

Enterprise COllaboration & INteroperability



COIN Workshop Semantic Interoperability Services

Budapest, 05.05.2009
Francesco Taglino, Michele Missikoff
LEKS, IASI-CNR (ITALY)



Outline

- Information interoperability services
 - Information interoperability problem
 - The Athena semantic reconciliation approach and suite
 - The COIN innovative services
- Knowledge Interoperability services
 - Objectives
 - Beyond the State of the Art
 - Semantic Supporting Services (SSS)
 - Knowledge Interoperability Services (KIS)
- Conclusions



Introduction

Two kinds of semantic interoperability services for

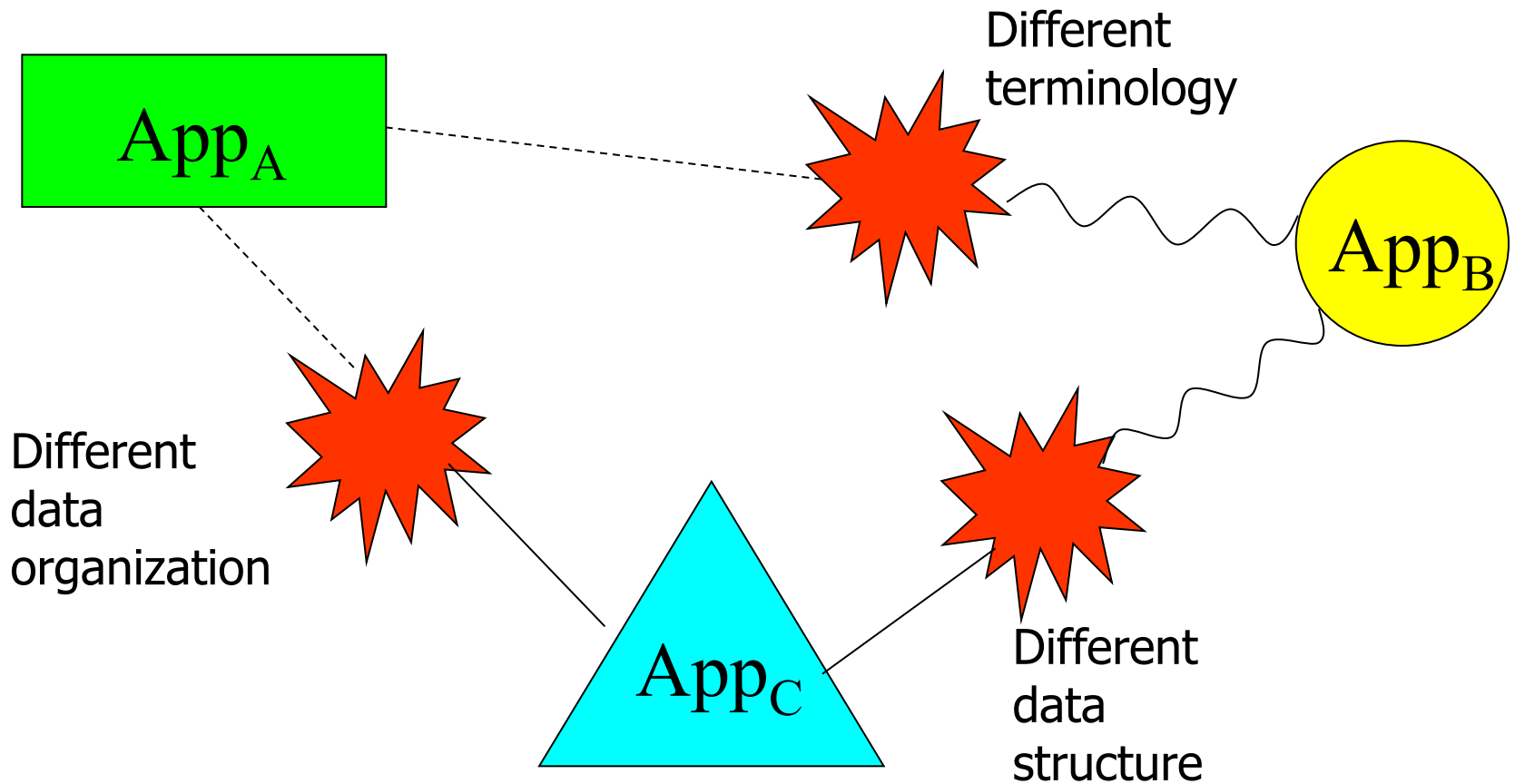
- Reconciliation of business documents (**information interoperability**)
- Management of enterprise competencies and skills (**knowledge interoperability**)



Information interoperability services

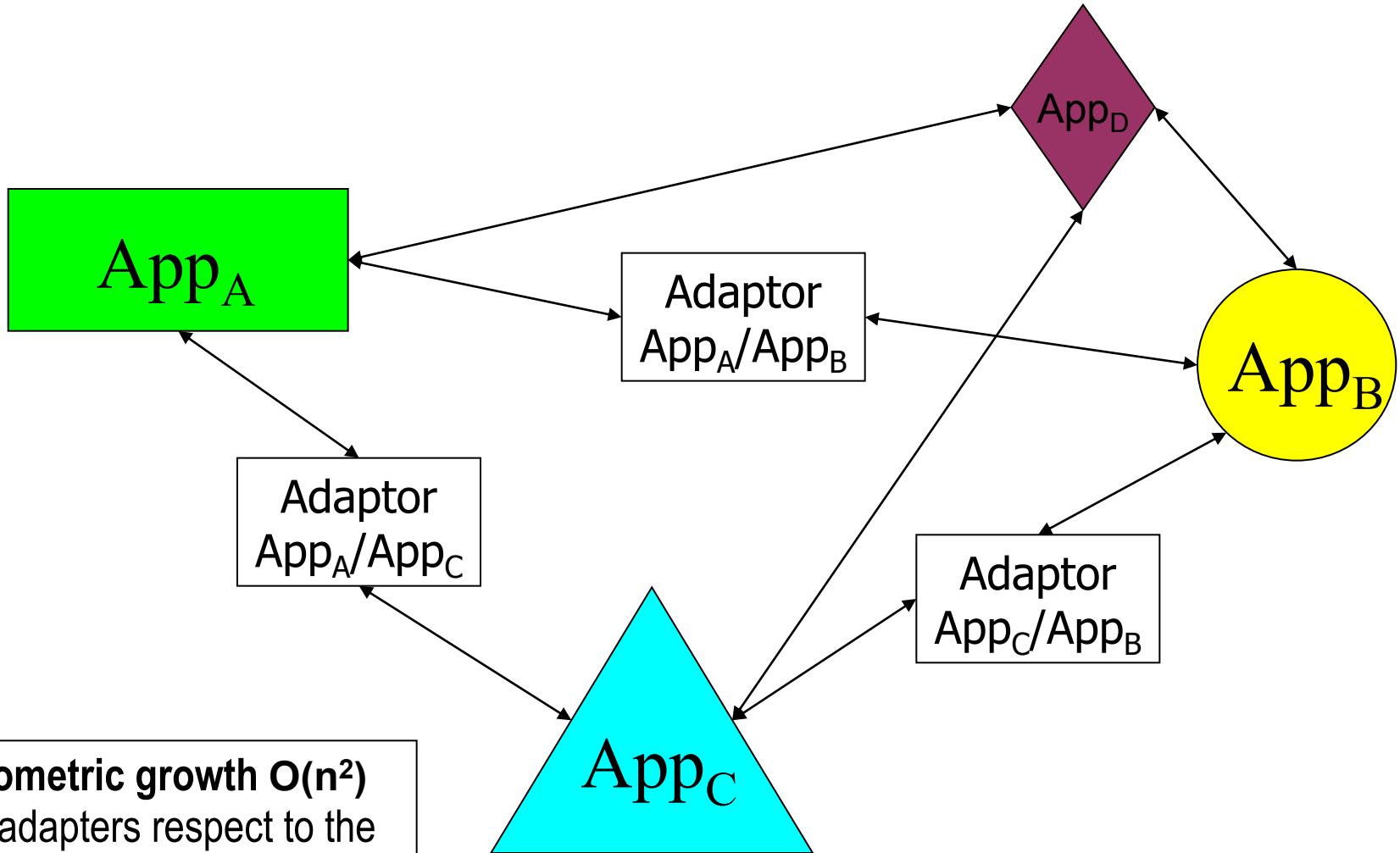


Information interoperability problem





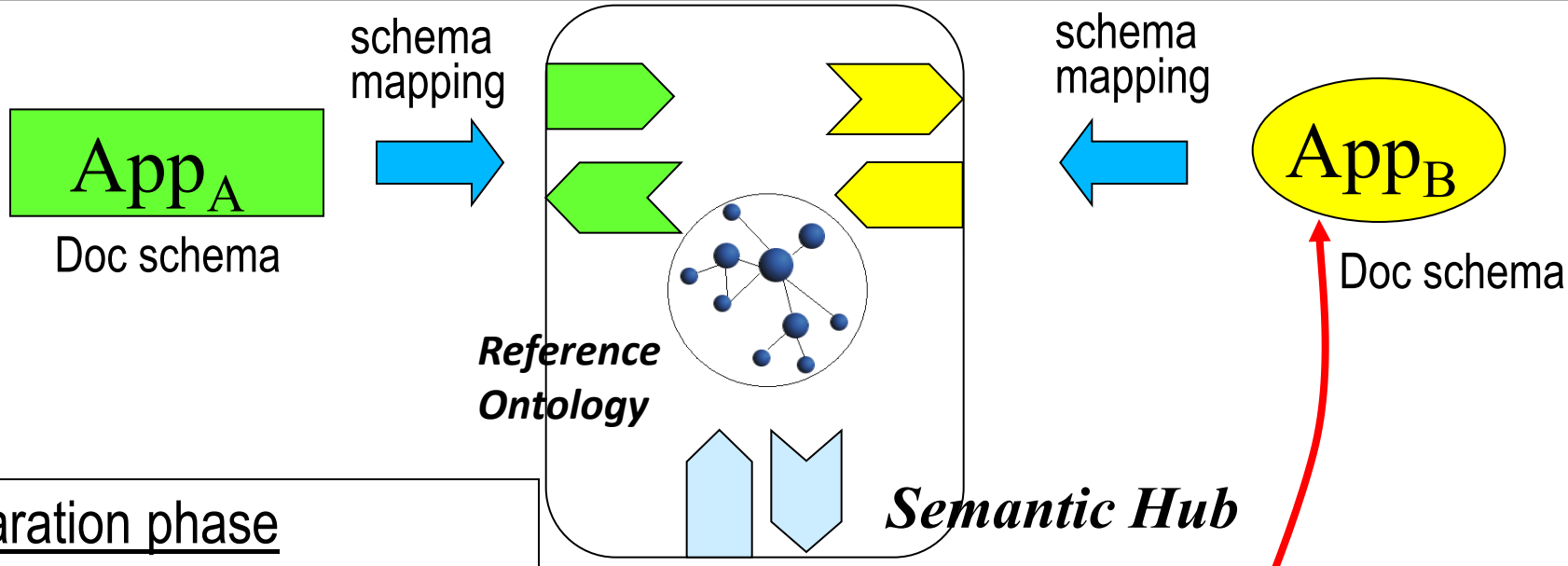
The EAI state-of-the-Art solution



- **Geometric growth $O(n^2)$** of the adapters respect to the number of parties
- **Maintenance issues**



The Athena reconciliation approach



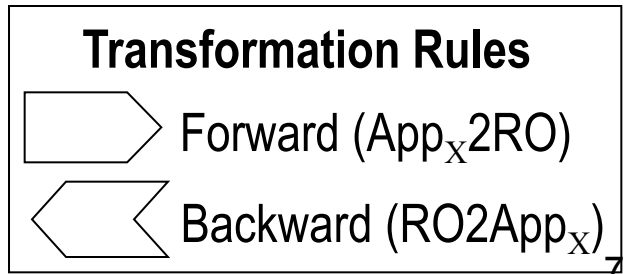
Preparation phase

- **Mapping** = Semantic Annotation + Transf rules
- Mapping expressed in terms of a **RO**

Two rulesets for each SA

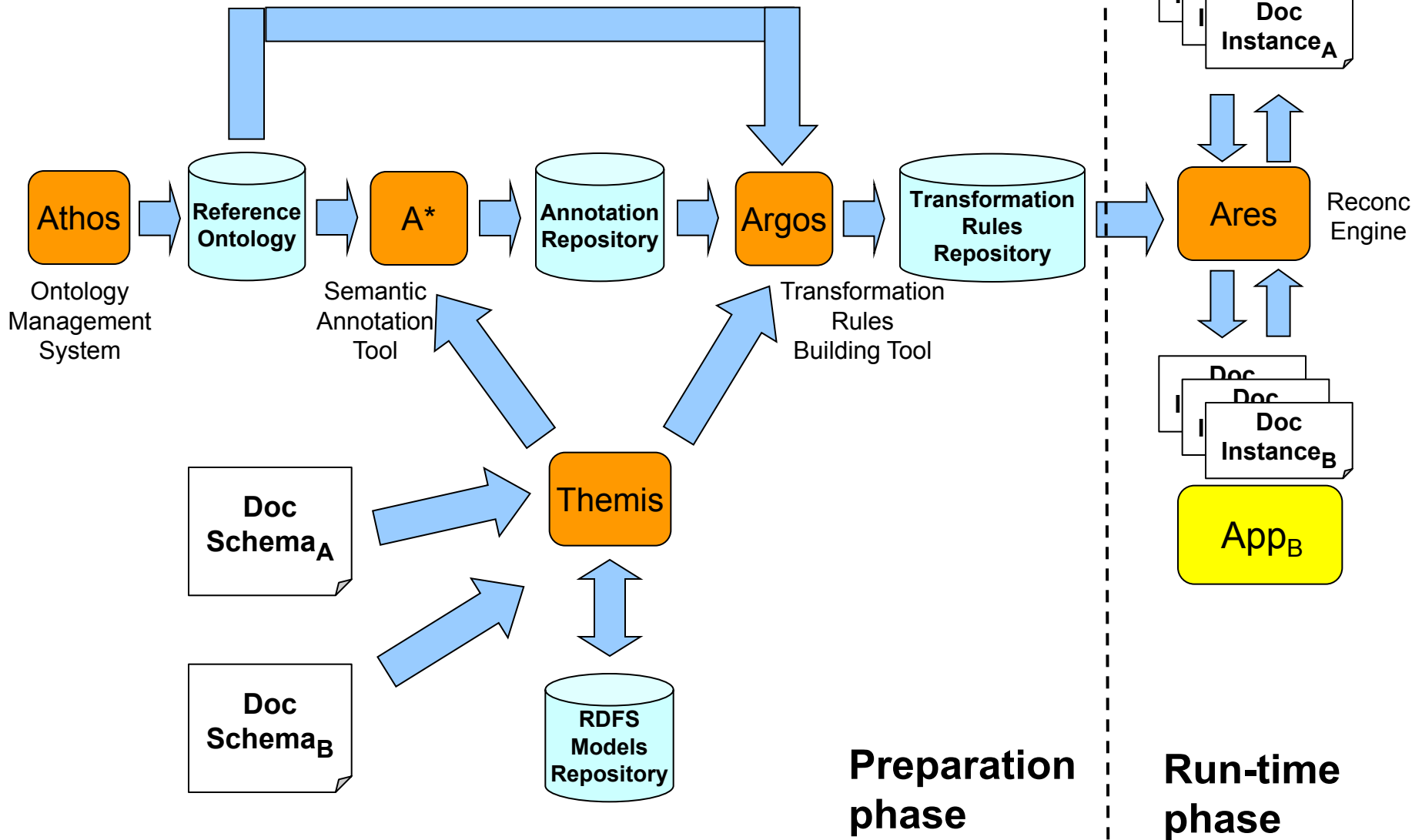
Run time phase

- **Reconciliation** through pipelining application of rules





Semantic Reconciliation suite





Semantic mismatch patterns

Lossless mismatches

can be solved without loss of information

- **Naming**: different labels for the same content
- **Attribute granularity**: the same information is splitted in a different number of attributes
- **Structure Organization**: different structures and organization of the same content
- **Subclass-Attribute**: an attribute with predefined value set is represented by a set of subclasses
- **Schema-Instance**: data hold schema information
- **Encoding**: different format of data or unit of measure

Lossy mismatches

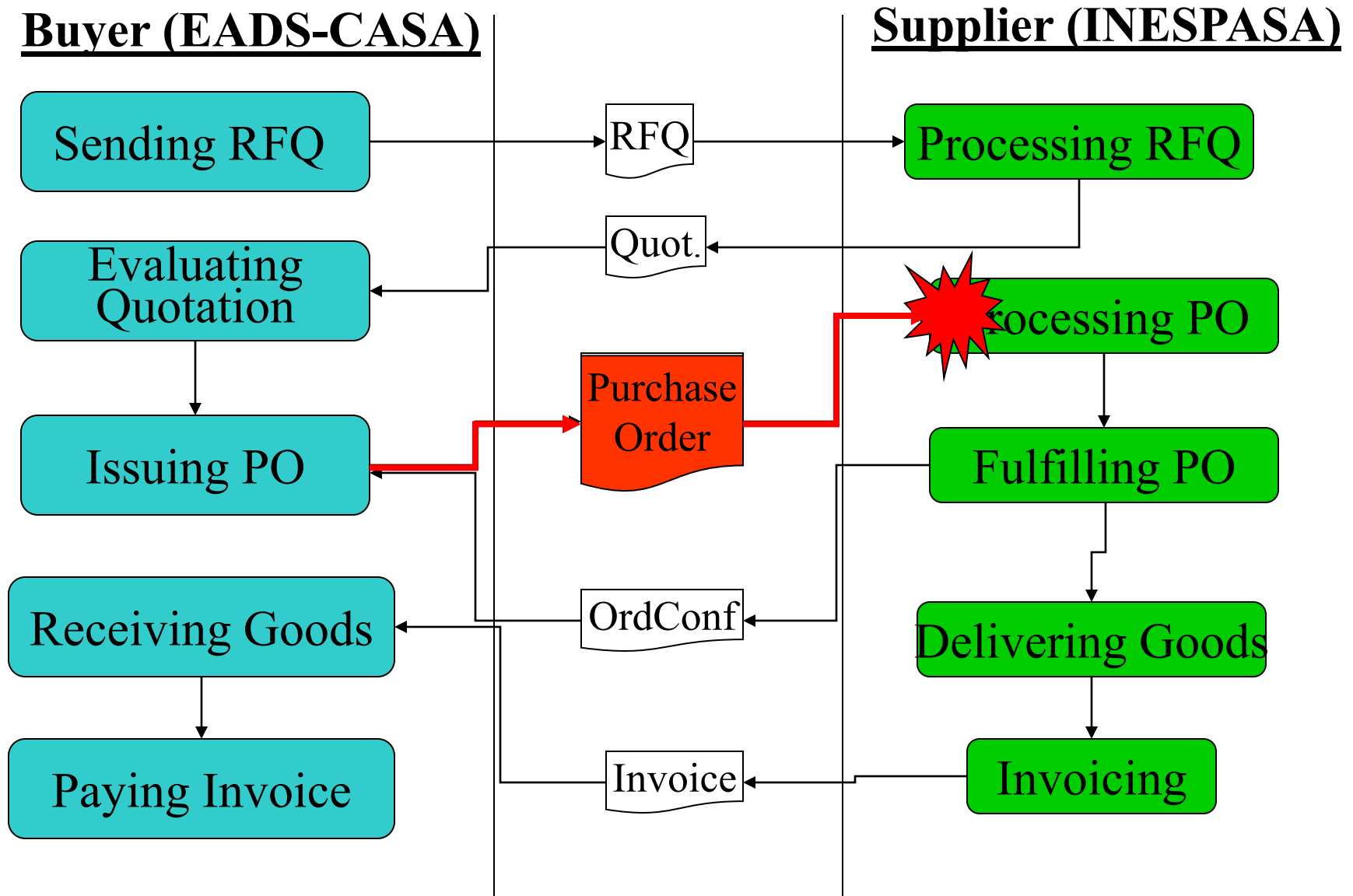
any transformation can cause a loss of information

- **Content**: different content denoted by the same concept (typically expressed by enumeration)
- **Coverage**: the presence/absence of information
- **Precision**: the accuracy of information
- **Abstraction**: level of specialisation refinement of the information



E-procurement scenario

(inspired by the ISOIN scenario)





Ontology for interoperability



The Reference Ontology

- The first key element of the proposed approach
- Used to **describe the domain** of interest and to give meaning to the structure of business documents to be reconciled
- Based on definition of **concepts and relationships (Structuring Ontology)**
- The construction of the ontology is supported by the **Athos** tool, based on the **OPAL** (Object, Process and Actor) ontological framework



OPAL meta-concepts

Primary

- **Actor_kind:** aimed at modelling any relevant entity of the domain that is able to activate or perform a process (e.g., Company, ...)
- **Object_kind:** aimed at modelling a passive entity, on which a process operates, typically to modify its state (e.g., RFQ, Invoice, ...)
- **Process_kind:** aimed at modelling an activity that is performed by an actor (e.g., Sending Invoice, ...)

Complementary

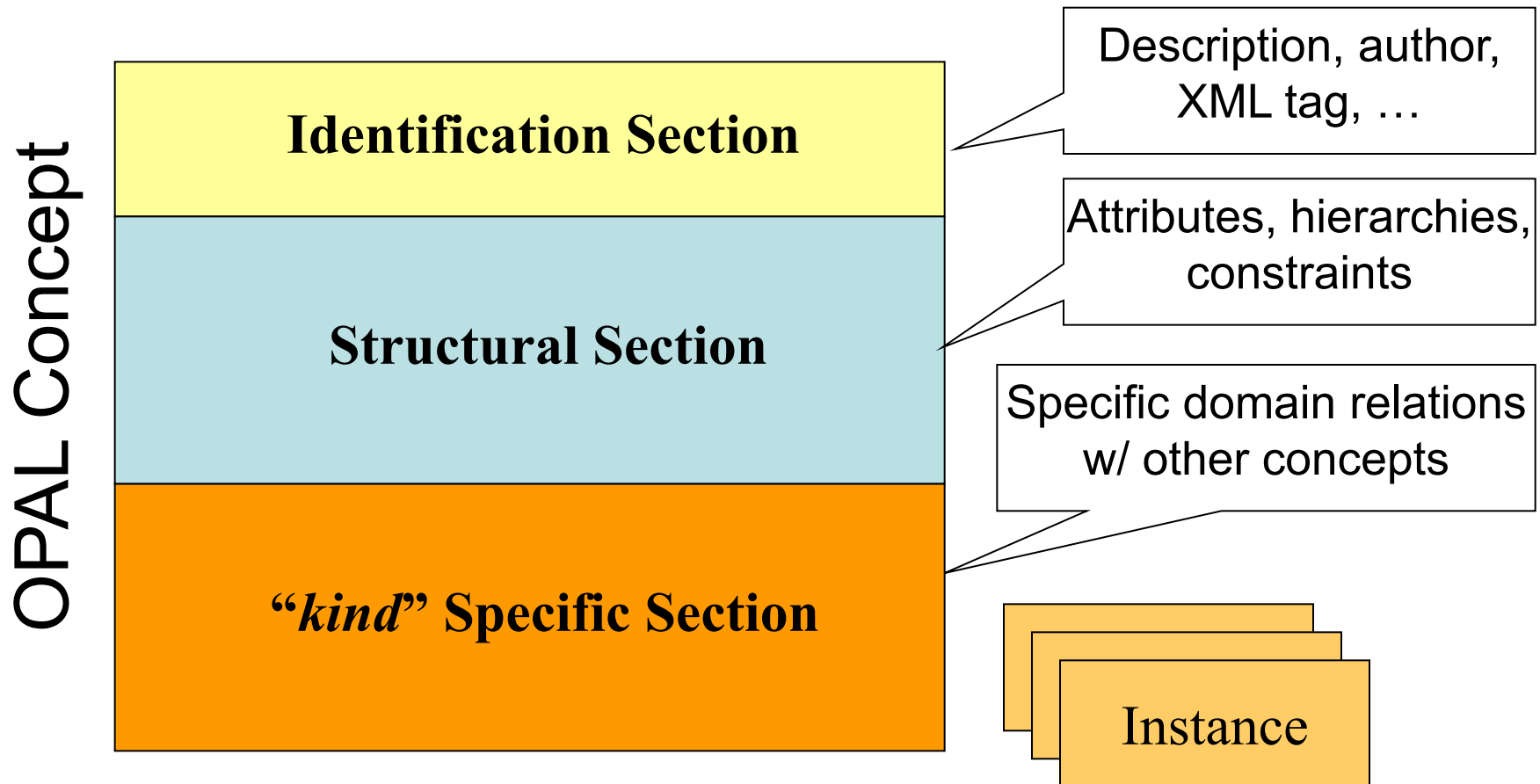
- *Atomic Attribute, Complex Attribute, Message, BOD*

The OPAL concepts are connected by conceptual relations

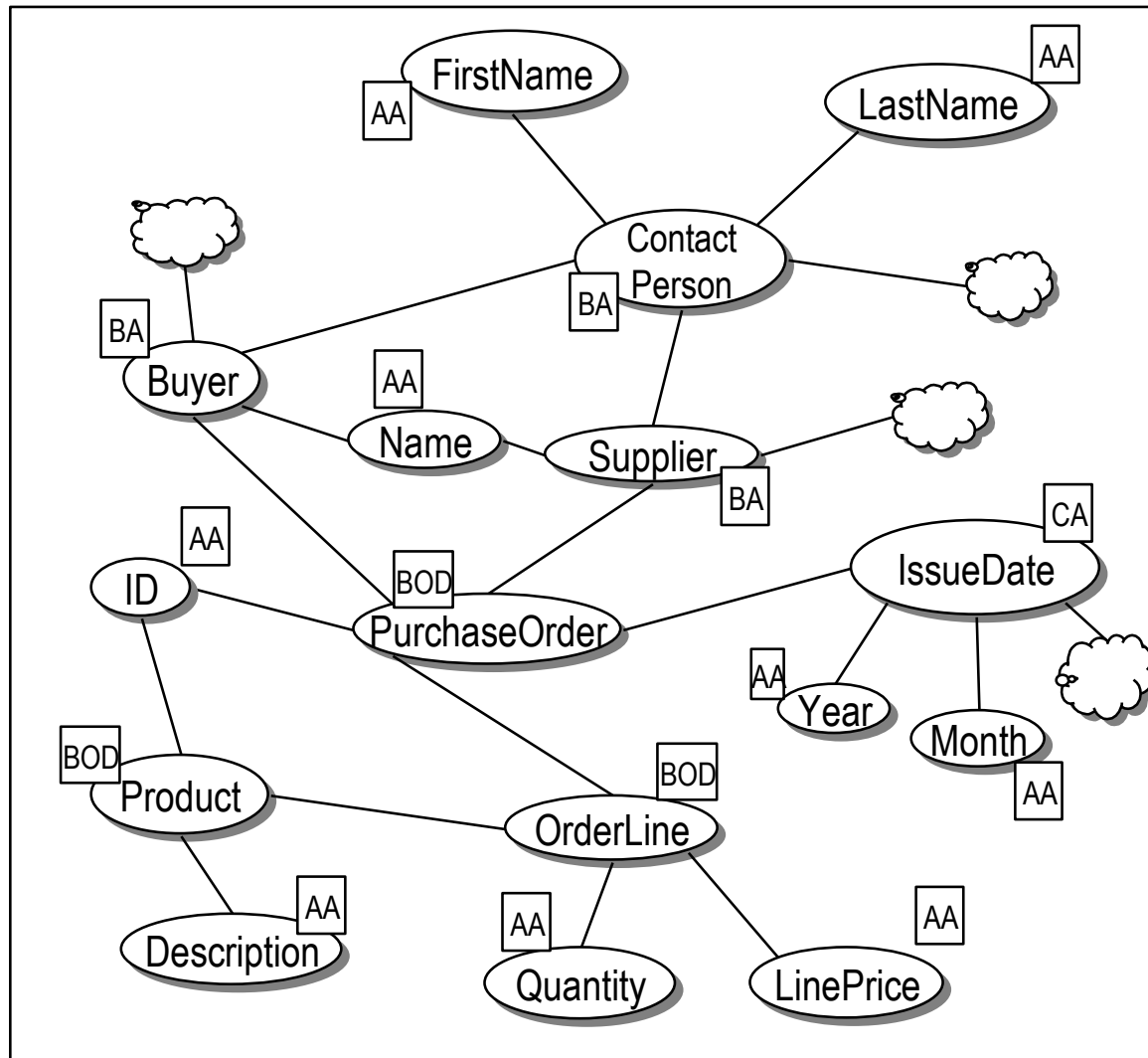


Concepts template

Building an **OPAL concept** means filling a template with the following structure



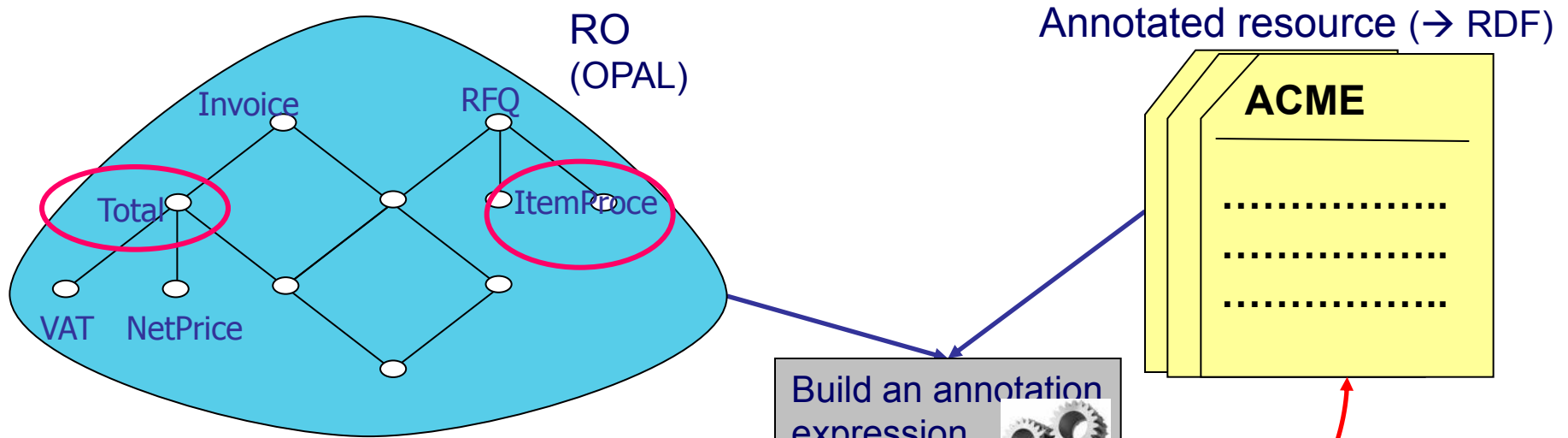
Ontology as a semantic net





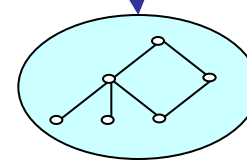
Semantic Annotation

Semantic Annotation Process overview



- Building an annotation expression:
 1. Using an existing concept of RO
 - or
 2. Creating a new concept by composing elements of RO
- Linking the annotation expression to the resource

Annotation expression



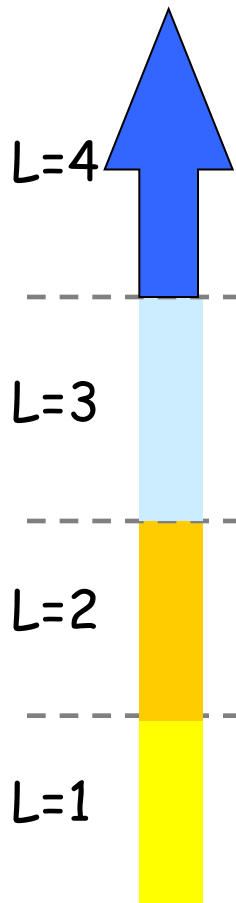
$\{ x | \exists y (Invoice(x)$

$\wedge hasTotal(x,y) \wedge Total(y)) \}$



A Semantic Annotation progressive approach

SA methodology defined as a **stepwise approach**



L=4 **Full SA (FSA)**, The annotation is an **OWL expression**.

L=3 **Simple SA (SSA)**, Paths from the PSA level are combined to build **simple expressions** by using abstract operators (\oplus , binary operator; φ , unary operator). Identification of the covered mismatch

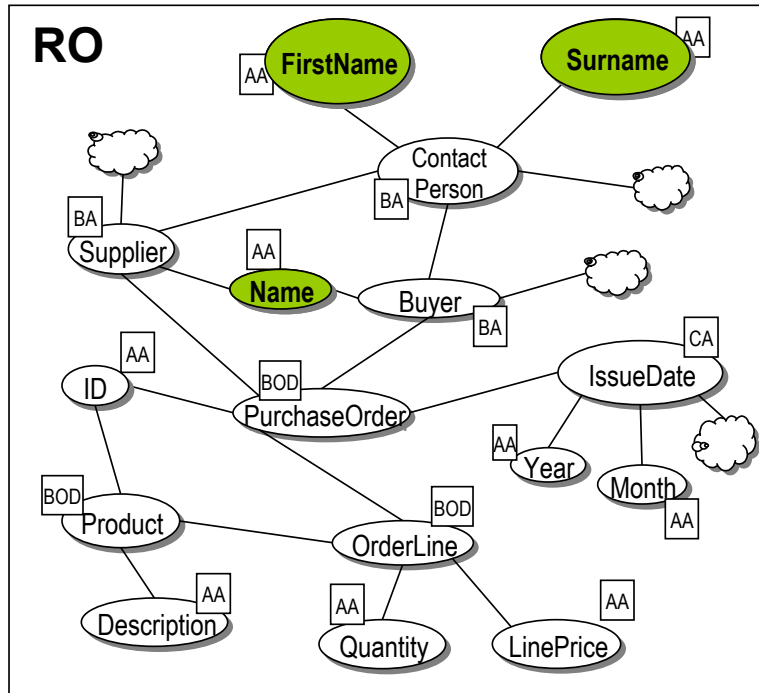
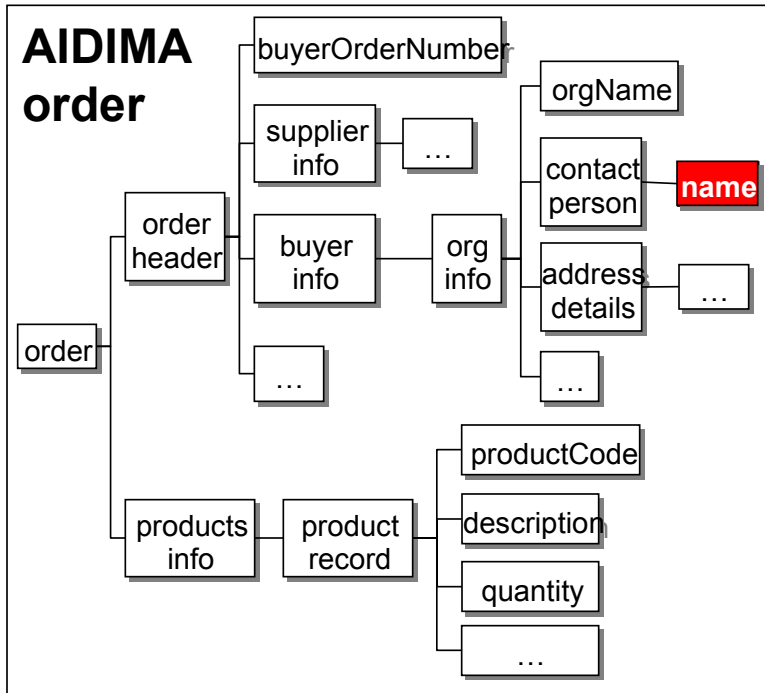
L=2 **Path SA (PSA)**, the annotation is represented by a **set of paths** from the RO

L=1 **Terminological SA (TSA)**, The annotation is represented by a **set of terms** taken from the definition of concepts in the RO



Terminological SA – an example

The annotation is represented by a **set of terms** taken from the definition of concepts in the RO

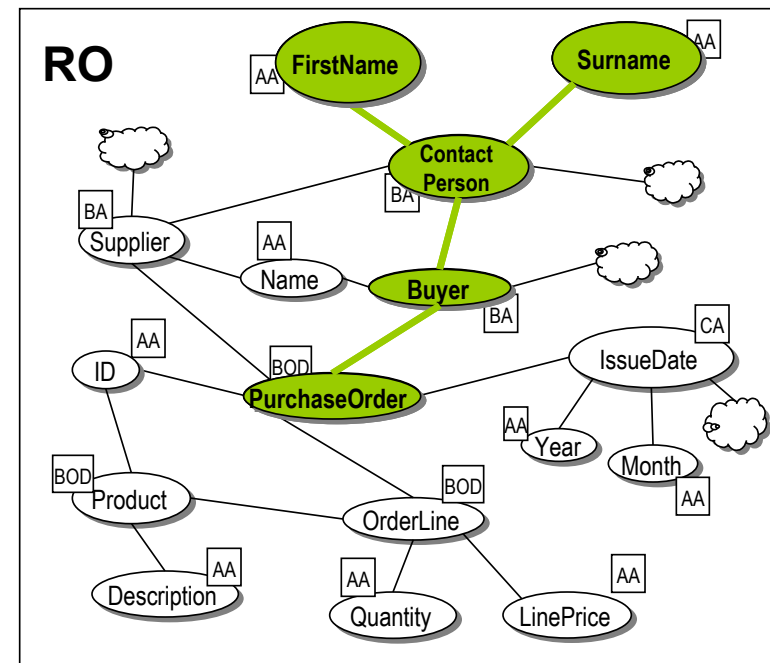
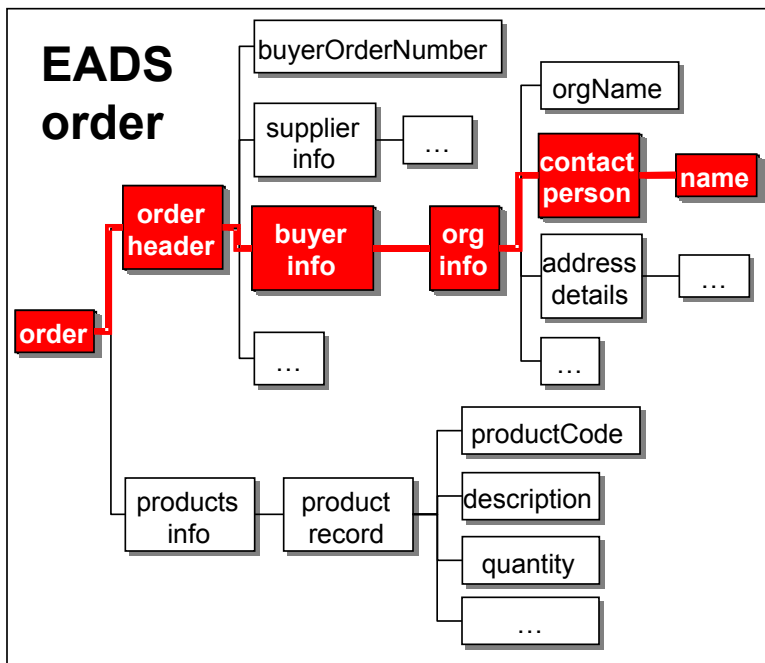


order	=: PurchaseOrder_BOD	} TSA Set
buyerInfo	=: Buyer_BA,	
contactPerson	=: ContactPerson_BA	
name	=: FirstName_AA, Surname_AA, Name_AA	
...		



Simple SAX – an example

- Identification of the covered mismatch kind
- This level is taken as specification for the **Transf Rules** definition



order.has_orderHeader.has_buyerInfo.has_organisationInfo.has_contactPerson.**has_name**

=:

PurchaseOrder_BOD.relTo_Buyer.relTo_ContactPerson.**hasPart_FirstName** ⊕

PurchaseOrder_BOD.relTo_Buyer.relTo_ContactPerson.**hasPart_Surname**

SSA
Expression
(Granularity/
Splitting)



Transformation Rules



Transformation Rules

- **Semantic Annotation** as *declarative* specifications for mismatches solution
 - a **conceptual correspondence** between resources and concepts in the ontology
- **Transformation Rules** as an *operational* specification for transforming ground resources (i.e., data) into ontology instances (forward transf.) and viceversa (backward transf.)
- The reconciliation platform is based on the **Jena2** suite (an open-source reasoning platform produced by HP)
- Rules are written in accordance with the **Jena2 rules syntax**
- The transformation Rules building is supported by the **Argos** tool



ARGOS: a Transformation Rules building tool

- A graphical environment supporting a user in defining transformation rules guided by
 - Document Schema
 - Semantic Annotations
 - Reference Ontology
 - A set of Rule Templates (e.g., map, split, merge, ...)
- using an abstract but expressive syntax

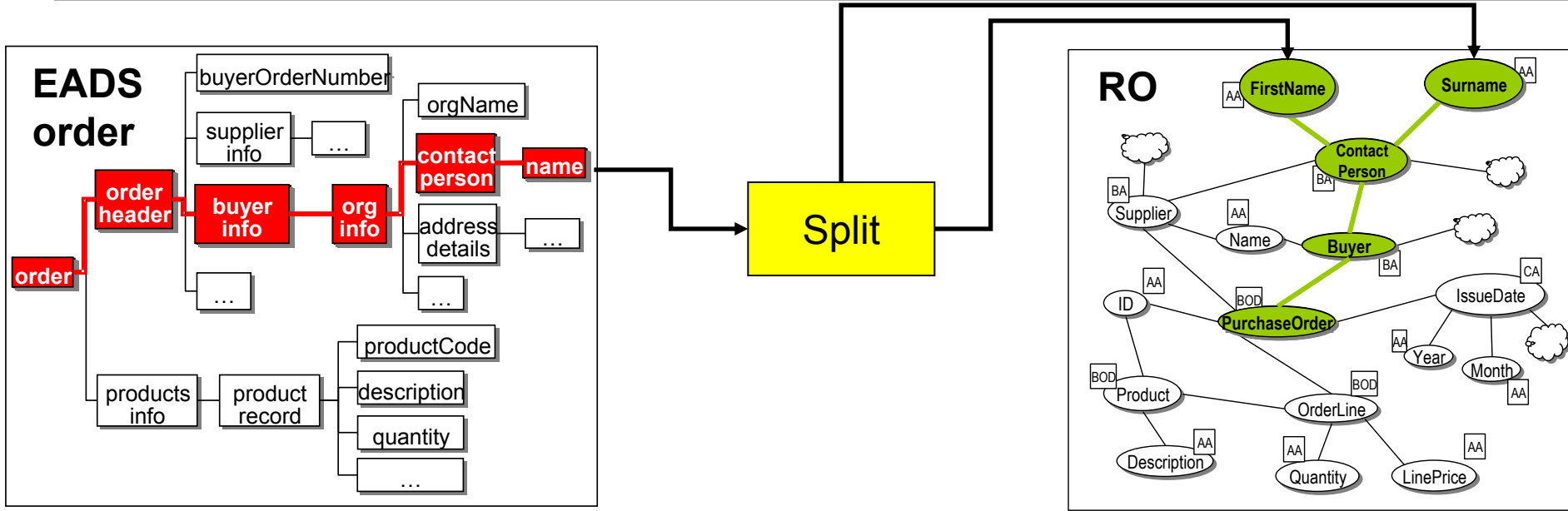


instantiated Rules are *automatically transformed* by ARGOS into executable code ([Jena rules](#)) for the reconciliation engine (ARES)

- An intuitive interface supports the user in parametrising *transformation templates* (Rule Templates)



From Semantic Annotation to Transformation Rules



order.has_orderHeader.has_buyerInfo.has_organisationInfo.has_contactPerson.has_name

=:

PurchaseOrder_BOD.relTo_Buyer.relTo_ContactPerson.hasPart_FirstName ⊕

PurchaseOrder_BOD.relTo_Buyer.relTo_ContactPerson.hasPart_Surname

SSAX

SPLIT

order.has_orderHeader.has_buyerInfo.has_organisationInfo.has_contactPerson.has_name

INTO

PurchaseOrder_BOD.relTo_Buyer.relTo_ContactPerson.hasPart_FirstName

PurchaseOrder_BOD.relTo_Buyer.relTo_ContactPerson.hasPart_Surname

SEPARATOR blankspace

Forward
Transf Rule

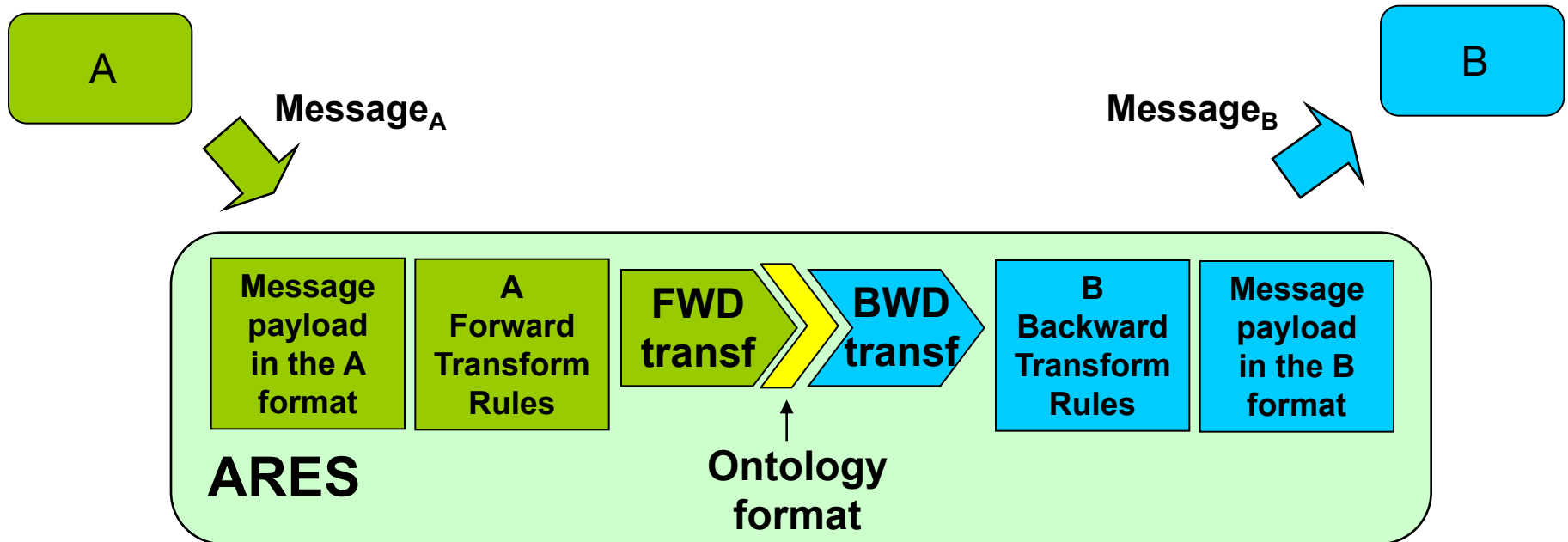


Reconciliation execution on a document instance



Reconciliation execution process

- Actual document reconciliation from A to B is performed by **Ares**, applying the previous defined Transformation Rules
- Two transformation:
 - **Forward**, from the A format into the Ontology format (FWD rules)
 - **Backward**, from the Ontology format into the B format (BWD rules)





An example of forward transformation

SPLIT

order.has_orderHeader.has_buyerInfo.has_organisationInfo.has_contactPerson.**has_name**

INTO

PurchaseOrder_BOD.relTo_Buyer.relTo_ContactPerson.**hasPart_FirstName**

PurchaseOrder_BOD.relTo_Buyer.relTo_ContactPerson.**hasPart_Surname**

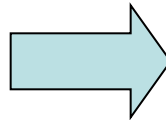
NameSplitting
Transform Rule

SEPARATOR blankspace

An instance of the EADS order

```
<ai:order rdf:ID="ex_AIDIMA_order_1">
  <ai:has_orderHeader>
    <ai:orderHeader rdf:ID="ordHea_ID1">
      <ai:has_buyerInfo>
        <ai:buyerInfo rdf:ID="buyInf_ID1">
          <ai:has_organisationInfo>
            <ai:organisationInfo rdf:ID="orgInf_ID1">
              <ai:has_contactPerson>
                <ai:contactPerson rdf:ID="conPer_ID1">
                  <ai:has_name>John Smith</has_name>
                </ai:contactPerson>
              </ai:has_contactPerson>
            ...
          ...
        ...
      ...
    ...
  ...
</ai:order>
```

**Name
Splitting
Rule
Application**



The EADS order in the Ontology format

```
<ro:PurchaseOrder_BOD rdf:ID="ex_AIDIMA_order_1">
  <ro:relTo_Buyer>
    <ro:Buyer_BA rdf:ID="buyInf_ID1">
      <ro:relTo_ContactPerson>
        <ro:ContactPerson_BA rdf:ID="conPer_ID1">
          <ro:hasPart_FirstName>John</ro:hasPart_FirstName>
          <ro:hasPart_Surname>Smith</ro:hasPart_Surname>
        </ro:ContactPerson_BA>
      ...
    ...
  ...
</ro:PurchaseOrder_BOD>
```



COIN innovative services for information interoperability



Motivations and objectives

Motivations

- Semantic annotation and building of transformation rules are mainly manual activities (error prone and time consuming)
- Two transformations at run time (forward and backward)

Objectives

- Improving and extending the semantic reconciliation suite in three directions:
 - **Semantic Declarative Mapping Discovery:** semi(automatic) support to semantic annotation
 - **Semantic Operational Mapping generation:** semi(automatic) support to semantic transformation rules generation
 - **Transformation rules fusion:** to optimize actual reconciliation phase



Semantic Declarative Mapping Discovery

Objective: automatically support the Semantic annotation definition (currently is mainly a manual activity)

Organized into 3 incremental steps

- Terminology-based mapping
 - using Web resources (e.g., Wordnet)
 - and similarity techniques for assessing similarity b/w terms
- Graph matching techniques for structure-based mapping with terminological mapping as input
- Semantic mismatch patterns (lossless and lossy) discovery



Semantic operational mapping generation

- Objective: automatically support the TR building by reusing semantic annotation knowledge
- Semantic transformation rules patterns
 - From Declarative to Operational Mapping
 - Correspondence between semantic mismatch patterns and transformation rules patterns
 - Automatic (partial) filling of TR templates
 - From Abstract Operational Mapping to executable transformation rules

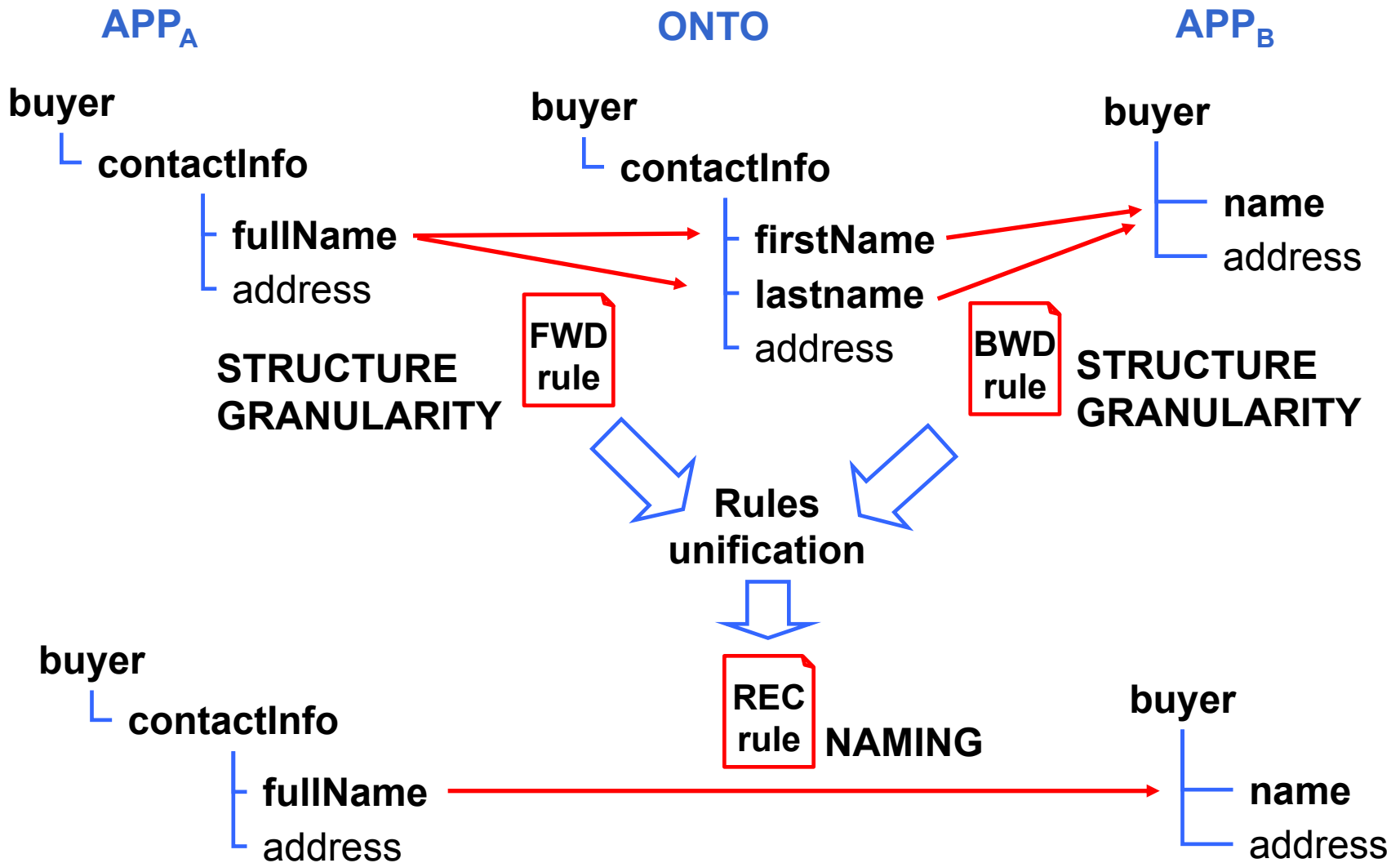


Transformation Rules Fusion

- Objective: to fuse sender's forward and receiver's backward rules in order to improve actual doc exchange
 - More straight forward reconciliation
 - Avoid useless transformations
- Semantic preserving fusion
 - Finite set of transformation patterns
- Resulting rules
 - still semantic rules (i.e., the ontology is still behind)
 - but no actual generation of intermediate document representation (ontology-based)

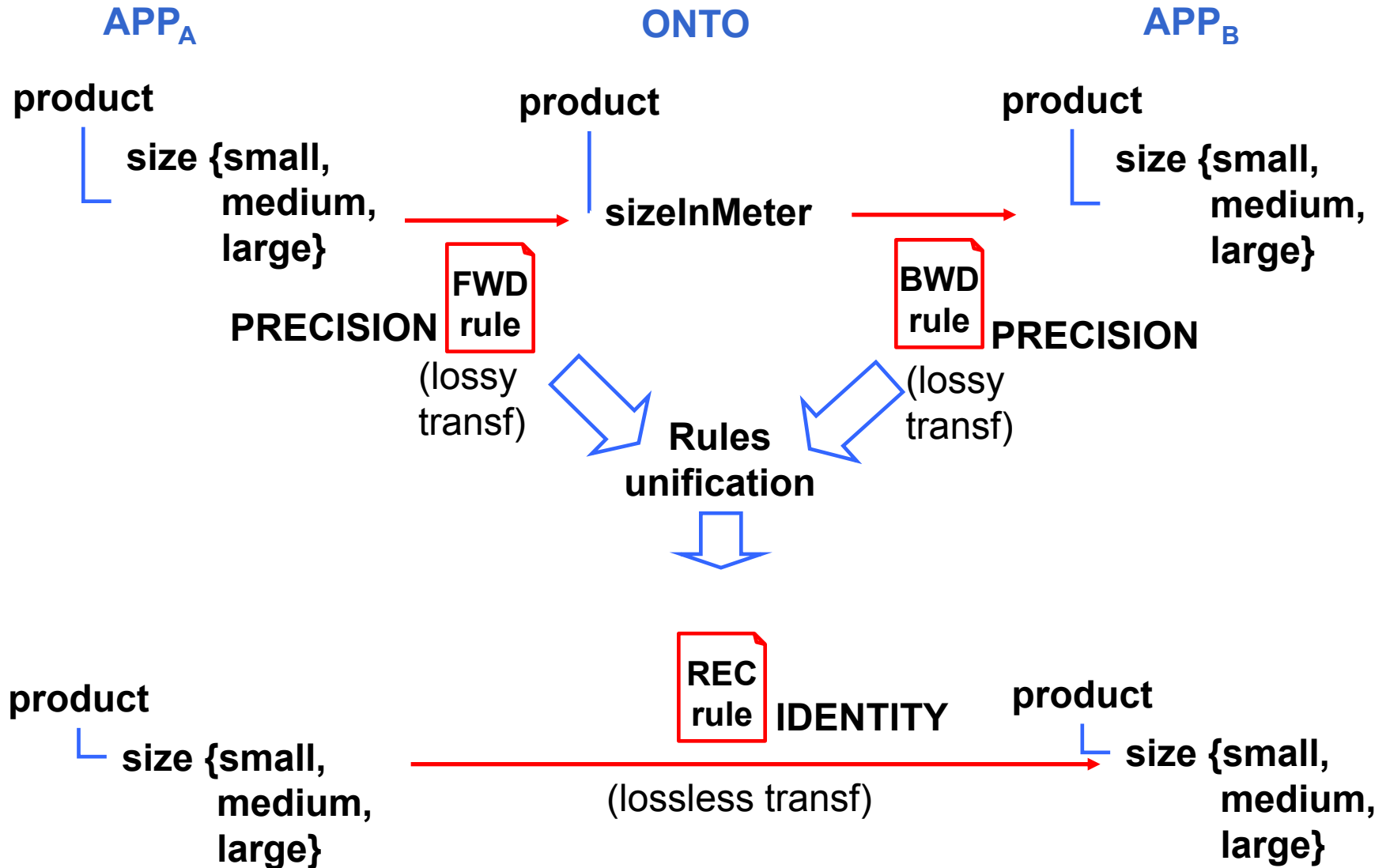


Example of Rules Fusion



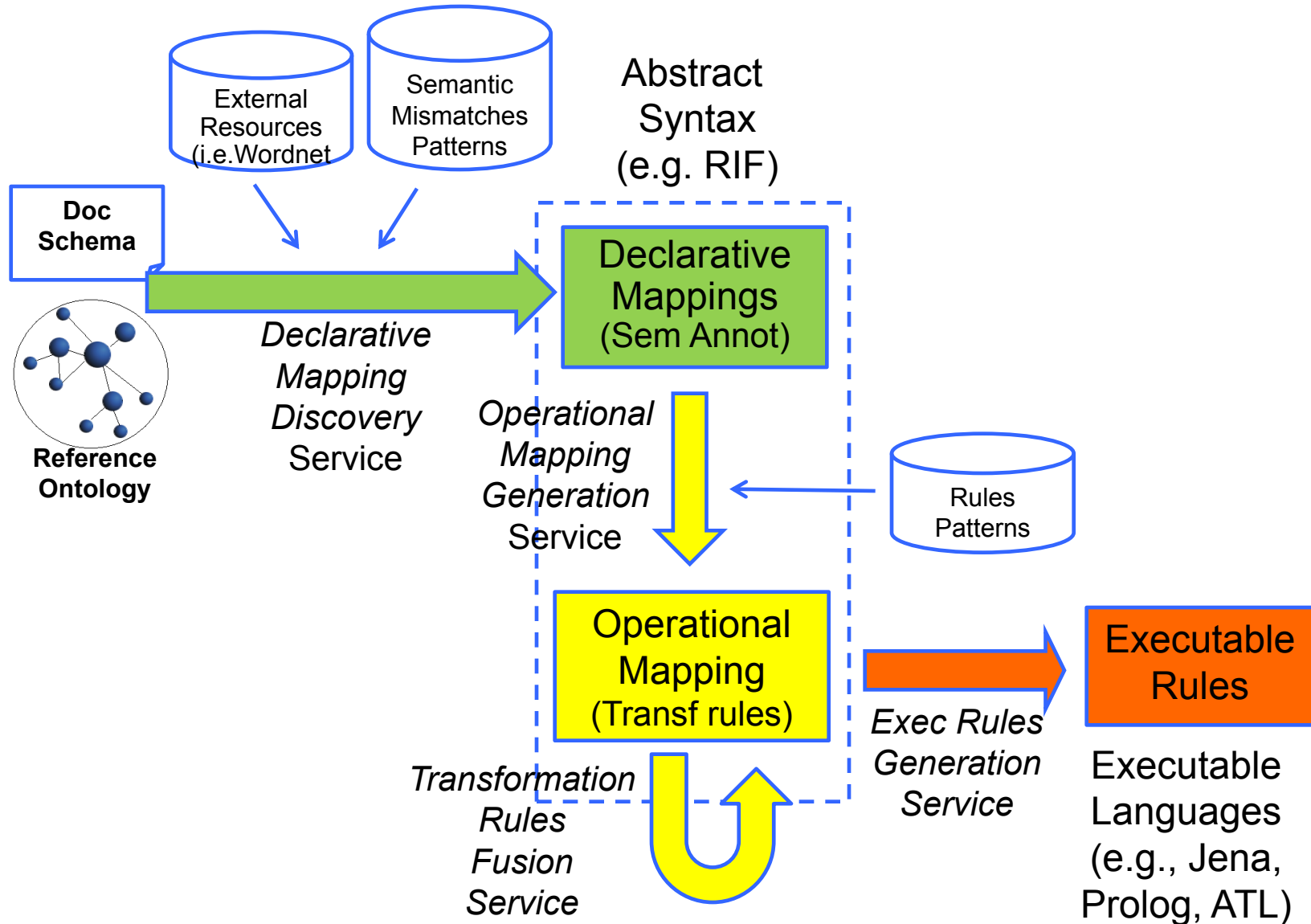


Example of Rules Fusion





COIN information interoperability services





Knowledge interoperability services



Objectives

Achieving balanced and harmonised **Collaborative Network**

- Modelling **Enterprise Semantic Profile (ESP)** focussing on **Competencies** and **Skills (CS)**
- Semantically enriching **CS** of each enterprise in the CN to:
 - Understand the **overall CS** of the CN
 - Identify **CS gaps and complementarities**
 - Define the “ideal” CS asset for the CN and identify the **missing CS** (e.g., to be acquired with new partners)

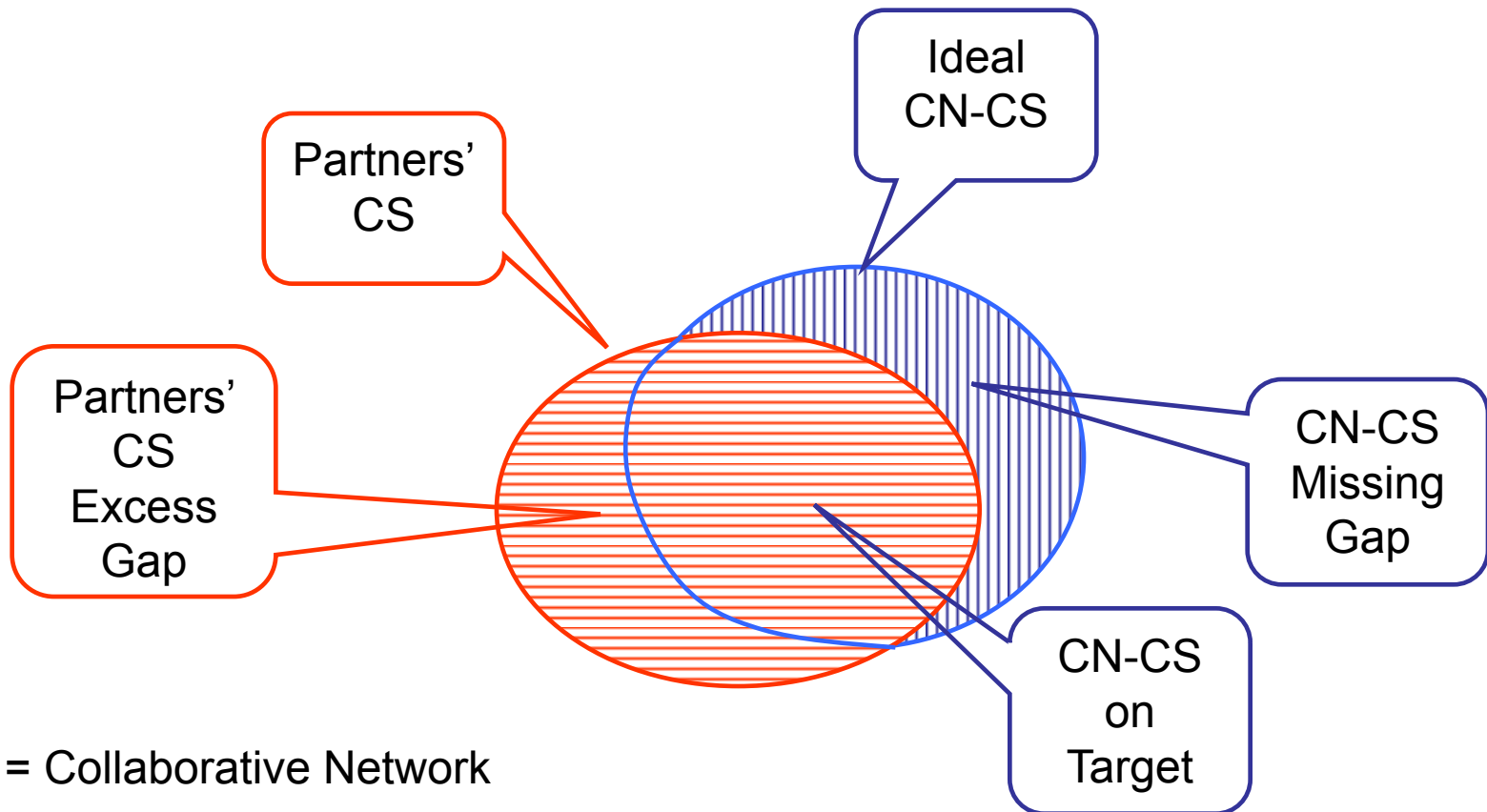
To this end, we need to develop:

- Semantic Supporting Services: **SSS**, necessary for KIS
- Knowledge Interop Services: **KIS**, to support the harmonisation of CS for better enterprise cooperation in the CN



CS Map (gaps and overlapping)

We consider: CS of partners, ideal CS of CN, CS Missing Gap



CN = Collaborative Network



Beyond the State of the Art (1)

- **Objective:** achieve a unified method to model Enterprise capabilities (i.e., CS)
- **SotA:** Very complex and heterogeneous Enterprise Modelling (EM) frameworks (e.g., POP*, CIMOSA, GERAM, Zachman)
- **Innovation:** Describe CS by using ontology-based **Enterprise Semantic Profiles (ESP)**:
 - **Extract CS** from the human-oriented knowledge (filtering the technical documentation)
 - Map the filtered Enterprise Knowledge to the *CSOnto*, creating the **ontology-based ESP**



Beyond the State of the Art (2)

- **Objective:** ontology-based analysis of overlapping and complementary CS through ***Semantic Similarity Reasoning***
- **SotA:** A plenty of semantic similarity methods (e.g., Lin, Resnik, ...) for comparing pairs of concepts, not semantic profiles
- **Innovation:** define an enterprise semantic matchmaking method for
 - Computing similarity b/w ESP
 - Identify gaps/overlapping among ESP



Knowledge EI Services

Semantic Supporting Services (SSS)

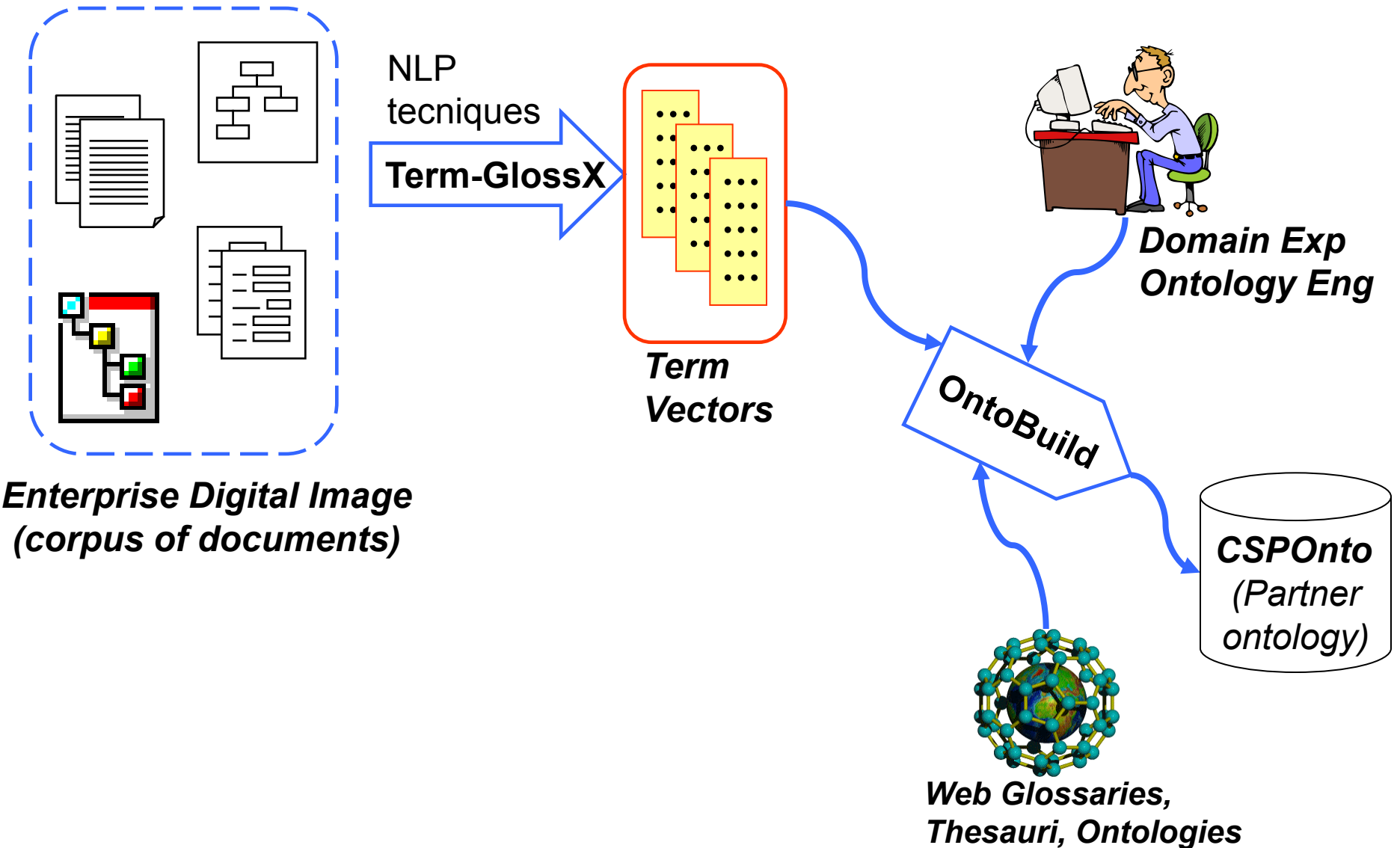
- Mass Ontology Building (from doc & Web 2.0)
- Enterprise Semantic Profiling
- Enterprise Semantic Matchmaking

Knowledge Interoperability Services (KIS)

- Semantic analysis of new partners entering CN
- Semantic analysis of enterprises' CS in CN, to identify CS gap & overlapping
- Semantic enhancement of CN CS, by transferring the Knowledge acquired during VO activities / loss of competencies



Mass Ontology Building

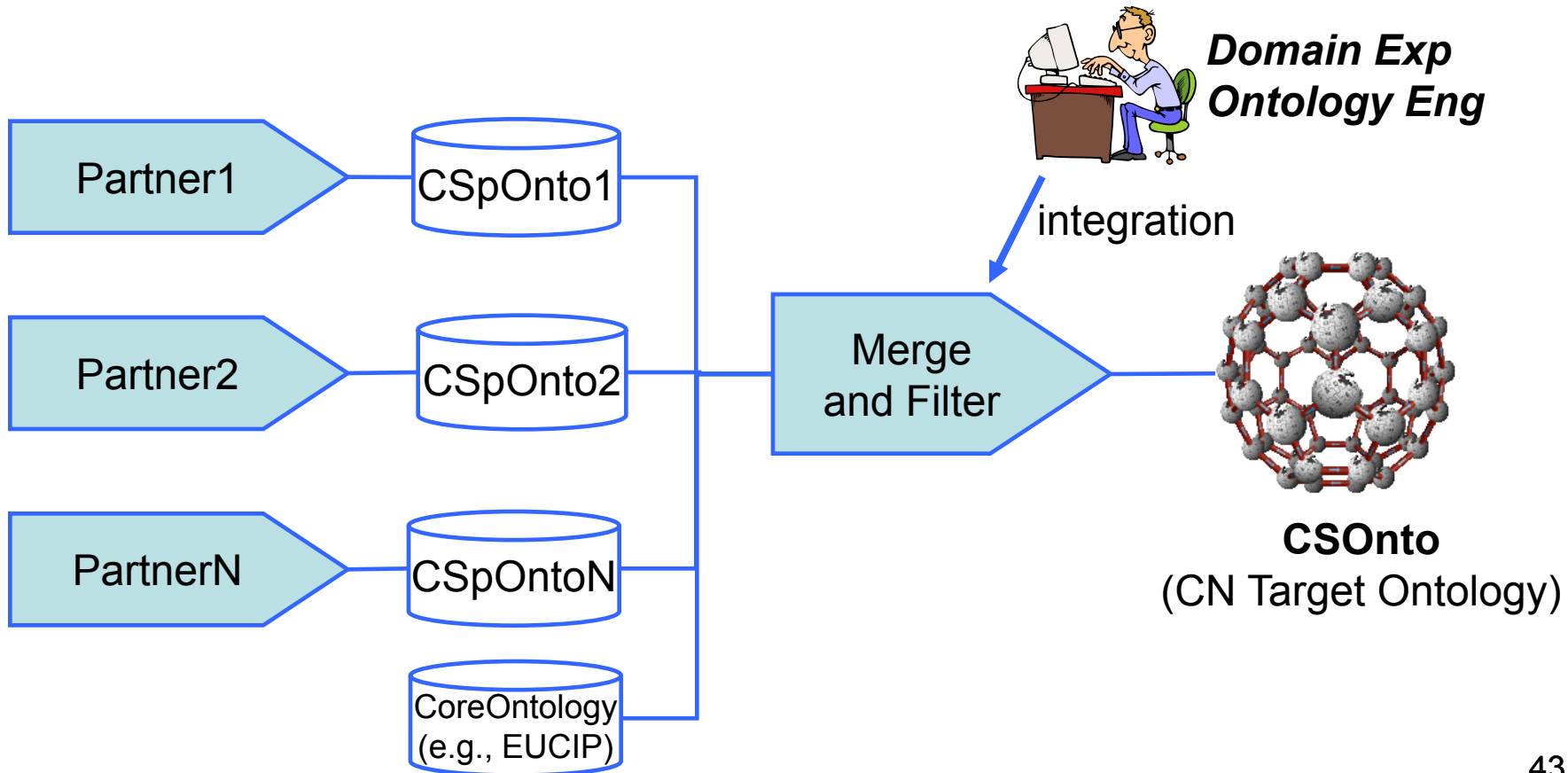


NLP = Natural Language Processing



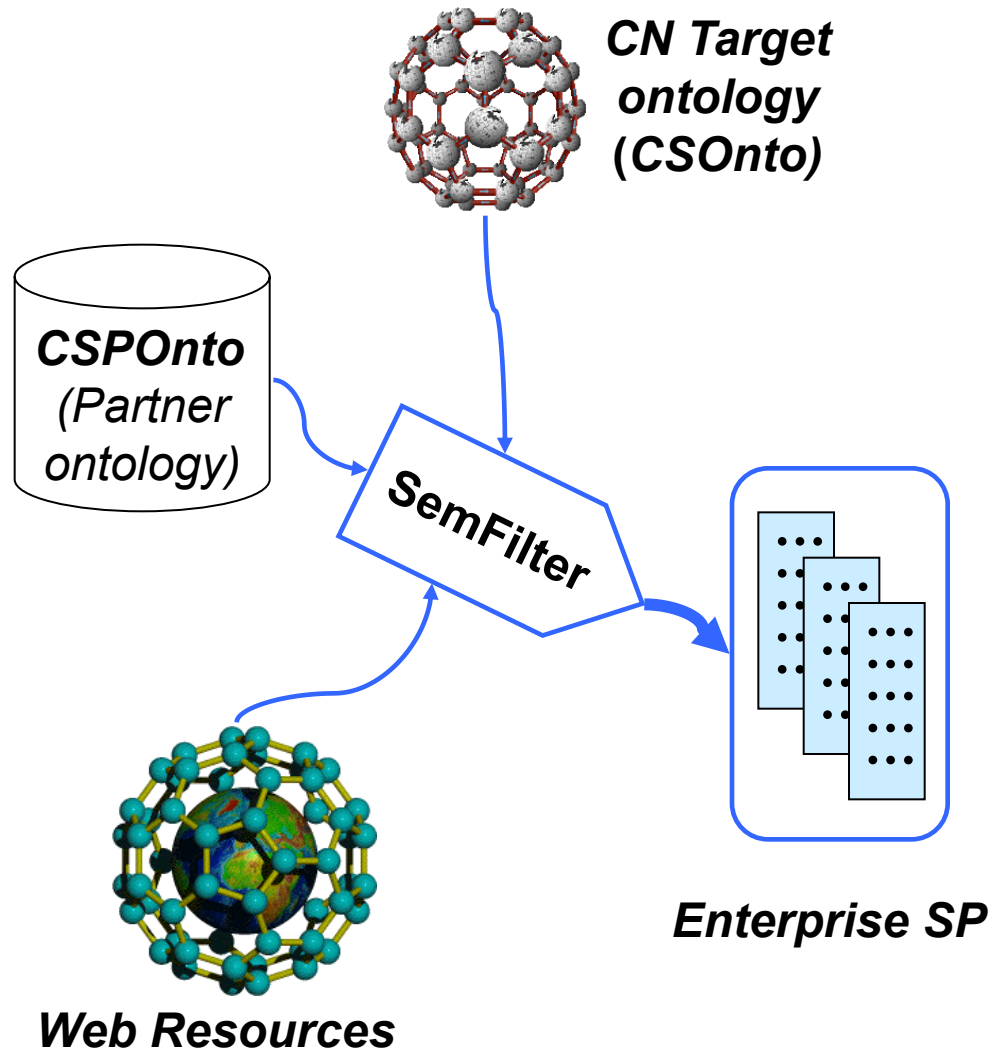
Building a semantic-aware CS asset

- The CN CS asset is mainly derived from the partners' asset



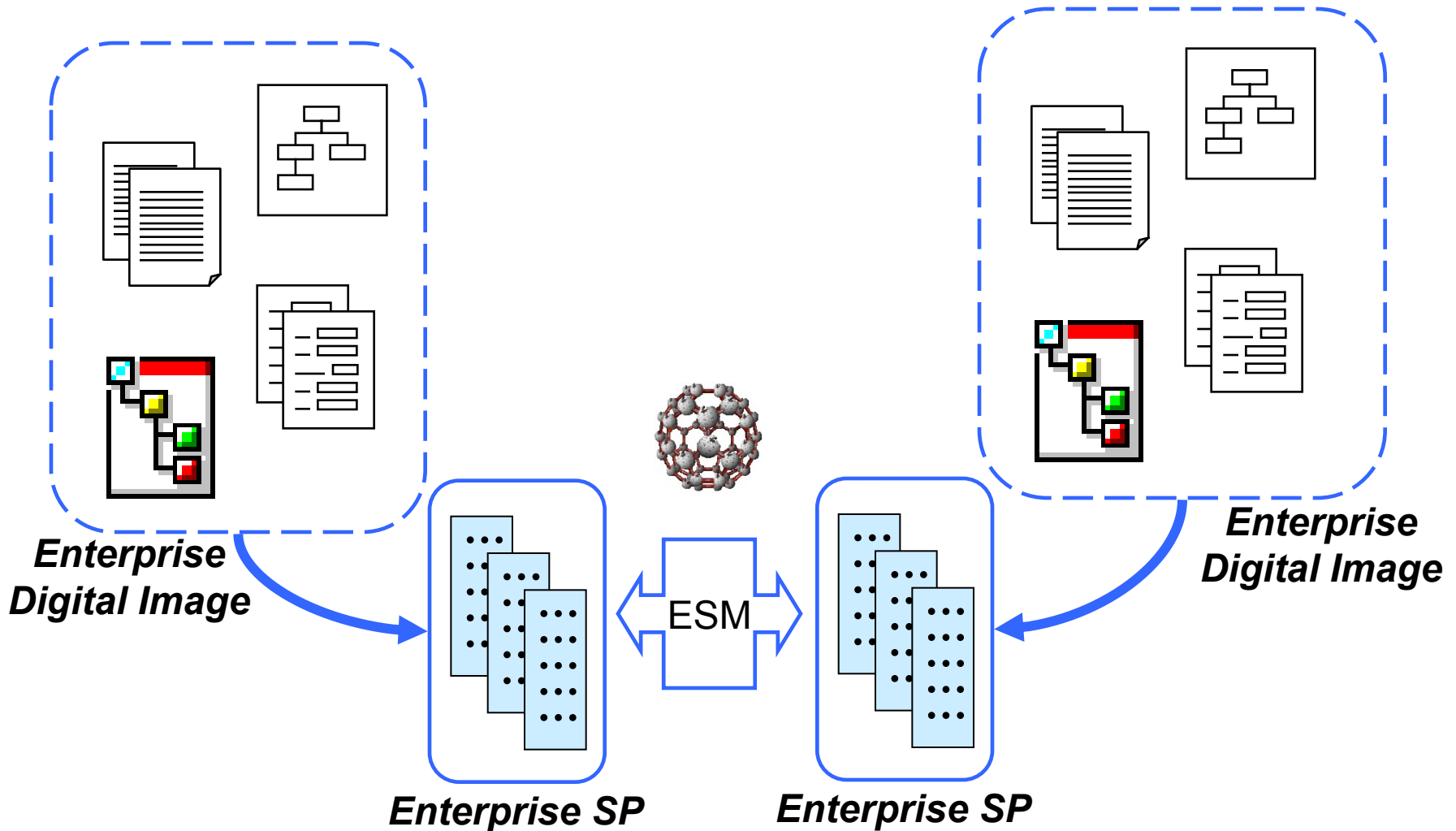


Enterp Semantic Profile Building



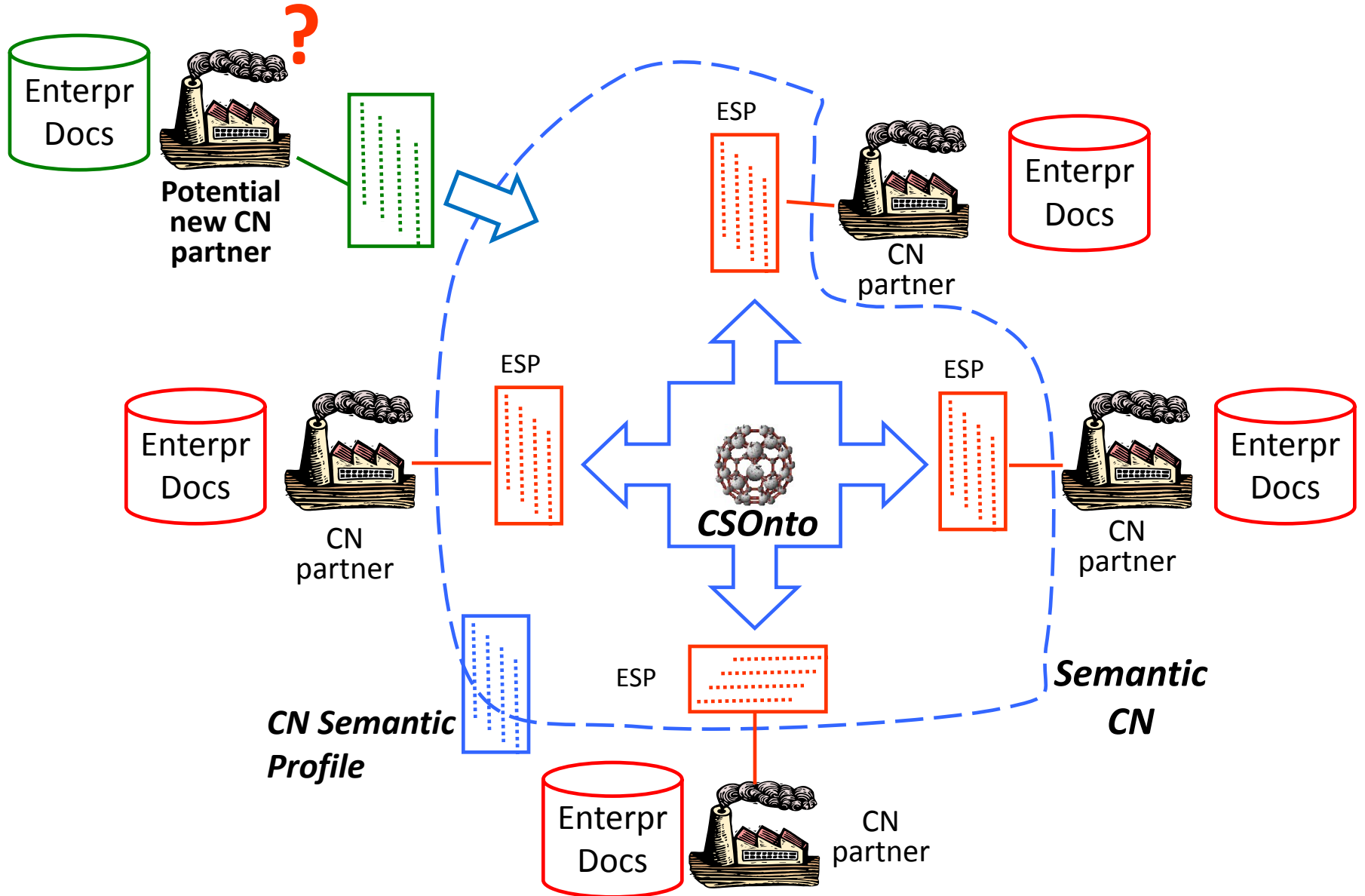


Enterprises Semantic Matchmaking





Semantic Collaborative Network





Conclusions

- Two kinds of semantic interoperability services for
 - Reconciliation of business documents for documents exchange between heterogeneous software applications
 - Management of enterprise competencies and skills for harmonization of collaborative networks