Conceptual modelling, ontology design, and semantic interoperability

Professional master on technologies for e-government
Lecture 10

Roberta Ferrario
Laboratory for Applied Ontology, ISTC-CNR
ferrario@loa-cnr.it

Trento, 31.10.2007

Outline

- ▷ A proposal of a model of organizations
- ▷ Foundational Ontologies and interoperability in e-gov
- ▷ Summary of the course

A proposal of a model of organizations

Basic Assumptions of the Model

- ▷ Organizations are observed and analyzed by taking the **teleological** perspective
- ▷ The logic assumed is **first order modal logic with identity**, where states of affaires are represented by propositions and the material implication between two propositions represents specialization between states of affaires ($\phi \rightarrow \psi$)
- ▷ **Discrete branching time** is assumed

Time (1)

We represent time with the following modal operators, i.e. we don't refer directly to moments in the theory, but only via these modal operators

$\Box\phi$: "ϕ is true at all the moments";

$\overrightarrow{\Box}\phi$: "ϕ is true at all the moments (including the present) in the future";

$\overleftarrow{\Box}\phi$: "ϕ is true at all the moments in the past";

$\diamond\phi$: "there exists at least a moment where ϕ is true";

Time (2)

$\overrightarrow{\Diamond}\phi$: "there exists at least a moment in the future (including the present) where ϕ is true";

$\overleftarrow{\Diamond}\phi$: "there exists at least a moment in the past where ϕ is true";

$\bigcirc\phi$: "ϕ is true at the next moment";

$\overline{\bigcirc}\phi$: "ϕ is true at the previous moment".

Agentivity, Actions and Organizations

- Organizations are assumed to be specific kinds of agents, thus they have mental attitudes, like **goals** and **beliefs**
- The modal operator $\mathsf{G}_a\phi$ stands for "the agent a aims at achieving the state ϕ (called the goal)"
- The modal operator $\mathsf{B}_a\phi$ stands for "the agent a believes that the state/proposition ϕ obtains/is true"
- We also introduce the set of *actions* (ACT) and a finite set of *types of actions* $(\alpha, \beta, ...)$ where $\alpha x \rightarrow \text{ACT}x$

Actions and preconditions

- ▷ We take the operator $\mathsf{E}_a^\alpha\phi$ to mean "the agent a has completed the execution of one action of kind α and ϕ is now (at the present moment) true"
- ▷ $\mathsf{E}_a^\alpha\top$ is instead used to mean "agent a executed an action of type α", without specifying the goal of the action
- ▷ We also take $\overline{\mathsf{E}}_a^\alpha\phi$ to mean"when agent a started the execution of action α, ϕ was true"
- ▷ Both these operators can be generalized by abstracting away from specific agents
- ▷ The preconditions π^α of action type α are thus expressible in the following way:
  $\Box((\mathsf{E}^\alpha\top \rightarrow \overline{\mathsf{E}}^\alpha\phi) \leftrightarrow (\pi^\alpha \rightarrow \phi))$

Basic and Complex Actions

- **Basic** actions are the most specific actions; temporally, they are simple transitions from one moment to the successive one
- **Complex** actions are sequences (parallel) of basic actions
- $\mathsf{cmp}(\alpha|\alpha_1, ..., \alpha_n)$ is taken to mean that the action *type* α is a complex action coming from a structured composition of the action *types* $\alpha_1, ..., \alpha_n$ by $\mathsf{cmp}(\alpha|\alpha_1, ..., \alpha_n)$
- In order to express how the actions of kind $\alpha_1, ..., \alpha_n$ are structured, we constrain them at the level of execution: $\mathsf{cmp}(\alpha|\alpha_1, ..., \alpha_n) \wedge \mathsf{E}^{\alpha}\top \rightarrow \overleftarrow{\Diamond}\mathsf{E}^{\alpha_1}\top \wedge ... \wedge \overleftarrow{\Diamond}\mathsf{E}^{\alpha_n}\top$

Collective Actions

- Collective actions are complex actions that are composed by basic actions executed by different agents
- We need a way to constrain the executors of action types (types of agent) and we do this again by constraining execution: $\mathsf{cmp}(\alpha|\alpha_1{:}A_1, ..., \alpha_n{:}A_n) \wedge \mathsf{E}^{\alpha}\top \rightarrow \overleftarrow{\diamond}\mathsf{E}^{\alpha_1}_{A_1}\top \wedge ... \wedge \overleftarrow{\diamond}\mathsf{E}^{\alpha_n}_{A_n}\top$
- We can also express the fact that we can have specific executors for specific goals via *composition*$(\phi|\phi_1, ..., \phi_n)$, which expresses the fact that ϕ is a goal composed by subgoals $(\phi_1, ..., \phi_n)$: $\mathsf{cmp}(\alpha|\alpha_1{:}A_1, ..., \alpha_n{:}A_n) \wedge \mathsf{E}^{\alpha}_{[a_1,...,a_n]}\phi \rightarrow \overleftarrow{\diamond}\mathsf{E}^{\alpha_1}_{a_1}\phi_1 \wedge A_1(a_1) \wedge ... \wedge \overleftarrow{\diamond}\mathsf{E}^{\alpha_n}_{a_n}\phi_n \wedge A_n(a_n) \wedge \mathsf{cmp}(\phi|\phi_1, ..., \phi_n)$

Capabilities

- Capabilities are states of affairs agents can *potentially* achieve (thus, their definition involves a *modal* notion)
- $\mathsf{C}_a^\alpha \phi \triangleq \overrightarrow{\diamond} \mathsf{E}_a^\alpha \phi$ expresses the fact that agent a is capable of achieving ϕ by performing α
- $\mathsf{C}_a \phi \triangleq \mathsf{C}_a^\Lambda \phi$ expresses the fact that agent a is capable of achieving ϕ, without any limitation on the possible actions
- We can also express the fact that an agent is capable of achieving ϕ under some constraint θ:
  $\mathsf{C}_a^{\alpha|\theta} \phi \triangleq \overrightarrow{\diamond}(\mathsf{E}_a^\alpha \phi \wedge \overline{\mathsf{E}}_a^\alpha \theta)$

Collective capabilities and dependencies

- Collective capabilities identify what an agent a can achieve with the help of other agents $[a_1, ..., a_n]$. They are characterized respectively in this way:
  $\mathsf{cC}_a^\alpha \phi \triangleq \exists[a, a_1, ..., a_n](\mathsf{C}^\alpha_{[a,a_1,...,a_n]}\phi)$;
  $\mathsf{cC}_a \phi \triangleq \exists[a, a_1, ..., a_n](\mathsf{C}_{[a,a_1,...,a_n]}\phi)$.
- When an agent is not individually capable of achieving a goal, but needs the help of other agents, it ends up depending on these agents:
  $\mathsf{D}_{\{a|a_1,...,a_n\}}\phi \triangleq \mathsf{G}_a\phi \wedge \neg\mathsf{C}_{\mathrm{AG}\setminus\{a_1,...,a_n\}}\phi$
- Note that from $\mathsf{D}_{\{a|a_1,...,a_n\}}\phi$ it does not follow that $\mathsf{C}_{[a,a_1,...,a_n]}\phi$, the help of the other agents can be necessary but not sufficient to achieve the goal.

Plans (1)

1. At the beginning, an agent believes that an objective is achievable, given certain constraints; at this stage the plan can be very incomplete and vague
2. Then, it can be made more precise by a step-by-step process of refinement, until a complete description of the needed complex action is obtained
3. Then the agent considers alternative ways for achieving the objective
4. Finally, the agent must choose the preferred alternative, which it intends to pursue

Plans (2)

1. Achievable objective, given the constraints: $\mathsf{B}_a \overrightarrow{\diamond}(\theta \wedge \overrightarrow{\diamond}\sigma)$

2. Step-by-step refinement: $\diamond(\mathsf{B}_a\mathsf{C}_a^{\alpha|\theta}\sigma \wedge \bigcirc\mathsf{B}_a\mathsf{C}_a^{\beta|\theta}\sigma \wedge \beta \Rightarrow \alpha)$

3. Alternative ways (individually or collectively): $\mathsf{B}_a\mathsf{C}_a^{\alpha|\theta}\sigma \wedge \mathsf{B}_a\mathsf{C}_a^{\beta|\theta}\sigma \wedge \mathsf{B}_a(\alpha \not\Rightarrow \beta \wedge \beta \not\Rightarrow \alpha)$ or $\mathsf{B}_a\mathsf{C}_a^{\alpha|\theta}\sigma \wedge \mathsf{B}_a\mathsf{c}\mathsf{C}_a^{\beta|\theta}\sigma$

4. Chosen alternative: $\mathsf{I}_a^{\alpha|\theta}\sigma \rightarrow \mathsf{G}_a\sigma \wedge \mathsf{B}_a(\mathsf{C}_a^{\alpha|\theta}\sigma \vee \mathsf{c}\mathsf{C}_a^{\alpha|\theta}\sigma) \wedge \alpha \neq \Lambda$

Note that the condition $\alpha \neq \Lambda$ guarantees that a has a strategy, it cannot have a totally incomplete plan (Λ stands for "whichever action")

Delegation and adoption

- Delegation ($\mathsf{del}(b,\phi,\beta)$) is considered as a basic action type
- When a performs a delegation action ($\mathsf{E}_a^{\mathsf{del}(b,\phi,\beta)}\top$), it delegates not only a sub-goal, but also a way of achieving it
- The final goal of a delegation action can be the execution of that action: $\mathsf{E}_a^{\mathsf{del}(b,\phi,\beta)}\mathsf{E}_b^{\beta}\phi$
- An intermediate goal of a can be the adoption of a sub-goal by b: $\mathsf{E}_a^{\mathsf{del}(b,\phi,\beta)}\mathsf{I}_b^{\beta}\phi$
- Finally, if we consider also the basic action type adoption ($\mathsf{adp}(a,\phi,\beta)$), the intermediate goal can be that of making the other agent adopt the subgoal: $\mathsf{E}_a^{\mathsf{del}(b,\phi,\beta)}\mathsf{E}_b^{\mathsf{adp}(a,\phi,\beta)}\mathsf{E}_b^{\beta}\phi$

Foundational Ontologies and interoperability in e-gov

The scope of the problem of interoperability in PA

The problem of interoperability in PA applies to different ambits:

- ▷ Inside a single PA
- ▷ Among different PAs (f.i. among different offices that provide the same service)
- ▷ Between PAs and "external entities" (f.i. individual citizens or private firms)

Different levels of interoperability

The problem of interoperability can be articulated in different levels:

- ▷ Data integration (among different views and aspects of data)
- ▷ Workflow integration (between different processes and services)
- ▷ Organization integration (between services and organizations)
- ▷ Strategic integration (between the PA and the context in which it is active)

The former two are deeply studied in SOAs, while scarce attention has been dedicated so far to the two, equally important, latter

Semantic Interoperability in e-gov

- The problem: different organizations inside the PA often represent and manage information according to different *modalities*, with different *assumptions* w.r.t. the terms they use and utilizing different *tools* and *languages*
- The challenge: **semantic interoperability**: it amounts to making information available and transmissible while keeping its meaning intact and maximally reducing mistakes and misunderstandings
- Given the heterogeneity of actors and organizational structures involved in the various processes and services, the task is especially challenging and the required competences cannot be limited only to technological ones

Consortia for e-gov

- Many attempts have gone in the direction of the formation of consortia of PAs in which each PA, that has already implemented e-gov services shares its experience with the others
- The idea is that from this comparison *best practices* should be singled out
- These best practices should then be adopted as standards and reused by all the PAs that participate to a consortium
- Already at this stage, semantically based technologies show their usefulness, but…

Why consensus is not enough

- ▷ The "strategy of consensus", even though necessary and desirable, is not sufficient on its own to solve all the problems, being it too centered on technological applications
- ▷ When information is transmitted between different services it is not enough to have specific standard ontologies which describe the tasks composing the service
- ▷ They are built on radically different assumptions!

Foundational ontologies: an alternative approach (1)

- Solution at an higher level, *methodological*
- Basis: foundational ontology, general theory characterizing properties and relations fundamental for every ontological analysis, independently of the specific application
- Then: taking the foundational ontology and, by relativizing properties and relations to the domain at hand, building the ontology for the e-gov
- The focus is on singling out *basic semantic primitives* for the description of processes, which are understandable and the result of an accurate analysis of the domain to be represented

Foundational ontologies: an alternative approach (2)

- The issue is not to build a standard model valid for everyone
- Rather, it is to create first of all the conditions for the understanding and integration of the different models
- What has to be shared is not the whole model, but the primitives which constitute such descriptions, i.e. categories and relations that allow to understand and exchange such (different) descriptions
- An ontology for e-gov should include an ontology of services and an ontology of organizations, as often the quality of a service emerges from the relation between the entities of these two domains

Why an ontology of organizations

- Public Administration is constituted by structured institutions, that can be seen as particular instances of organizations
- Abstracting from specific institutions and providing a general model of organization is important because the PA interacts with collective entities of a varied nature, as private firms (ex. outsourcing)
- It is thus important to represent in the same model public institutions and private firms as specialization of the same kind of entity
- Modeling should not then be limited to the structure of the PA, but include the whole social and organizational context in which it is immersed

Which ontology of services

- The services provided by the PA are numerous and multi-faceted, sometimes very different one from the other
- Nonetheless, they share common properties that identify them as services (for instance, the fact of having someone who's responsible for their execution, a documentation to provide etc.)
- It is important not to limit the model to the representation of the workflow, but to represent in a rigorous way the *content* of services and, through it, the connection between different services as to facilitate the semantic-based access to services

Why it is important to integrate organizations and services (1)

- As for other organizations, the relations between the entities composing the PA are often regulated by norms
- The particular structure that the institution/organization assumes is a consequence of the division of tasks and functions made with the purpose of achieving the goal of the organization (for the PA, providing certain services)

Why it is important to integrate organizations and services (2)

▷ Two objectives:

- to understand the links between services provided by related institutions → understanding of the PA's internal procedures
- to analyze the relations among institutions which provide interconnected services → understanding the distribution of responsibilities

▷ Complementary analyses!

Summary of the course

Lecture 1: Introduction to knowledge representation, conceptual modeling, and semantic interoperability

- ▷ The importance of subtle distinctions of meaning
- ▷ A common alphabet is not enough!
- ▷ Importance and limitation of standard glossaries
- ▷ FIRST ontological analysis, THEN knowledge representation
- ▷ Content, independently on how it is represented

Lecture 1: Introduction to knowledge representation, conceptual modeling, and semantic interoperability

- The key problems: semantic matching and semantic integration
- The distinction between ontology, lexicon and semantics
- The distinction between meaning and sign
- The difference between extension and intension

Lecture 1: Introduction to knowledge representation, conceptual modeling, and semantic interoperability

- ▷ Concepts, properties and relations
- ▷ Conceptualization
- ▷ Adequacy of KR formalism:
  - expressive
  - inferential
  - cognitive
  - ontological

Lecture 1: Introduction to knowledge representation, conceptual modeling, and semantic interoperability

▷ Representation levels:

- logical
- epistemological
- conceptual
- linguistic

▷ Ontology helps in making a conceptualization explicit

▷ Ontology and ontologies

Lecture 2: Introduction to First Order Logic

▷ Formal systems: consistency and completeness

▷ Formal systems as tools for explicit representation

▷ Advantages of formal systems:

- automaticity
- testability
- reliability in sharing knowledge

Lecture 2: Introduction to First Order Logic

- ▷ To prove vs. to be true (syntax vs. semantics)
- ▷ Logic gives precise and unambiguous languages
- ▷ Logical and non logical symbols
- ▷ Terms and formulas (open and closed), sentences

Lecture 2: Introduction to First Order Logic

- ▷ Interpretations
- ▷ Truth values
- ▷ To be true in a structure and in an interpretation
- ▷ Validity
- ▷ Logical equivalence and logical consequence

Lecture 2: Introduction to First Order Logic

- Proof theory
- Rules and deduction
- Consistency
- Soundness and Completeness

Lecture 3: Introduction to Knowledge Representation and Conceptual Modeling

- Informal representation
- Informal model
- Set theoretical model
- Relational data model (also with set theoretical view)

Lecture 3: Introduction to Knowledge Representation and Conceptual Modeling

- ▷ Modeling in FOL
- ▷ The relation "instance-of"
- ▷ Comparisons of theories and of models
- ▷ Formal mappings between theories
- ▷ Representing extension and intension

Lecture 3: Introduction to Knowledge Representation and Conceptual Modeling

▷ Relations in relational data models and relational schemes

▷ Keys

▷ Entity-Relationship diagrams

- ISA relations
- Roles
- Attributes/Keys

▷ Reasoning, knowledge engineering and evaluation of theories and models

Lecture 4: Concepts and Ontologies

- ▷ Kinds, roles, attributes
- ▷ Measuring the quality of an ontology: precision and coverage
- ▷ Levels of ontological precision (ranging from catalogs to axiomatic theories)
- ▷ Kinds of ontology changes:
  - reality changes
  - perception system changes
  - conceptualization changes
  - logical characterization changes

Lecture 4: Concepts and Ontologies

▷ Ontolgies vs.:

- Taxonomies
- Classifications
- Database schemas
- Knowledge Bases

▷ Ontology-driven information systems:

- Development time vs. run time
- Database, user-interface and application program components

▷ The task dependency problem

▷ The semantic web

Lecture 5: Introduction to Modal Logic

- ▷ Propositional logic
- ▷ Validity, Satisfiability, Contingency
- ▷ Possible worlds
- ▷ Propositional modal logic
- ▷ Modalities for FOL

Lecture 6: The OntoClean Methodology

- ▷ Essential properties for an individual and a type
- ▷ Essential properties and rigidity
- ▷ Identity and identity criteria
- ▷ Sortals and non sortals
- ▷ Dependence (existential vs. definitional)
- ▷ Unity and plurality

Lecture 7: Orders and Mereologies

▷ Formal ontology as rigorous and foundational

▷ Formal ontology for reuse, modularity and comparisons between theories

▷ Orders:

- partial orders
- strict partial orders
- total/linear orders
- dense and discrete orders
- right and left bounded orders
- right and left unbounded orders

Lecture 7: Orders and Mereologies

▷ The importance of the parthood relation

▷ Mereologies:

- Basic
- Minimal
- Extensional
- Closure
- Atomic

▷ Space of mereologies and libraries of theories

Lecture 8: Theories of Time, Space and Qualities

- ▷ The importance of time
- ▷ Theories of time:
    - Instant theories
    - Period theories
        - Van Benthem's
        - Allen's
    - Event theories (Kamp's)
- ▷ Comparing theories

Lecture 8: Theories of Time, Space and Qualities

▷ The space of ontological choices, two dimensions:

- vision (basic ontological choices)
- specificity (domain)

▷ Space of regions and connection

▷ Families of mereotopologies

▷ Relative space and spacetime

Lecture 8: Theories of Time, Space and Qualities

▷ Location (exact and broad)

- partial
- whole
- general
- tangential

▷ Co-localization (spatial, temporal and spatio-temporal)

▷ Constitution

▷ Qualities vs. Features

▷ Qualities, Qualia and Quality regions

Lecture 9: Roles and Organizations

▷ Social reality

▷ Roles in:

- sociology
- linguistics
- object oriented programming and conceptual modeling

▷ Relational, Processual and Social roles

Lecture 9: Roles and Organizations

- Roles are properties
- Roles are dynamic and anti-rigid
- Roles have a relational nature
- Roles are determined by contexts

Lecture 9: Roles and Organizations

▷ Philosophical and foundational analysis of organizations

▷ Organizations as:

- structured and multilayered
- designed
- agentive
- realized
- situated
- dynamic