

# ***Semantic Help***

An eGov2007 Workshop

## **An ontology-based method for Business Process anomalies detection and correction**

***Michele Missikoff***  
LEKS-IASI, CNR  
(Rome, Italy)

Regensburg, 3 September 2007

# Objectives of the talk

---

- Motivate: why we need ontological modeling of Business Processes
- Introduce **Business Process Ontologies**
- Why **Semantic Help** is relevant when talking about BP
- A **deductive approach** to ontologies and Semantic Help

# A World of Processes

---

- My life, your life is a **process**, intricate, convoluted, fuzzy, often unconscious and escaping rationality
- Alternatively, it can be seen as structured as an immense **constellation** of inter-related processes
- Each life **event** triggers a new life process
- The request of a certificate triggers a process
- The interaction with a public office is inscribed within a process
- Our lives are complex, so are the processes we implement
- It would be nice to **shad** some light on a few of them
- To realise methods and mechanisms for:
  - supporting us in modeling our life process,
  - keeping under control their unfolding,
  - support us in the different phases,

# On the limits of modeling

---

- ... and, according to Borges, the impossibility to draw a 1:1 map of the territory ...
- This means that modeling means abstracting, i.e., dropping inessential details, capturing representing only relevant information
  - Therefore ... modeling is a hard job, error prone.
  - Modellers need a lot of support, by:
    - coordinated methods and tools,
    - grounded in solid mathematics,
    - but at the same time intuitive for the user and
    - easy to be learned and applied, therefore
    - ... natural and usable

# Why Process Ontologies?

---

- A solid background can be obtained by using semantic technologies
- A rich ontological framework may provides useful services
- OPAL-BPAL: a structured, stratified ontological framework
- A strong logic-based rooting, to access a wealth of semantic services
  - Consistency checking
  - Search and retrieval
  - ...

# A rich ontological framework

---

- ***Structural ontology*** – in **OPAL**
  - A conceptual catalogue of what exists
  - A static view of the reality
- ***Behavioral ontology*** – in **BPAL**
  - The principle according to the entities evolve and interact
  - A dynamic view of the reality
- BPAL-O can only describe the behavior of entities defined in OPAL-O
- Enrichment with Axioms: rules and constraints

# Focus of the presentation

---

- Provide an ontological framework for process ontologies

Covering the functional areas:

- ***Modeling***
- ***Manipulation***
- Search and retrieval
- Update
- Consistency checking

# A formal foundation

---

- Ontology:

$$\mathbf{Onto} = (\mathbf{C}, \mathbf{R}, \mathbf{A})$$

Where:

- **C**: set of concepts (e.g., unary pred, class)
- **R**: set of n-ary relations (e.g., n-ary pred, association)
- **A**: set of axioms (e.g., FOL formulae)

We need a language to build the above sets:

**BPAL**

(Business Process onto Abstract Language)

# Semantic Help of BPAL

---

## Support

- the building of formal, well founded BP models
- the evolution of BP, with their update
- the search and retrieval
- the merging of different processes

# BPAL: a quick look

---

The essence of a BPAL Diagram:

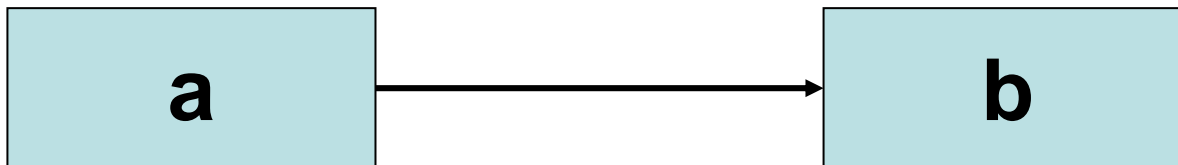
- ***Unary pred***

***act(a), act(b), dec(k)***

- ***Binary pred***

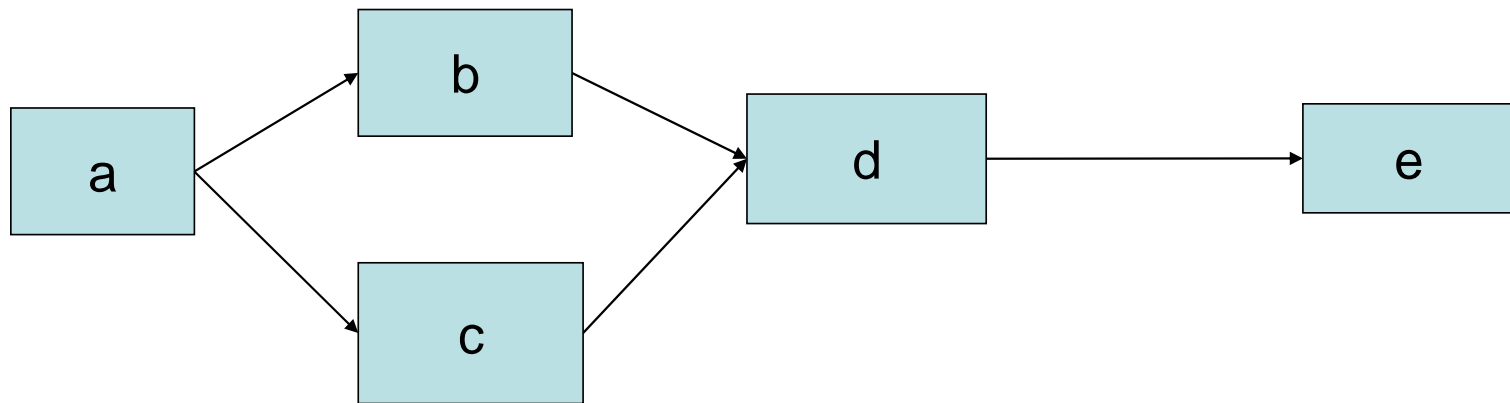
***prec(a,b)***

A **BPMN** diagram



# An example

*act(a), act(b), act(c), act(d), act(e);  
prec(a,b), prec(a,c), prec(c,d), prec(b,d), prec(d,e).*



*A graceful BP definition methodology: progressive, but monotonically consistent  
process knowledge representation*

# Axioms: an example

**Branching Axiom** - *If a node is followed by two or more immediate successor activities, then it must be a decision.*

$$\forall x, y \in CBO : act(y) \wedge S(x) = \{y \in CBO : prec(x, y)\}$$
$$\wedge |S(x)| > 1 \rightarrow dec(x)$$

*(this is a modeling axiom. More critical are **domain axioms**)*

# Semantic Help

---

Rule-based Recommendation System

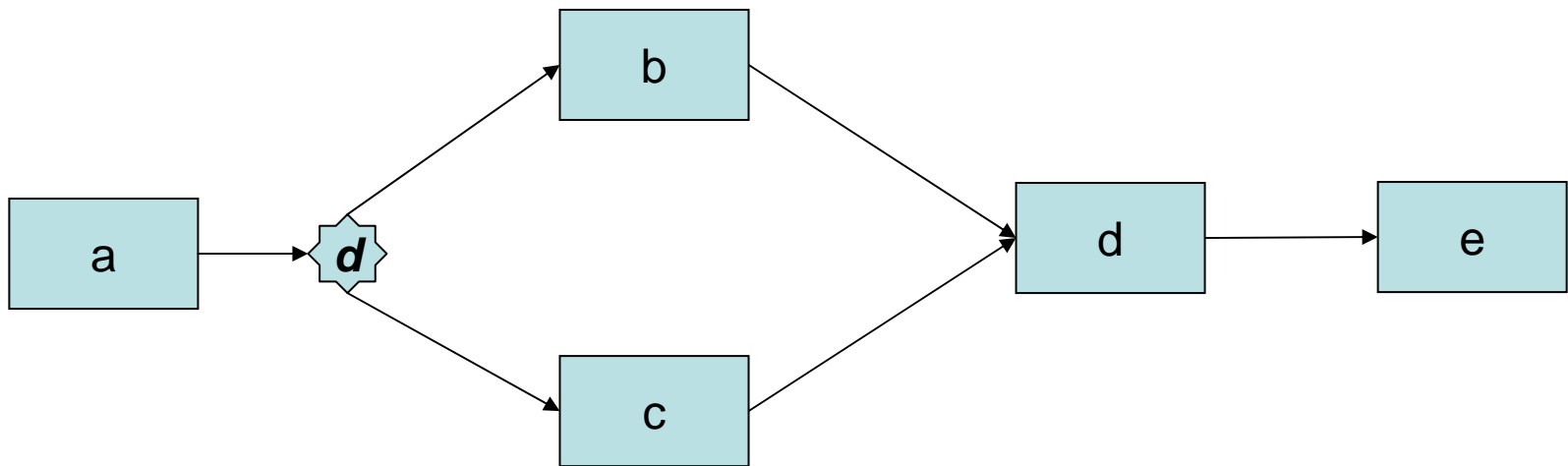
**IF** <BranchingAxiom = False> **THEN**

*assert*([*dec*(*k*), *prec*(*k*,*b*), *prec*(*k*,*c*), *prec*(*a*,*k*)]),  
*retract* ([*prec*(*a*,*b*), *prec*(*a*,*c*)]).

The resulting BPAL process will be...

# The correct BP

*act(a), act(b), act(c), act(d), act(e), dec(k);*  
*prec(a,k), prec(k,b), prec(k,c), prec(c,d), prec(b,d), prec(d,e).*



# Unary predicates

---

***act(\_a)*** - a business activity, element of an abstract diagram

***role(\_x)*** - a business actor, involved with a given role in one or more activities

***dec(\_bexp)*** - a generic decision point. Its argument is a Boolean expression evaluated to {true, false}. It is used in the preliminary design phases when developing a BP with a stepwise refinement approach. In later phases, it will be substituted with one of the specific decision predicates (see below).

***adec(\_bexp)***, ***odec(\_bexp)*** - decision points representing a branching in the sequence flow, where the following paths will be executed in parallel or in alternative, respectively

***cont(\_obj)*** - an information structure. For instance a business document (e.g., birthCertificate)

***cxt(\_obj)*** - a context, represented by a collection of information structures

***goal(\_bexp)*** – a desired state of the world

# Conclusion

---

- BP Ontology are very important in eGov
- Among the wide variety of methods for process modeling, we propose BPAL
- BPAL is rooted in mathematical logic
- We are building a full framework to support the building of
  - BP Ontologies
  - Domain Axioms
  - Engine to check consistency
  - Recommendation Engine to support the correction of detected deviations