Engineering an Ontology for a Multi-Agents Corporate Memory System

Fabien Gandon

Engineering an Ontology for a Multi-Agents Corporate Memory System

Fabien Gandon

ACACIA Project - INRIA, 2004, route des Lucioles, B.P. 93
06902 Sophia Antipolis, France - Fabien.Gandon@sophia.inria.fr
Fabien.Gandon@sophia.inria.fr

Abstract:

XML technology and multi-agents systems are new approaches offering interesting assets for corporate memory management. Since ontologies appear as key assets in the new generation of information systems and also in the communication of multi-agents systems, it comes with no surprise that it stands out as a keystone of multi-agents information systems. We present here the overall approach of the CoMMA project, and then focus on the first elements of our return on experience in building the ontology O'CoMMA used by this multi-agent corporate memory system.

With our entrance in the information society and the associated shift in economical rules of game, organizations had to adapt and their interest in corporate knowledge capitalization and management grew stronger. The semantic web technologies provide interesting techniques to materialize and structure memories to prepare their exploitation and management. At the same time, distributed artificial intelligence proposes appropriate paradigms, especially the multi-agents one, to deploy a software architecture over this distributed information landscape. Through intelligent collaboration agents can achieve a global capitalization of the corporate knowledge while being able to locally adapt to individual resources and users specificity. In a first part we present the overall approach of CoMMA, and we then focus on the methodology and our return on experience in building an ontology for such a system.

1 Overall approach

1.1 Organisational memory

The last decade information technology explosion led to a shift in the economy and market rules forcing corporations to adapt their organization and management in order to improve their reaction and adaptation time. Information systems became backbones of organizations enabling project-oriented management and virtual teams, therefore the industrial interest in methodologies and tools enabling capitalization and management of corporate knowledge grew stronger.

A corporate memory is an explicit, disembodied and persistent representation of knowledge and information in an organization, in order to facilitate their access and reuse by members of the organization, for their tasks [Rabarijaona *et al.,* 2000]. The stake in building a corporate memory management system is the coherent integration of this dispersed knowledge in a corporation with the objective to promote knowledge growth, promote knowledge communication and in general preserve knowledge within an organization [Steels, 1993].

ACACIA, our research team, is part of the CoMMA project (IST-1999-12217) funded by the European Commission, aiming at implementing a corporate memory management framework based on several emerging technologies: agents, ontology engineering and knowledge modeling, XML, information retrieval and machine learning techniques [CoMMA, 2000]. The project intends to implement this system in the context of two scenarios (1) assisting the insertion of new employees in the company and (2) supporting the technology monitoring process. The technical choices of CoMMA are mainly motivated by three observations:

(1) The corporate memory is, by nature, an heterogeneous and distributed information landscape. The corporate memories are now facing the same problem of information retrieval and information overload than the Web. To quote John Naisbitt, "we are drowning in information but starved of knowledge". The initiative of a semantic Web [Berners-Lee *et al.,* 2001] is a promising approach where the semantics of documents is made explicit through metadata and annotations to guide later exploitation. Ontobroker [Decker *et al.,* 1999], Shoe [Heflin *et al.,* 1999], WebKB [Martin and Eklund, 1999] and OSIRIX [Rabarijaona *et al.,* 2000] are examples of this technique, relying on annotation based on ontologies. XML being likely to become an industry standard for exchanging data, we use it to build and structure the corporate memory. We are especially interested in RDF, the Resource Description Framework defined in XML, that allows us to semantically annotate the resources of the memory. The corporate memory can then be studied as a "corporate semantic Web".

(2) The population of the users of the memory is, by nature, heterogeneous and distributed in the corporation. Agents will also be in charge of interfacing users with the system. Adaptation and customization are a keystone here and CoMMA relies on machine learning techniques in order to make agents adaptive to users and context. This goes from basic customization to user's habits and learning of preferences, up to push technologies based on interest groups and collaborative filtering.

(3) The tasks, as a whole, to be performed on the corporate memory are, by nature, distributed and heterogeneous. The corporate memory is distributed and heterogeneous. The user population is distributed and heterogeneous. Therefore, it seems interesting that the interface between these two worlds be itself heterogeneous and distributed. As noted in [Wooldridge *et al.,* 1999], programming progresses were achieved through higher abstraction enabling us to model systems more and more complex. Like them, we believe that multi-agents systems (MAS) are a new stage in abstraction that can be used to understand, to model and to develop a whole new class of distributed systems. The MAS paradigm appears to be suited for the deployment of a software architecture above the distributed information landscape of the corporate memory. On the one hand, individual agents locally adapt to users and resources they are dedicated to. On the other hand, thanks to cooperating software agents distributed over the network we can capitalize an integrated and global view of the corporate memory.

1.2 Agents in an annotated memory

We will not discuss the definition of an agent here. In CoMMA we use the weak notion of agency [Wooldridge and Jennings, 1995]. We do not claim that all our agents are currently one hundred percents compatible with this definition, but it is this definition that we use to consider what could be an agent and what will have to be something else. The information agents are part of the 'intelligent agents', a notion nicely commented by Lieberman [1999]. A Multi-Agents System (MAS) can be defined as a loosely coupled network of agents that work together as a society aiming at solving problems that would generally be beyond the reach of any individual agent. Such a system is said to be heterogeneous when it includes agents from at least two or more agent classes. A Multi-Agents Information System (MAIS) is then defined as a MAS aiming at providing some or full range of functionality for managing and exploiting information resources. The application of

MAIS to corporate memories means that agents' cooperation aims at enhancing information capitalization in the company. Based on these notions we define the CoMMA software architecture as an heterogeneous MAIS.

Unlike a lot of other MAIS projects we do not stress the heterogeneous sources reconciliation aspect: documents are heterogeneous but annotations are in RDF and based on a shared ontology. We are focusing on the design of an architecture of cooperating agents, being able to adapt to the user, to the context, and supporting information distribution. The duality of the definition of the word 'distribution' reveals two important problems to be addressed : (1) Distribution means 'dispersion', that is the spatial property of being scattered about, over an area or a volume ; the problem here is to handle the naturally distributed data, information or knowledge of the organization. (2) Distribution also means the act of 'distributing or spreading or apportioning' ; the problem then is how to make the relevant pieces of information go to the concerned agent (artificial or human).

Figure 1 shows an overview of the architecture of CoMMA. Agents are able to communicate with the others to delegate tasks, and to make elementary reasoning and decisions. They have inference mechanisms exploiting ontologies. They help authors of documents to annotate the documents, to diffuse the acquired innovative ideas to the interested employees of the company or proactively suggest to newcomers the information essential for their integration.

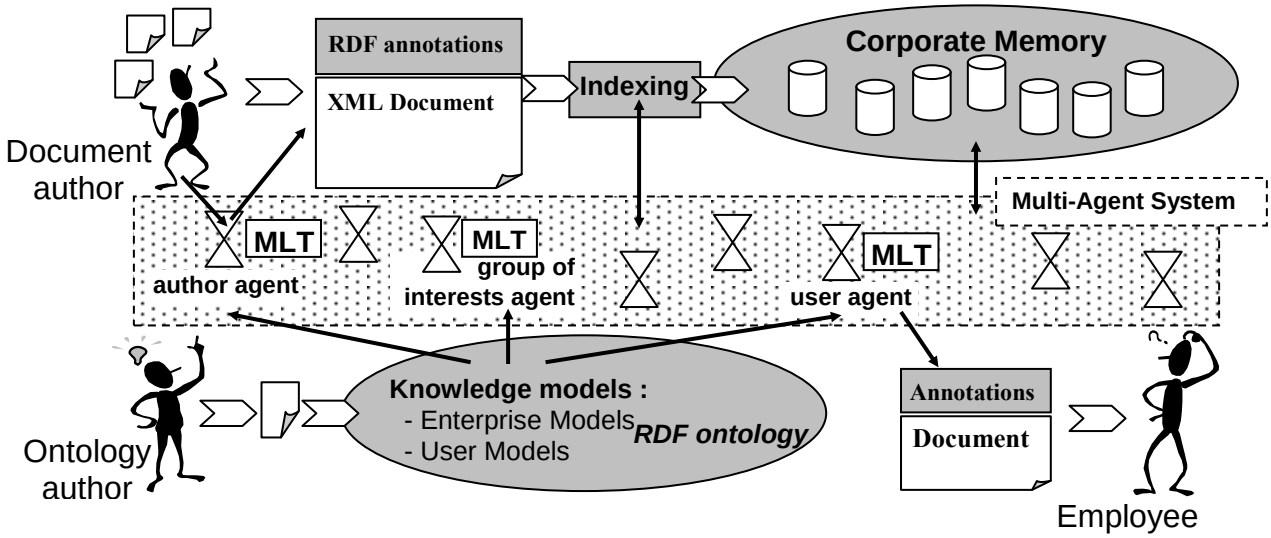


Figure 1 - Overview of the architecture of CoMMA

We identified four sub-societies of agents. A detailed presentation of these societies is out of the scope of this article and more can be found on that subject in [Gandon *et al.,* 2000]. However we would like to stress the pivotal role of the ontology agent sub-society, that provides a common context as a semantic grounding that is vital for agents interoperation [Singh and Huhns, 1999]. These agents provide downloads, updates and querying mechanisms of the ontologies for other agents. They provide, for instance, the user agents with the ontological elements needed for query elicitation and the mediators or resources agents with the ontological elements needed for query solving. When the system handles several ontologies, ontology agents may be in charge of the mapping and translation between ontologies using, for instance, mappings to a common ontology. When the system handles different points of view, they enable other agents to use them to filter their view/access to the ontology. When there exists a terminological level, additional services such as queries on terms and synonyms for a given concept may be part of their job.

The ontology is vital to the agent communication and interaction protocols. From these interactions, emerge at the societal level the functionalities meeting users requirements.

In their article about "Agents in Annotated Worlds" Doyle and Hayes-Roth [1998] explained that software agents must have the ability to acquire useful semantic information from the context of the world they evolve in. They introduced the notion of "annotated environments containing explanations of the purpose and uses of spaces and activities that allow agents to quickly become intelligent actors in those spaces". Their remark is transposable to information agents in complex information worlds, and this leads us to say that annotated information worlds are, in the actual state of the art, a quick way to make information agents smarter. If the corporate memory becomes an annotated world, agents can use the semantics of the annotation and through inferences help the users exploit the corporate memory. RDF uses a simple data model expressed in XML syntax to represent properties of Web resources and their relationships. It makes no assumptions about a particular application domain. With RDF, we describe the content of documents through semantic annotations and then use and infer from these annotations to successfully search the mass of information of the corporate memory.

Just as an important feature of new software systems is the ability to integrate legacy systems, an important feature of a corporate memory management framework would be the ability to integrate the legacy archives. An RDF annotation is either internal or external to the document, therefore existing documents of the corporate memory may be kept intact and annotated externally. The annotations are based on an ontology and this ontology can be described and shared thanks to RDF Schema. RDF Schema is related to object models (Classes, Properties, Specialization,…), however properties are defined independently from classes.

In order to manipulate the ontology, the annotations, and infer on them, we developed CORESE a prototype of a search engine enabling inferences on RDF annotations by translating the RDF triplets to Conceptual Graphs and vice versa [Corby *et al.,* 2000]. CORESE combines the advantages of using the standard RDF language for expressing and exchanging metadata, and the query and inference mechanisms available in CG formalism. Modules of CORESE are integrated in the ontology, models and documents dedicated agents to provide the abilities needed for their roles.

1.3 Model-based annotations

Compared to the Web, a corporate memory has a more delimited and defined context, infrastructure and scope: the corporation. In the corporate context we can more precisely identify the stakeholders (e.g.: information providers, information seekers) and moreover the corporate community shares some common global views of the world (e.g.: company policy, best practices) and thus an ontological commitment is conceivable to a certain extent. Based on this ontology we describe the organizational state of affairs and the users' profile. This process is illustrated in Figure 2.

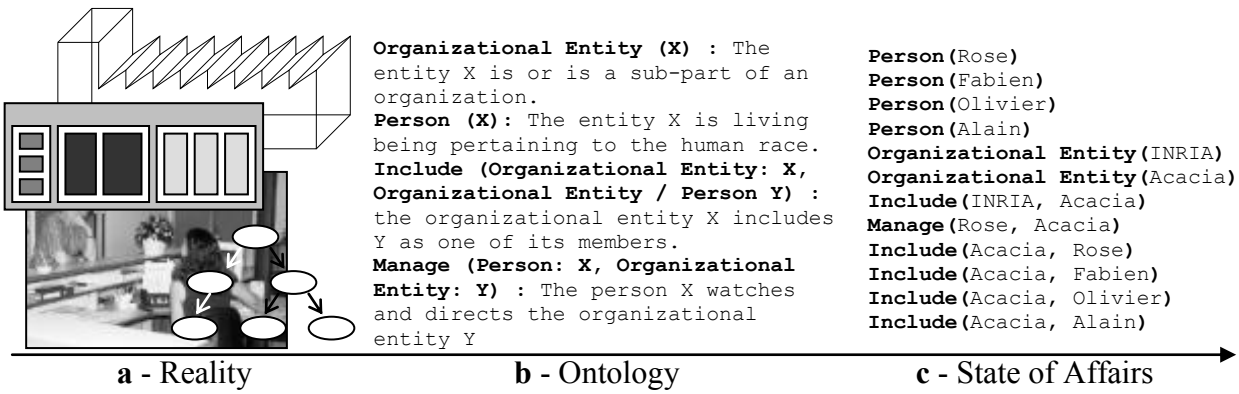
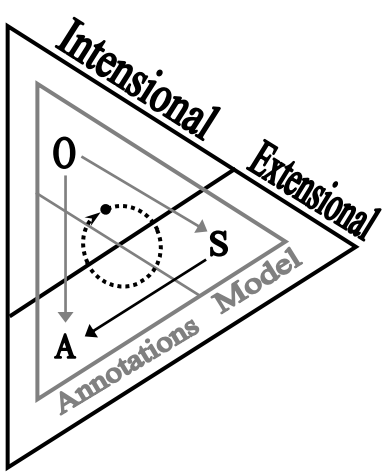


Figure 2 - Ontology-based organizational modeling

An enterprise model is an oriented, focused and somewhat simplified explicit representation of the organization. The level of details/simplification, just like the points of view adopted, depends on the specifications of the system (computerized or not) exploiting the formal model, and therefore it ultimately depends on the stakeholders' expectation. So far, the enterprise modeling field has been mainly concerned with simulation and optimization of the production system design. They provide benchmark for business processes and are used for re-engineering them. But the new trends and the shift in the market rules led enterprises to become aware of the value of their memory and the fact that enterprise model has a role to play in this application too [Rolstadås, 2000]. In CoMMA, the model we envisage aims at supporting corporate memory activities involved in the new employee scenario and the technology monitoring scenario. Our use of this explicit partial representation of reality is to enable the CoMMA system to get insight in the organizational context and environment and to intelligently exploit the aspects described in this model for the interaction between agents and overall between agents and users. Papazoglou and Heuvel [1999] explained that "It is necessary to have an understanding of the organizational environment, its goals and policies, so that the resulting systems will work effectively together with human agents to achieve a common objective." Many formal languages exist to model enterprises, see, for instance, the comprehensive overview of Gastinger and Szegheo [2000]. The methodology IDEF5 in the beginning of the 90s proposed to develop ontologies for the enterprise modeling. In CoMMA, to benefit from the XML standard technology assets, we use the RDF Schema and RDF language to describe our ontology and implement our organizational description. This choice enables us to base our system on a standard that benefits from all the web based technologies for networking, display and navigation, and this is an asset for the integration to a corporate intranet environment.

Likewise, the users' profile captures all aspects of the user that were identified as relevant for the system behavior. It contains administrative information and directly made explicit preferences that go from interface customization to topic interests. It also positions the user in the organization: role, location and potential acquaintance network. In addition to explicitly stated information, the system will derive information from the usage made by the user. It will collect the history of visited documents and possible feedback from the user, as well as the user's recurrent queries, failed queries, and from this it can learn some of the user's habits and preferences. These derived criterions can then be used for interface purposes or push technology. Finally the profiles enable to compare users, to cluster them based on similarities in their profiles and then use the similar profiles to make suggestions. Figure 3 shows the modeling framework we adopted in CoMMA.



Ontology : captures the conceptual vocabulary and is formalized in RDFS

State of affairs : captures organizational and users' descriptions and is formalized in RDF

Annotations : describe the corporate memory resources and is formalized in RDF

Instantiation of the ontology
Reference to the state of affairs
Interdependency prototype life cycle

Figure 3- OSA Schema

We will comment this schema detailing the different stages of our approach:

1. We apply techniques from knowledge engineering for data collection to provide the conceptual vocabulary detected as needed in the scenarios. We specify the corporate memory concepts and their relationships in an ontology and formalize the ontology in RDF using the RDF Schema.
2. We use the ontology and the results from data collection to propose enterprise and user models to describe the organizational State of affair. The models are implemented in RDF instantiating the RDFS ontology description.
3. We structure the corporate memory writing RDF annotations of the documents instantiating the RDFS ontology description and referencing the state of affair.
4. These annotations, the model and the ontology are used through inferences to search, manage and browse the memory.

It is really the union of the ontology and the state of affairs that forms the model. The archiving structure will depend on both and that is why we say that the memory is said to be a model-based information system. However both the state of affairs and the annotations are instances of the schema and implemented as RDF annotations. That is why the Ontology is considered to be at the intensional level whereas the state of affairs and the annotations are at the extensional level. The ontology, the state of affairs and the annotations are tightly linked and will evolve in a prototype life cycle style. The ontology and the state of affairs capture the modeling on which the inferences will be based.

Actual keyword-based search engines such as the ones used for web searching are limited to the terms denoting extensions of concepts, the introduction of ontologies frees us from this restriction by enabling agents to reason at the intensional level. In CoMMA, the ontology is the keystone of our system since it provides the building blocks for models, annotations and agent messages, with their associated semantics. The next section details the design process of O'CoMMA (Ontology of CoMMA)

2 Engineering O'CoMMA

Our work on ontology has been influenced by the return on experience and the analysis on TOVE and the Enterprise Ontology done by Uschold and Gruninger [1996], the comparison and elements of methodology presented in [Fernandez *et al.,* 1997] and [Gómez-Pérez *et al.,* 1996] and the seminal works done on theoretical foundations of ontologies by Bachimont [2000], Guarino and Welty [Guarino, 1992; Guarino and Welty, 2000]. We present in the following parts our approach, the first elements of our return on experience, and our expectations and goals for the evolution of the ontology engineering field.

2.1 Position and definitions

Ironically, the ontology field suffered a lot from ambiguity, therefore, before we go any further, we will state the definitions and the points of view we adopted in the project. The system exploit an organizational state of affairs, that is a system-relevant description of the general state of things and the combination of circumstances in an organization. To do so, we rely on an ontology to define the primitives required for the representation and to provide their semantic. Guarino and Giaretta [1995] defined an ontology as "a logical theory which gives an explicit, partial account of a conceptualization", the conceptualization being defined as "an intensional semantic structure which encodes the implicit rules constraining the structure of a piece of reality". The ontology is a partial

explicit representation because it focuses on those aspects of the conceptualization that are critical for the behavior of the application.

A concept is a constituent of thought formed in mind (an idea, a notion, a principle) and semantically evaluable and redeployable. The set of attributes characterizing a concept is called its intension and the collection of things to which the concept applies is called its extension. There exist a duality between intension and extension: to two included intensions $I_1 \subset I_2$ correspond two included extensions $E_1 \supset E_2$. An intension is determined by identifying the qualitative or functional properties shared by all the entities the concept applies to. The set of characteristics provides a definition. In order to express and communicate an intension we choose a symbolic representation, usually verbal, e.g.: the different notions associated to a term and given by a dictionary. Note that exemplification and illustration used in dictionaries show that it is sometime necessary to clarify a definition in natural language, to produce, respectively, a representative sample of the extension (i.e. examples) or to use other means of representation.

The representations of intensions can be organized, structured and constrained to express a logical theory accounting for relations existing between concepts. An ontology is an object capturing the expressions of intensions and the theory accounting for the aspects of reality selected for their relevance in the envisaged application scenarios. The representation of the intensions and the ontological structure can make use of more or less formal languages, depending on the intended use of the ontology. The formal expression of an intension provides a precise and unambiguous representation of the meaning of a concept, and it allows to manipulate it in software and a use it as a primitive for knowledge representation in models and annotations. Since the expression of an intension nearly always starts from a natural language definition, "defining an ontology is a modeling task based on the linguistic expression of knowledge" [Bachimont, 2000]. Through iterative refinements, we augment the ontology, developing the formal counter parts of semantic aspects relevant for the system in the application scenarios. In the ontology, concepts in intension are usually organized in a taxonomy, that is, a classification based on their similarities.

When a group of person agree on the use and the theory specified in the ontology, we say they have made an ontological commitment. Ontology engineering deals with the practical aspects, essentially methodologies and tools, of applying results from the Ontology science to build ontologies needed in a specific context and for a specific purpose.

2.2 Scenario analysis and Data collection

Scenarios are textual descriptions of the organizational activities and interactions concerning the intended application. Following Caroll [1997] we used scenarios to capture in a natural and efficient way the end-users' needs in their context. This is vital for a symbiotic integration of the system in the work environment. The main advantages we recognized when using scenarios for CoMMA are:

- They enable us to focus on the specific aspects of knowledge management involved in our case;
- They capture the whole picture and enable us to view the system as a component of a possible knowledge management solution for a company;
- They represent a concrete set of interaction sequences with the corporate memory which is understandable and accessible to all the stakeholders and which is a perfect start to build formal use cases, and interaction diagrams;
- They provide a framework to check up on every new idea, every contribution.

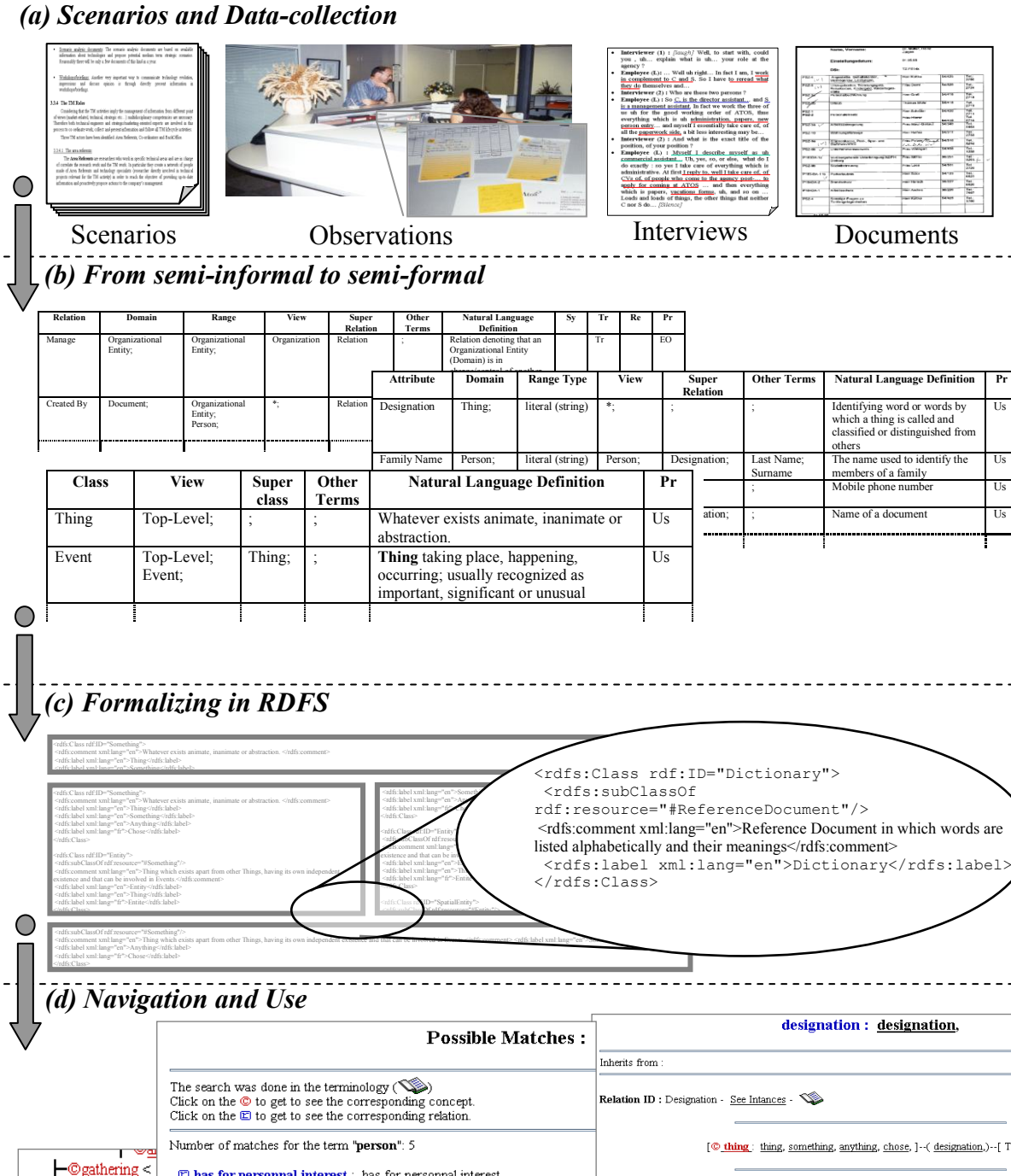


Figure 4- Snapshot of the ontology engineering process

Scenario analysis led us to define a table suggesting key aspects to be considered when describing a scenario. It provides suggestions as for what are the aspects to be investigated, what to look for and what could be refined when describing a scenario. It helps define the scope of our intervention and thus the scope of the ontology: the conceptual vocabulary must provide the expressivity required by the interactions described in the scenarios.

Scenario analysis produces traces, in our case: scenario reports. These reports are extremely rich story-telling documents that are good candidates to be included in the corpus of a terminological study and therefore they were our first step in data collection.

Several techniques exist for data collection that feeds the whole modeling process. We essentially used three of these techniques: semi-structured interview, observation and document analysis.

Semi-structured interview were in three steps: (a) A free opening discussion where the dialog is initiated with broad questions usually concerning the tasks and roles of the people interviewed. If needed this spontaneous expression can be kept running using short questions. (b) Flashbacks and clarifications of the first part with also additional specific questions prepared before the interview. (c) A self-synthesis to make interviewees synthesize, summarize and analyze themselves what they said and make them conclude. The generalization and grouping they make in this last part are especially interesting for the ontology structuring.

We carried out an interview with a newcomer at ATOS (industrial partner of the consortium). The interview showed for instance that the definition she had of her role and position was different from what was stated in the official organization chart ; this emphasizes the importance of having and exploiting user profiles in a solution. We also discussed tasks linked to documents (e.g. to issue an order) and the importance of the acquaintance network in her day-to-day activity (use of the organizational model).

Another data-collection technique we applied is the *observation*. The observation can be about people, the way they work (with or without making them comment), on a real task or on a simulated scenario, in real time, recorded or based upon traces of the activity. It can also be focused on chosen indicators (documents manipulated, desk organization…). Depending on the actor and the scenario the interesting situation may be very different (a newcomer looking for information, a mentor explaining, a technology an observer commenting…).

We observed the desk of the newcomer we interviewed before because a lot of documents in the company go through her. It revealed, for instance, that she was using two criteria to label the files: the type of documents (e.g.: vacation forms) and the process to be done on these documents (e.g.: to be signed). Another observation showed different types and use of annotations on documents (targeted people, deadline, how to fill and uses a form). Finally we also noticed that there exist documents (in our case it was the company name card with the phone number and the fax number) that she doesn't want to sort, and put away with others (phone book, address book) because when needed she wants to access them at the first glance or if you prefer "at the first click". These observations help understand what types of vocabulary is needed for annotation.

The last type of data-collection techniques we used is the *document collection and analysis*. The gathering of typical documents is a vital observation in our case, since the project focuses on information retrieval in a corporate documentary memory. It consists of the systematic collection of documents involved in and relevant for the considered scenarios. It includes lexicon, terminology, reports, etc. and also graphical documents (forms, organization charts...). A good practice is to collect both empty and filled forms to see their use, and also to 'follow document' to discover their track in the organization. An example is the new employee route card of one of our partners: it describes what to do, where to go, who is the contact, how to contact the person, and what is the

order of the tasks to be carried out by a new employee arriving. This document gives an idea of the stakeholders involved in the scenario, and it reveals the existence of a process of integration and the vocabulary needed to describe the whole state of affair.

2.3 Reusing ontologies and other sources of expertise

These first data collection methods are extremely time-consuming and their use must therefore be limited. To speed-up the process we also decided to study and *reuse existing ontologies* whenever possible:

- Enterprise Ontology [Uschold *et al.,* 1998]
- TOVE Ontology [TOVE, 2000]
- Upper Cyc® Ontology [Cyc, 2000]
- PME Ontology [Kassel *et al.,* 2000]
- CGKAT & WebKB Ontology [Martin and Eklund, 2000]

The reuse of ontologies is both seductive (it should save time, efforts and would favor standardization) and difficult (commitments and conceptualizations have to be aligned between the reused ontology and the desired ontology). These ontologies have not been imported directly or translated automatically. We found that the best way to reuse them was to analyze their informal version as a natural language document that would have been collected. Divergences in the objectives and the contexts of modeling and use of these ontologies and our ontology (end-users and application scenarios are indeed different) led us to think that no automatic import was possible and that human supervision was compulsory. However natural language processing tools, such as the ones used in approaches described by [Aussenac-Gilles *et al.,* 2000], could have helped the analysis and translator between formal languages could have ease reuse.

We enriched these different contributions by considering other informal sources. Some very general helped us structure upper parts of some branches. For instance thoughts from the book “Using Language” from Herbert H. Clark were used for structuring the document branch on representation systems, providing an expertise on semiotic that was needed and could not be gained through previous data collection techniques (interviewed stakeholders are not expert in semiotics, and collected documents do not explicitly contain that information either). Other very specific standards enabled us to save time on enumerating some leaves of the taxonomic tree. For instance the MIME standard was an excellent input for electronic format description and the Dublin Core suggested most common properties global of documents.

The systematic use of dictionaries or available lexicons is good practice. In particular, the meta-dictionaries have proved to be extremely useful. They give access to a lot of dictionaries (some of them specialized in specific field e.g. economy, technology...) and therefore enable us to compare definitions and identify or build the one that correspond to the notion we want to introduce. We make extensive use of the meta-dictionary [OneLook, 2000] that enabled us to produce the first expressions of intensions in English.

Reused sources have to be pruned before being integrated in the ontology, and scenarios can be used for that purpose. They capture the scope of the intervention and a shared vision of the stakeholders, thus they can help decide whether or not a concept is relevant. For instance, the ownership relation introduced in the Enterprise Ontology was not reused in our Ontology because this relation appears not to be exploited in our scenarios.

2.4 Terminological stage

The candidate terms denoting concepts that appears relevant in the application scenarios are candidate for the terminological analysis. Synonymous are selected too, and related terms are considered in a chain reaction. For instance, if we consider the terms *document*, *report*, and *technological trend analysis report* implicated in the technology monitoring scenarios their candidacies to terminological analysis are linked. The starting point can be the term *document* du to a study of the existing top ontologies, or the term *report* identified during the interview of the ATOS newcomer or finally the term *technological trend analysis report* encountered when collecting documents for the technology monitoring scenario (ex: "technological trend analysis report entitled 'Capitalizing WAP experiences for UMTS transition' "). Candidate terms are organized in a set of informal tables which is a semi-informal data collection structure. ACACIA proposed definitions in natural language for each term. This first terminology is presented to members of the consortium and low-level extensions (terms and definitions) are proposed by the industrial partners, for instance:

> *"**Area Referent** : Observer responsible for an expertise area and who has a group of contributors observers to manage."*

The interest of having a continuous collaboration between a knowledge engineer for methodological aspects and bootstrapping of the ontology, and stakeholders for specific concept and validation is flagrant.

The terminological study is at the heart of ontology engineering and the main work on identified terms is the production of consensual definitions expressing the intension of each concept. There are three cases when studying terms and notions:

- One term corresponding to one and only one notion: we label the notion with the term.
- Several terms corresponding to one notion: the terms are synonyms, we keep the list of synonyms and choose the most commonly used term to label that notion.
- One term corresponding to several notions: the term is kept, but, noted as ambiguous and several expressions of intensions are defined with non ambiguous labels (e.g. compound terms).

Labeling concepts with one term is both convenient and dangerous. It is a major source of 'ambiguity relapse' : when we unconsciously relapse in ambiguity using the label terms according to the definition we associate with it and not the definition actually associated to it in the process of semantic commitment. The explicit representation of the two levels (terms level and notion levels) and the existence of management functionality in tools assisting ontologist for the terminological aspects in ontology engineering and users in their interactions with the system are real needs.

2.5 Structuring the ontology

The *obtained concepts are structured in a taxonomy*. The principles behind this structure go back to Aristotle who defined a specie by giving its genus (genos) and its differentia (diaphora). The genus is the kind under which the species fall. The differentia is what characterizes the species within that genus. Thus we started regrouping concepts firstly in an intuitive way, then iteratively organizing and reviewing the structure following the extended Aristotelian principles given by Bachimont [2000]. These principles tend to eliminate multiple inheritance which is a problem with the role concepts making an extensive use of this mechanism. One idea would be to introduce multiple view points [Ribière, 1999] and limit the application of these extended principles to a point of view. An approach is proposed in [Kassel *et al.,* 2000] introducing semantic axis as means to group the types of criteria used for the differentia. The extended principles go then be applied to concepts inheriting for the same semantic axis. Likewise, the extensive work of Guarino and Welty [Guarino, 1992;

Guarino and Welty, 2000] contributes to clean-up the theoretical foundations of ontology engineering providing definitions, theoretical framework and constraints based on them to be verified by the taxonomy so that an ontologist relying on these definitions can check some validity aspects of his subsumption links. The only problem is that, as far as we know, no tool is available to help an ontologist do that work easily and independently of a formalization language; it is a titanic work to apply this theory to large ontologies. These contributions appeared to be adapted to validation of top ontologies ensuring, by extension, a minimal coherence in the rest of the ontology.

The way to design an ontology is still debated. Our understanding of the different contributions to the field is that there is a tendency to distinguish between three approaches that are the three common options when building an ontology:

- Bottom-Up: Where the ontology is built by low taxonomic level concepts and generalization. It is prone to provide tailored and specific ontologies.
- Top-Down: Where the ontology is built by top concepts and specialization. It is prone to the reuse of ontologies.
- Middle-Out: Where the priority is the identification of core concepts followed by the generalization and specialization to complete the ontology. It is prone to encourage emergence of thematic field and modularity.

Our first opinion was that it would be interesting to try a cross approach by merging Bottom-Up and Top-Down approaches to associate the asset of being specific and the ability to reuse other ontologies. After our experience, we are not convinced that there exists such a thing as a purely top-down, bottom-up or middle-out approach. It looks like they are the three aspects or complementary perspectives of a complete methodology. It seems to us that the activities of finding structure by specialization from a generic concept, or by generalization from a specific concept are concurrent processes present and at work at every levels of depth in the ontology (bottom, middle or tops) and at different detail grains (concepts or groups of concepts). It seems that the holistic nature of knowledge leads to the holistic nature of ontologies, and the holistic nature of ontologies leads to the holistic nature of methodologies to build them. We will not deny that for a given case, an approach can mainly rely on one perspective (e.g. some ontologies of chemical substances made extensive use of Bottom-Up approach), but we would not oppose here the different approaches, we would rather say that they represent three perspectives that are combined in ontology engineering and when engineering an ontology, an ontologist should have the tasks defined in these three perspectives on the go at one time.

In our case we can say that some tasks were performed in parallel in the different perspectives, for instance : we studied existing top-ontologies, and upper parts of relevant ontologies to structure our top part and recycle parts of existing taxonomies (top-down); we studied different branches, domains, micro-theories of existing ontologies as well as core subjects identified during data-collection to understand what were the main areas we needed and regroup candidate terms (middle-out); we exploited reports from scenario analysis and data-collection traces to list scenario specific concepts and then started to regroup them by generalization (bottom-up). The different buds (top concepts, core concepts, specific concepts) opening out in the different perspectives are the origins of partial sub-taxonomies. The objective then is to ensure the joint of the different approaches and each time an event occurs in one perspective it triggers checks and tasks in the others.

2.6 From semi-informal to semi-formal

Starting from the informal terminology we ended-up separating attributes and relations from the concepts and obtained three tables. These *tables evolved from a semi-informal representation (terminological tables term & notion) toward semi-formal representation (taxonomic links, signatures of relations)* with the following structure: (1) the label of the concept / relation / attribute, (2) the concepts linked by the relation or the concept and the basic type linked by the attribute, (3) the closest core concept or the thematic fields linked by the relation, (4) the inheritance links, (5) synonymous terms for the label, (6) a natural language definition to try to capture the intension, and (7) the collection source. This last column introduces the principle of traceability and it is interesting for the purpose of abstracting a methodology from the work done. It enables us to know what sort of contribution influenced a given part of the ontology and to trace the effectiveness of reuse. However this is by far not enough and the complete design rationale of the ontology should be captured because it explains what motivated its actual state and helps people understand and commit to or adapt it.

2.7 On a continuum between formal and informal

The final formal degree of the ontology depends on its intended use. The formalization goal is not to replace the informal version by a formal one but to augment the informal version with the formal counterpart of relevant semantic aspect in order to obtain a documented (informal description possibly augmented by navigation capabilities from the formal description) operational ontology (formal description of the relevant semantic attributes needed for the envisioned application). The ontologist stops his progression on the continuum between informal and formal ontology as soon as he reaches the formal level necessary and sufficient for his application. Therefore, the informal version of the ontology is not merely an intermediary step that will disappear after formalization, the formal form of an ontology must include the natural language definitions, comments, remarks, that will be exploited by humans trying to appropriate the ontology. "Ontologies have to be intelligible both to computers and humans" [Mizoguchi and Ikeda, 1997]. This plays an important role in documenting the ontology and therefore in enabling reuse and maintenance of ontologies. The tables described before evolved from semi-informal to semi-formal until the taxonomic links were sufficiently explicit to be translated in RDFS by scripts. The ontology content did not changed, but the underlying structure switched from informal tables to formal taxonomic relations exploitable by software. In RDF(S) concept intensions are formalized as resources classes, relations and attributes are formalized as property classes and extensions are instances of these classes. As multi-instantiation is permitted in RDF, one resource may be an instance of several classes. Figure 5 shows how RDFS can be used to implement the different levels introduced previously:

- the terminological level where collected terms are organized. Relations between the intensional level and the terminological level denote possible labels for each intension (property `rdfs:label`). An intension with several terms linked to it (e.g. in figure 5 : C_4) is characteristic from the synonymy of these terms. A term with several intensions linked to it (e.g. in figure 5 : T_3) is characteristic of the ambiguity of this term.
- the intensional level where the intensional structure of the ontology is formalized. Relations between the intensional and the extensional level represent the instantiation of a class of a concept. The bundles of relations link an intension to its extension (e.g. in figure 5 : C_8).
- the extensional level where the factual memory is organized (annotations, state of affairs, user profiles). An extension linked to several intensions (e.g. in figure 5 : C_6 and C_7) is characteristic from multi-instantiation.

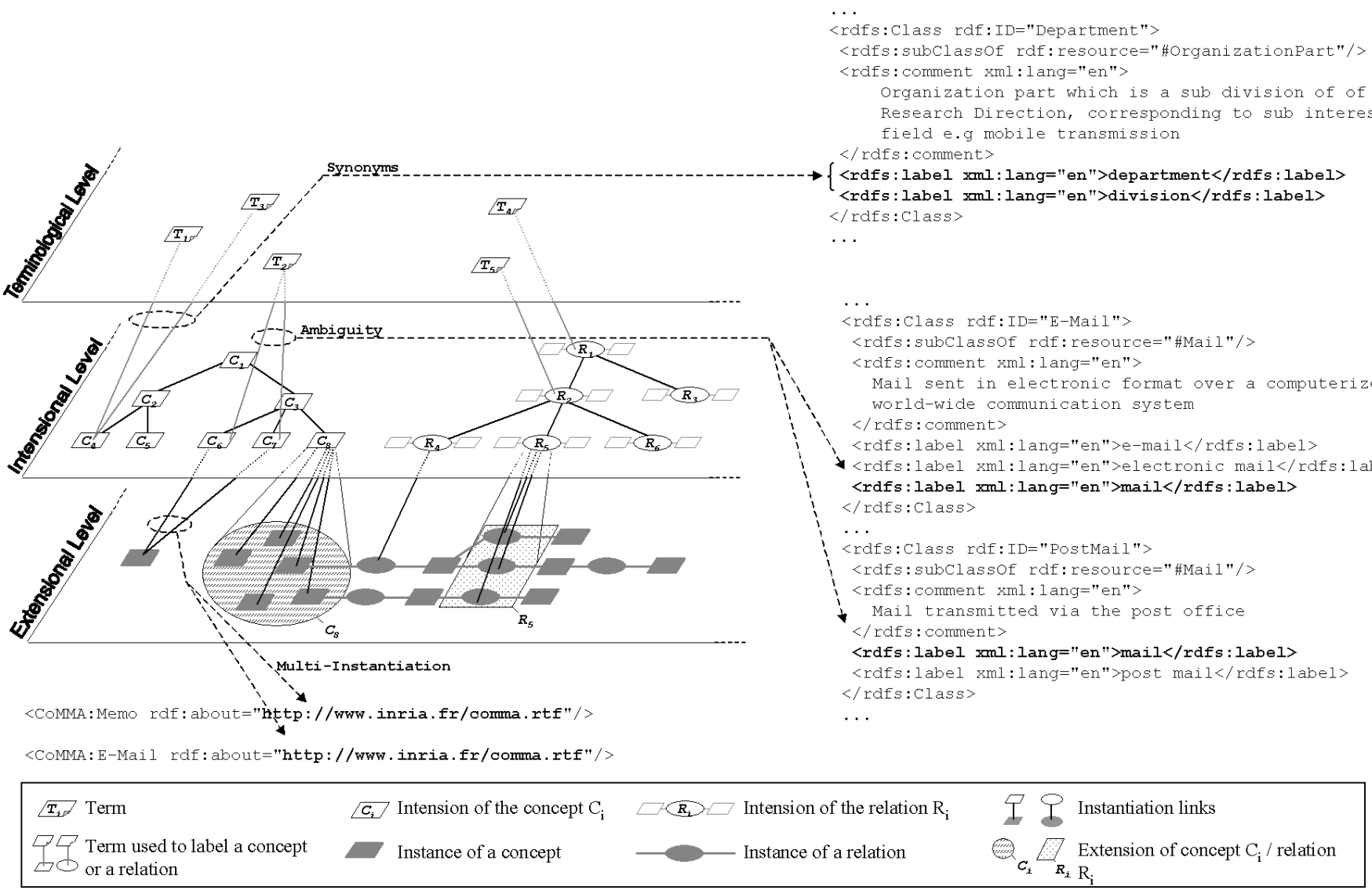


Figure 5 - The terminological, intensional and extensional levels

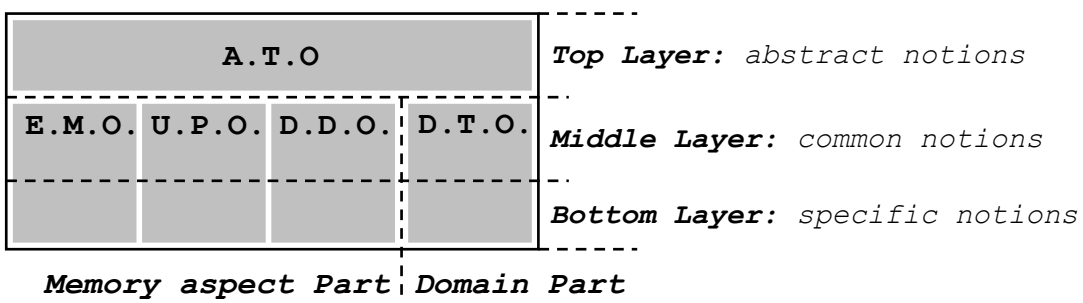
Using XLST style sheets we kept the informal views: (a) a style sheet recreates the initial terminological table representing a sort of lexicon of the memory (b) two other ones recreate the tables of concepts and properties (c) a set of five style sheets propose navigation and research at the conceptual or terminological levels, search for concepts or relations linked to a term, navigation in the taxonomy, search for relations having a signature compatible with a given concept. The user can also ask for the listing of the extension of a concept. A sample of this extension can play the role of examples to ease understanding of a notion (d) a style sheets filters the ontology with the user profile to propose preferred entrance point in the ontology to start browsing (e) a last style sheet proposes a new view as an indented tree of concepts with their attached definition as a popup window following the mouse pointer.

The last view is an interesting improvement. It is a first attempt to investigate how to proactively disambiguate navigation or querying. Before the user clicks on a concept the system displays the natural language definition; this popup window invites the user to check his personal definition upon the definition used by the system and avoid misunderstandings.

2.8 Result and Discussion

The current ontology contains more than 420 concepts organized in a taxonomy with a maximal depth of 12 subsumption hops (using multi-inheritance), and more than 50 relations. As shown in figure 6, O'CoMMA has more or less three layers:

- A very general top that roughly looks like other top-ontologies
- A very large and ever growing middle layer that tends to be divided in two main branches: (1) one generic to corporate memory domain (documents, organization, people...) and (2) one dedicated to the topics of the application domain (telecom: wireless technologies, network technologies...)
- An extension layer which tends to be scenario and company specific with internal complex concepts (Trend analysis report, New Employee Route Card...)



A.T.O. : Abstract Top Ontology
E.M.O. : Enterprise Modeling Ontology
U.P.O. : User Profile Ontology
D.D.O. : Document Description Ontology
D.T.O. : Domain Topic Ontology

Figure 6 - Structure of O'CoMMA

Concerning the equilibrium between usability and reusability of the ontology the upper layer is extremely abstract and the first part of the second layer is describing concept common to corporate memory (e.g. person, employee, document, report, group, department, ...), therefore they both seem to be reusable in other application scenarios. The second part of the middle-layer deals with application domain (in our case telecom and building industry). Therefore it would be reusable for scenarios only in the application domain. The last layer extends the two previous parts with specific concept that should not be reusable as soon as the organization, the scenario or the application domain change.

Industrial partners confronted to the current ontology explained that the complexity of it is too big which makes it completely abstruse to people. Especially the upper part of the ontology that introduces philosophical distinctions which are extremely interesting from a modeling point of view (the top of the ontology provides sound building blocks to start modeling and ensure coherence in the rest of the ontology) but are extremely abstruse and usually useless for typical users of the system. Moreover the higher you are in the taxonomy the more difficult the agreement is. Two colleagues will more easily agree on the modeling of concepts they manipulate and exchange in their day-to-day work (e.g. a news is a type of document presenting new information) than on the top of the ontology that requires a higher level of generalizing and abstraction and deals with concepts that we are not used to discuss every day (e.g. things are divided between entities and situations, entities being things capable of playing a role in a situation). The top-level deals with cultural and even personal beliefs. Top-level concepts are useful for system internal manipulations (generalization of queries, structuring of higher layers) but not for the direct interaction with users. An ergonomic and pedagogical representation interface is a critical factor for the adoption of the ontology by the users; if the user is overloaded with details or lost in the meandering of the taxonomy he will never use the system and the life-cycle of the ontology will never complete a loop. In the CoMMA project, we are investigating this problem and developing a tool for annotation that should also enable us to manipulate the ontology. However, results from the first prototypes show that the ergonomics issues are far from being solved.

We investigated the use of user's profiles to record preferred entrance points into the ontology in order to hide the upper level and propose middle concepts (e.g. person, document, organization) from which the user can start his navigation in the ontology. This proved to be a great improvement for ontology browsing. Current work on machine learning techniques also studies how to identify frequently used concepts in order to improve navigation and result presentation.

Another problem spotted during the formalization was the redundancy of information that may appear. For instance, annotating a document as multi-modal is redundant with the fact that it is annotated with the different modes it uses. So we decided that the multi-modal was not a basic annotation concept and that it should be a 'defined concept', that is a concept derived from other existing concepts where possible. However the notion of defined concept, does not exist in RDFS, and we will have to extend the schema as proposed in [Delteil *et al.,* 2001]. The same applies to unstable or fuzzy concepts for instance a 'Newcomer' : how long are you a new comer in a company ? the definition could change even inside a project involving different companies. These concepts have to be defined based on another information in our example it could be the date of hiring. These choices also arise when formalizing, where sometime a concept can become a relation, an inference... and vice-versa.

The notion of 'colleague' for instance, was first a concept. Then it became a relation 'x is a colleague of y'. This relation needed to be typed as 'symmetric' but RDF(S) does not provide that option. And then, considering the application scenarios, it appeared that this relation would not be used for annotation but that it will more likely be inferred from what can be described in the state of affairs (Mr. X and Ms. Y belong to department D therefore there is a 'colleague' relation between X and Y). Thus we needed the ability to formally define the conditions that make two people 'colleagues'.

These limitations of RDF(S) are quickly a problem since the ability to factorize knowledge in an ontology requires the ability to give formal definitions. In the current version of the system, the formal definition of 'Colleague' was coded in a rule (Figure 7) written in an XML rule language specially created for RDF and CORESE. The symmetry, transitivity and reflexivity properties have also been added as extensions of RDF(S) specific to CORESE.

```
<cos:rule>
    <cos:if>
      <rdf:RDF>
          <CoMMA:OrganizationalEntity>
            <CoMMA:Include> <CoMMA:Person rdf:about="?x"/> </CoMMA:Include>
            <CoMMA:Include> <CoMMA:Person rdf:about="?y"/> </CoMMA:Include>
          </CoMMA:OrganizationalEntity>
      </rdf:RDF>
    </cos:if>

    <cos:then>
      <rdf:RDF>
          <CoMMA:Person rdf:about="?x">
            <CoMMA:Colleague> <CoMMA:Person rdf:about="?y"/></CoMMA:Colleague>
          </CoMMA:Person>
      </rdf:RDF>
    </cos:then>
</cos:rule>
```

Figure 7 - Rule formally defining the 'Colleague' relation

A first draft of the ontology was an good step for feasibility study and first prototypes, but refining, validation and checking works is hard and it comes with no surprise that the prototype life-cycle is time consuming. The review is also triggered by feedback from trials, end-user's complaint about what is missing, or what has not been conceptualized or formalized properly. The ontology is a living object the maintenance of which has consequences beyond its own life-cycle : it has an impact on everything that was built upon it. A software where the ontology was hardwired has to be versioned, knowledge bases coherence have to be maintained... Therefore, although the problem of the ontology evolution itself is a hard one, one should consider the fact that ontologies provide building blocks for modeling and implementation. What happens to the elements that were built thanks to these building blocks when a change occurs in the ontology? Deletion and modification obviously raise the crucial problem of coherence and correction of the annotation base. But an apparently innocuous addition of a concept also raises the question of the annotations using a parent concept of the new concept and that could have been more precise if the concept had existed when they were formulated. The question is: should we review them or not ? These problems are obviously even more complex in the context of a distributed and heterogeneous system.

3 Conclusion

In this article, we presented the approach evaluated in CoMMA, to build a corporate memory management framework based on multi-agent systems, machine learning and semantic web technologies. We showed that the ontology is a keystone of such a multi-agent corporate memory system and we presented our experience in building it. We explained the data-collection techniques and the usefulness of application scenarios. We then described how other ontologies and expertise sources influenced our work, stressing the fact that no direct reuse was possible. We detailed the work on terminology explained why this aspect should be kept in the formal ontology. We then exposed the structuring stage especially arguing on the opposition of the bottom-up, top-down and middle-out approaches that we believe to be complementary perspectives of a complete methodology. Finally we explained how we formalized the terminological and intensional levels of the ontology in RDFS and how XSLT enables us to navigate within it. We insist on the fact that since ontologies are playing a seminal role, it becomes urgent to converge toward unified frameworks gathering the numerous contributions made this past ten years to the ontological research field and to provide ontologists with a comprehensive integrated set of tools required to follow the complete ontology life-cycle.

Acknowledgments

I warmly thank my colleagues Rose Dieng, Olivier Corby and Alain Giboin, the CoMMA consortium and the European Commission funding the CoMMA project.

References

[Aussenac-Gilles *et al.,* 2000] Aussenac-Gilles N., Biebow B, Szulman S (2000). Revisiting Ontology Design : a Method Based on Corpus Analysis, In Proc. EKAW'2000, Juan-les-Pins, p172-188.

[Austin, 1975] Austin JL How to Do Things with Words, Harvard Univ. , 2nd edition

[Bachimont, 2000] Bachimont B., Engagement sémantique et engagement ontologique: conception et réalisation d'ontologies en ingénierie des connaissances, In Charlet, Zacklad, Kassel, Bourigault, Ingénierie des connaissances Evolutions récentes et nouveaux défis. Eyrolles, 2000

[Berners-Lee *et al.,* 2000] Berners-Lee T., Hendler J., Lassila O., The Semantic Web, In Scientific American, May 2001, p35-43

[Caroll, 1997] Caroll J., Scenario-Based Design, In Helander, Landauer, Prabhu Handbook of Human-Computer Interaction. 2nd ed., Ch.17, Elsevier Science

[CoMMA, 2000] CoMMA Corporate Memory Management through Agents, In Proc. E-Work & E-Business, Madrid

[Corby *et al.,* 2000] Corby O., Dieng R., Hébert C. A Conceptual Graph Model for W3C Resource Description Framework. In Proc. ICCS'2000 Darmstadt Germany

[Cyc, 2000] www.cyc.com/cyc-2-1/cover.html

[Decker *et al.,* 1999] Decker S., Erdmann M., Fensel D., Studer R., Ontobroker: Ontology based access to distributed and semi-structured information. In Meersman et al Semantic Issues in Multimedia, Systems, Kluwer

[Delteil *et al.,* 2001] Delteil, Faron, Dieng, Extension of RDFS based on the CG formalism, In Proc. ICCS'01

[Doyle and Hayes-Roth, 1998] Doyle P., Hayes-Roth B., Agents in Annotated Worlds, In Proc. Autonomous Agents pp. 173-180, ACM Press / ACM SIGART

[Fernandez *et al.,* 1997] Fernandez M., Gomez-Perez A., Juristo N., METHONTOLOGY: From Ontological Arts Towards Ontological Engineering. In Proc. AAAI97 Spring Symposium Series on Ontological Engineering, pp. 33-40, Stanford, USA

[Gandon *et al.,* 2000] Gandon F., Dieng R., Corby O., Giboin A., A Multi-Agents System to Support Exploiting an XML-based Corporate Memory, In Proc. PAKM Basel

[Gastinger and Szegheo, 2000] Gastinger A., Szegheo O., Enterprise Modeling Approaches, In Rolstadås, Andersen Enterprise Modeling: Improving Global Industrial Competitiveness. pp. 55-69, Kluwer

[Gómez-Pérez *et al.,* 1996] Gómez-Pérez, A., Fernandez M. De Vicente A., Towards a Method to Conceptualize Domain Ontologies, In Proc. Workshop on Ontological Engineering ECAI'96. pp. 41-51

[Guarino and Welty, 2000] Guarino N., Welty C., Towards a methodology for ontology-based model engineering. In Proc. of ECOOP-2000 Workshop on Model Engineering

[Guarino and Giaretta, 1995] Guarino N., Giaretta P., Ontologies and Knowledge Bases: Towards a Terminological Clarification. In Mars N., Towards Very Large Knowledge Bases, IOS Press

[Guarino, 1992] Guarino N., Concepts, Attributes, and Arbitrary Relations: Some Linguistic and Ontological Criteria for Structuring Knowledge Bases. In Data and Knowledge Engineering pp. 249-261

[Heflin *et al.,* 1999] Heflin J., Hendler J., Luke S., SHOE: A Knowledge Representation Language for Internet Applications. Institute for Advanced Computer Studies, University of Maryland at College Park.

[Kassel *et al.,* 2000] Kassel G., Abel M.., Barry C., Boulitreau P., Irastorza C., Perpette S., Construction et exploitation d'une ontologie pour la gestion des connaissances d'une équipe de recherche In Proc. IC'2000

[Lieberman, 1999] Lieberman H., Personal Assitants for the Web : An MIT Perspective. In Klush Intelligent Information Agent: Agent-Based Information Discovery and Management on the Internet. pp. 279-292 Springer

[Martin and Eklund, 2000] Martin P., Eklund P., Knowledge Indexation and Retrieval and the Word Wide Web. In IEEE Intelligent Systems special issue Knowledge Management and Knowledge Distribution over the Internet

[Martin and Eklund, 1999] Martin P., Eklund P., Embedding Knowledge in Web Documents, In Proc. of the International World Wide Web Conference

[Mizoguchi and Ikeda, 1997] Mizoguchi R., Ikeda M., Towards Ontology Engineering, In Proc. Joint Pacific Asian Conference on Expert systems / International Conference on Intelligent Systems, pp. 259-266

[OneLook, 2000] www.onelook.com

[Papazoglou and Heuvel, 1999] Papazoglou, Heuvel, From Business Process to Cooperative Information Systems: An Information Agents Perspective, In Klush Intelligent Information Agent: Agent-Based Information Discovery and Management on the Internet. pp. 10-36, Springer,

[Rabarijaona *et al.,* 2000] Rabarijaona A., Dieng R., Corby O., Ouaddari R., Building a XML-based Corporate Memory, IEEE Intelligent Systems, Special Issue on Knowledge Management and Internet pp. 56-64

[Ribiere, 1999] RIBIERE M. Représentation et gestion de multiples points de vue dans le formalisme des graphes conceptuels. PhD from Univ. Nice Sophia-Antipolis

[Rolstadås, 2000] Rolstadås A., Development trends to support Enterprise Modeling; In Rolstadås A. and Andersen B. Enterprise Modeling: Improving Global Industrial Competitiveness. pp. 3-16, Kluwer

[Steels, 1993] Steels L., Corporate Knowledge Management. In Barthès Proc. ISMICK'93 pp. 9-30

[Singh and Huhns, 1999] Singh M.P., Huhns M.N., Social Abstraction for Information Agents. In Klush, Intelligent Information Agent: Agent-Based Information Discovery and Management on the Internet. pp. 37-52 Springer

[TOVE, 2000] www.eil.utoronto.ca/tove/ontoTOC.html From the Enterprise Integration Laboratory

[Uschold and Gruninger, 1996] Uschold M. and Gruninger M. Ontologies: Principles, methods and applications. Knowledge Engineering Review, (11)2: pp. 93-136

[Uschold *et al.,* 1998] Uschold M., King M., Moralee S. and Zorgios Y., The Enterprise Ontology. In Uschold and Tate The Knowledge Engineering Review Special Issue on Putting Ontologies to Use Vol. 13

[Wooldridge and Jennings, 1995] Wooldridge M., Jennings N.R., Intelligent Agents: Theory and Practice. The Knowledge Engineering Review, 10(2): pp. 115-152

[Wooldridge *et al.,* 1999] Wooldridge M., Jennings N.R., Kinny D. A Methodology for Agent-Oriented Analysis and Design. In Proc of Autonomous Agents '99 Seattle, ACM 1-58113-066-x/99/05