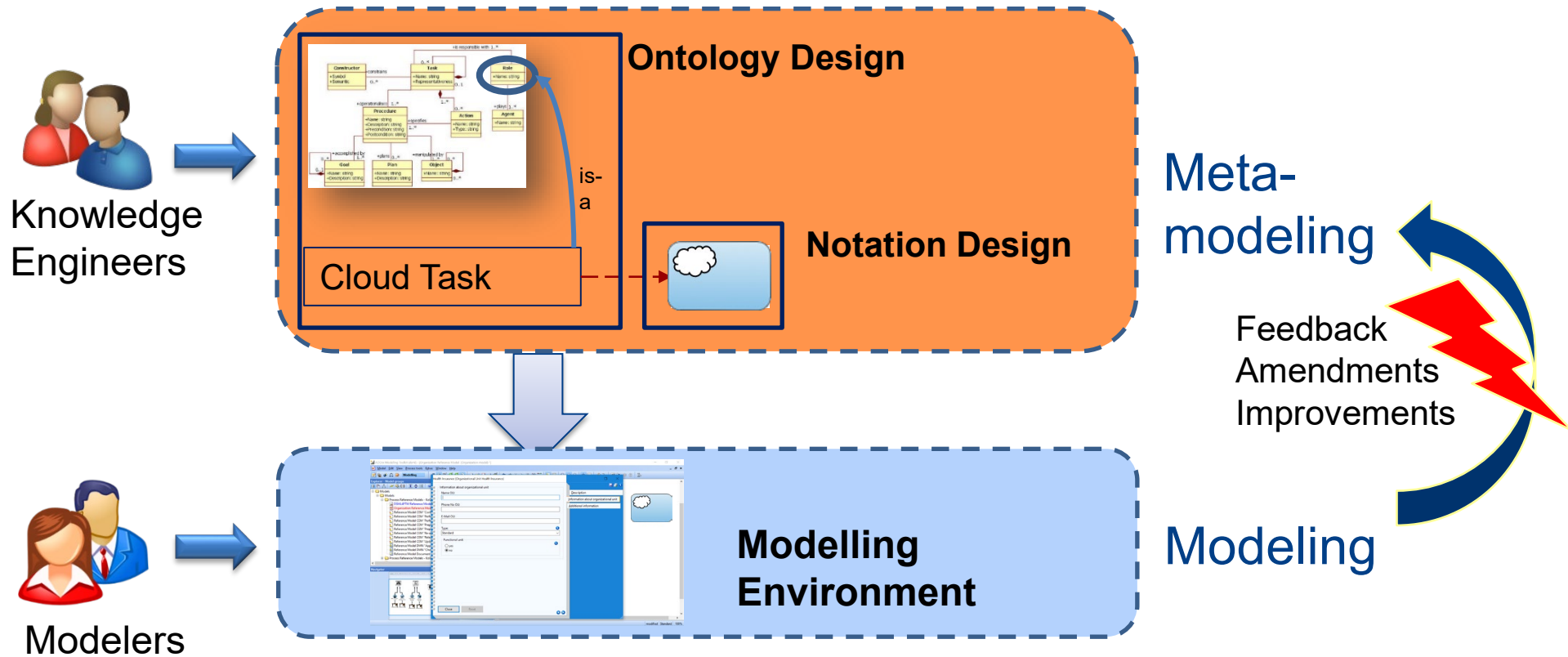Send invoice

Computer needs up-to-date Knowledge, too

THE COMPUTER NEEDS TO UPGRADE ITS ONTOLOGY
TO BE COMPATIBLE WITH THE NEW MODEL

Agile Modeling and Meta-Modeling

Objective

Ensure a precise shared interpretation of new modeling constructs to both **humans and machines**

Change of Metamodel

Knowledge Engineers

Ontology Design

Cloud Task

Notation Design

is-a

Meta-modeling

Feedback
Amendments
Improvements

Modelers

Modelling Environment

Modeling

Time-consuming engineering effort!

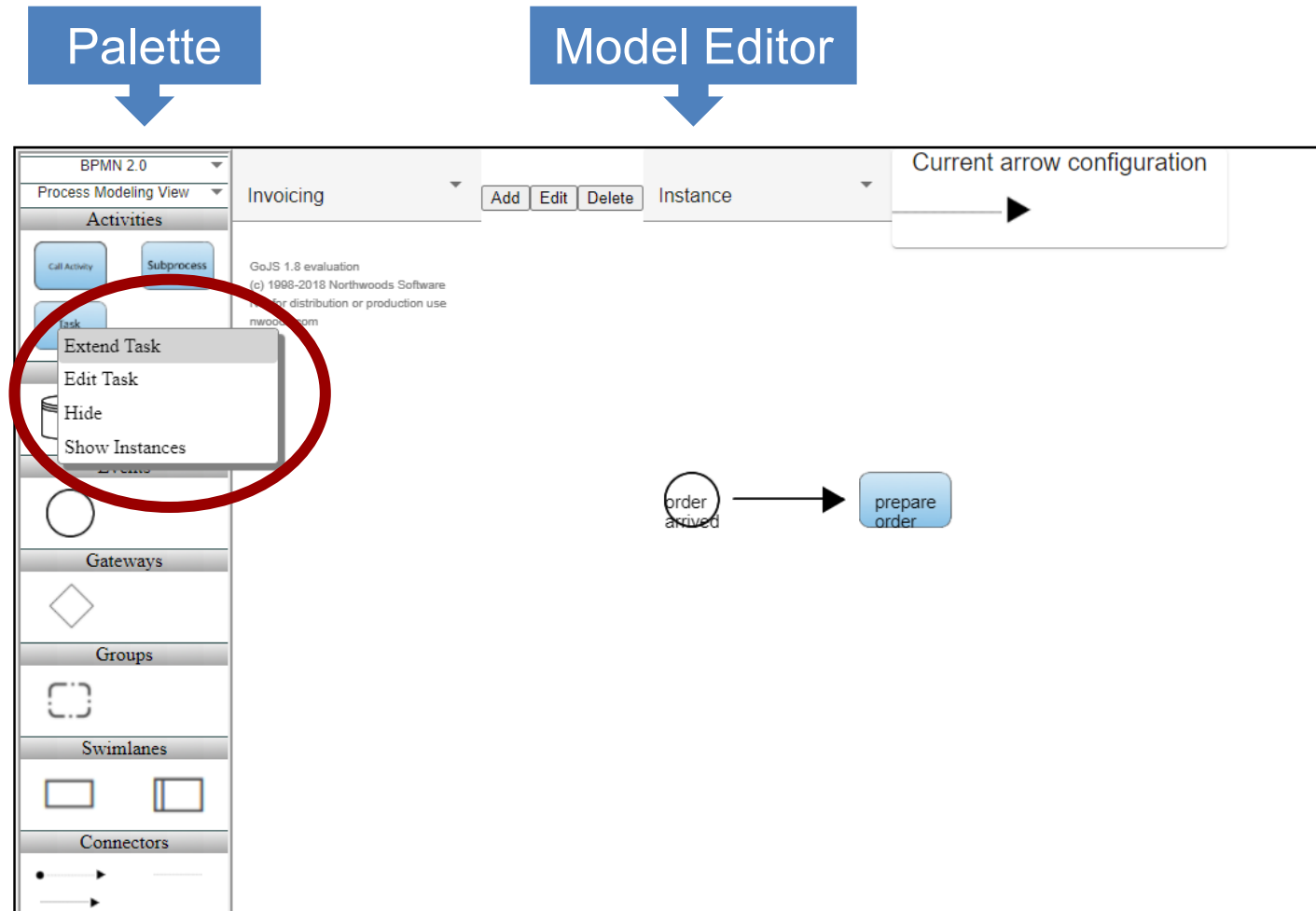# *Integration Modeling and Metamodeling in a Single Environment*

- Tight collaboration between metamodel developer and modeler
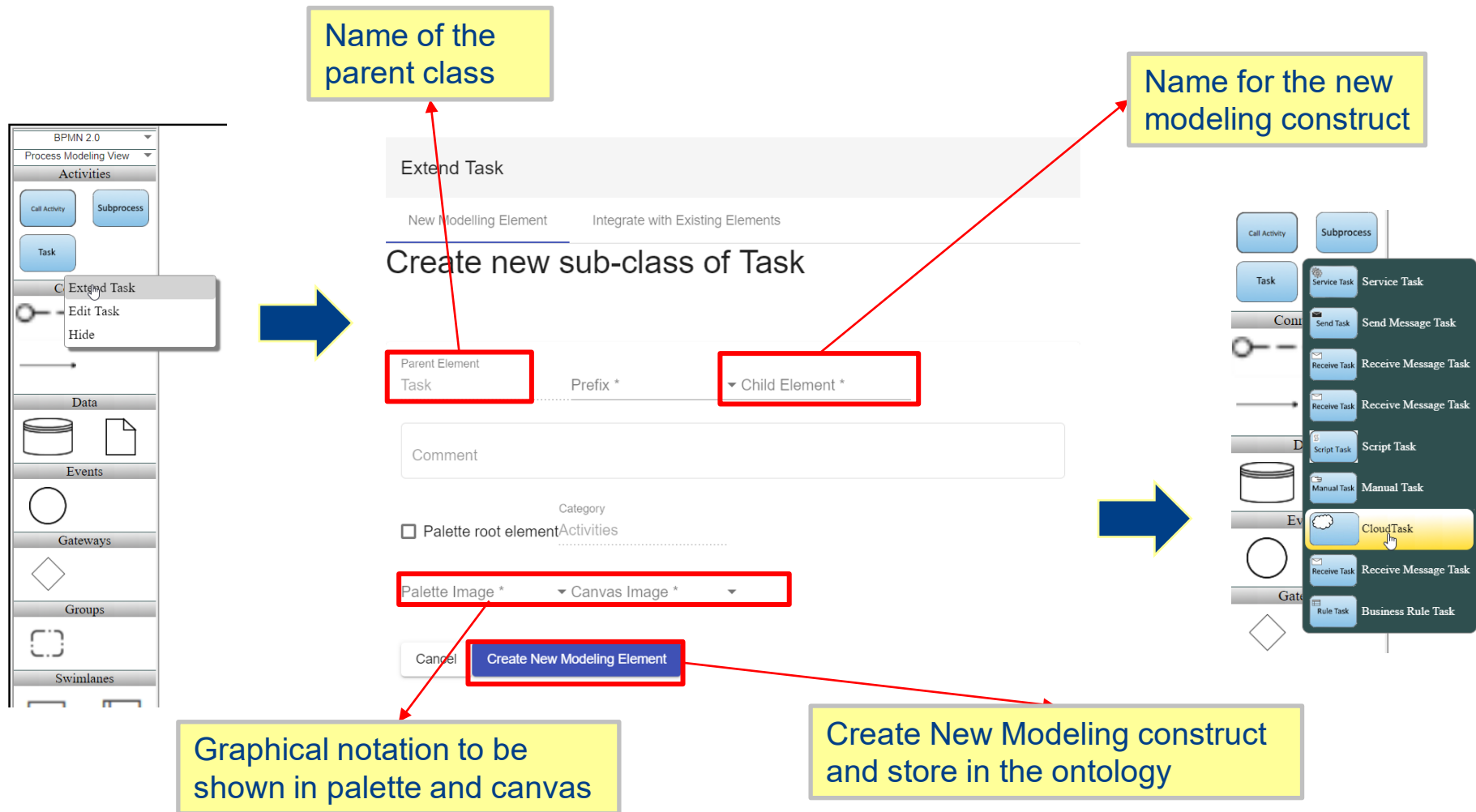- Modeler can also take the role of metamodel developer
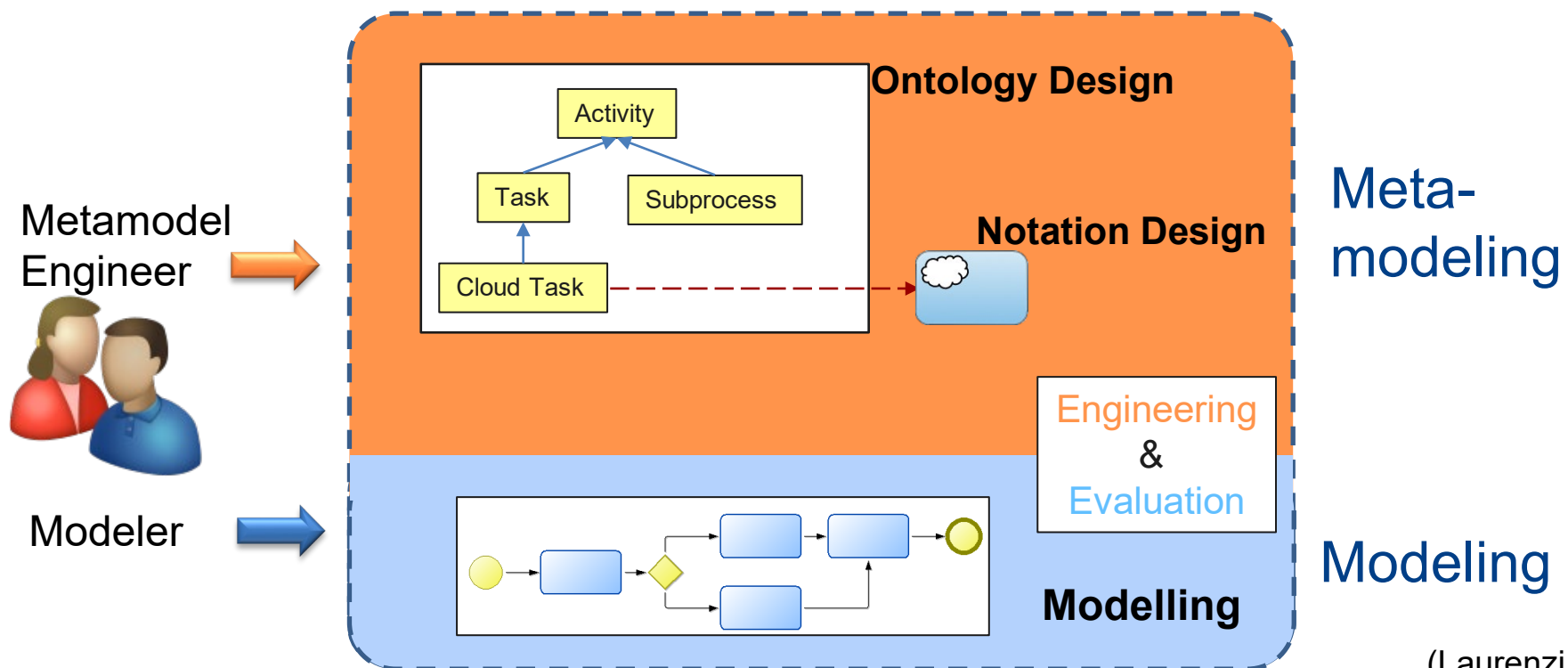
(Laurenzi et al. 2018)

Extending AOAME Modeling Languages – on the fly

Integration of Meta-modeling and Modeling: On-the-fly Modeling Language Adaptation

Name of the parent class

Name for the new modeling construct

Graphical notation to be shown in palette and canvas

Create New Modeling construct and store in the ontology

Semantic Alignment in AOAME

■ With Semantic Mapping modeling elements can be connected to domain ontology

The Cool Thing

AOAME is a graphical ontology development tool

- Start with an empty ontology: Just a root node

- Step by step expand the ontology

- Result
  - ♦ Domain ontology
  - ♦ Graphical representation of classes

➢ *Can be done by business people*
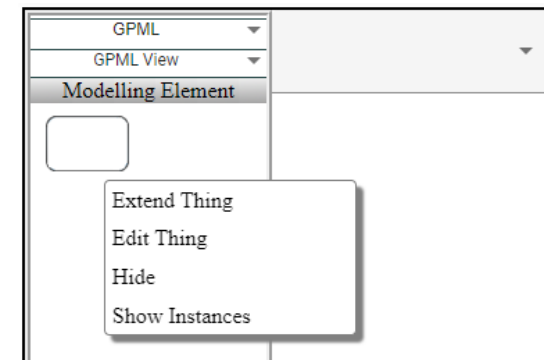
AOAME:
Agile and Ontology-Aided Modeling Environment

■ AOAME is a a prototypical implementation for Agile and Ontology-Aided Modeling

■ It is based on the PhD Thesis of Emanuele Laurenzi

■ Implementation of the current version by

   ♦ Emanuele Laurenzi

   ♦ Charuta Pande

   ♦ Devid Montecchiari

   ♦ Egemen Kaba

   ♦ Marco Di Ianni

   ♦ Jan Eich

The End

That's all Folks!