MOSTRO Del.7, WP 4
Case Study Description

University of Trento

March 27, 2007

Contents

Chapter 1

Introduction

The analysis of case studies is a fundamental activity in most current projects, especially when concerning a real organization. As MOSTRO project recognizes the crucial role of this activity, one of its objectives is the definition of at least one case study, which is the focus of the current deliverable, and the application of the organizational analysis methodology to it, which will be presented in Deliverable 8. This deliverable presents a detailed description of two case studies, along with the motivation behind their selection and analysis.

The examined contexts are e-Voting [1, 4] and Tutto-In-Uno; they are both derived from Autonomous Province of Trento (PAT) projects and hence regard an organization acting in Trento, as required by the MOSTRO project workplan. E-Voting case study describes the ProVotE project [1], which aims at a smooth transition towards e-based voting procedures before the next provincial elections in Trentino (that will be held in 2008). ProVotE is a multidisciplinary project, as changing voting procedures are not just a technological matter, but involve many areas, such as law, social studies, psychology, along with computer science and engineering. Hence, the nature and complexity of side effects which could be caused by introducing a new voting system have to be carefully studied and analyzed, in order to detect and prevent problems before the actual elections take place. Tutto-In-Uno case study is a project with the aim of providing the Legal Office of the PAT with a unified view on the information sources involved in the management of administrative proceedings. It should result in a reduction of the difficulties and costs of retrieving, accessing and using data that are now distributed over distinct applications.

Some motivations for selecting these two case studies are shared by both ProVotE and Tutto-In-Uno, namely, in both scenarios the organizational setting is rich and complex. The voting process involves a number of interacting actors,

organizational entities and technological systems, which must meet strict legal requirements while carrying out their activities. Tutto-In-Uno models the complex structure of PAT keeping track of the legal documents associated to the design steps that conduced to that structure, and the administrative proceedings/competences assigned to all its sub-structures and positions. Security issues are especially crucial for the e-voting case study, as the corresponding procedures must guarantee privacy, secrecy and trust of the voting process. Instead, the system developed in Tutto-In-Uno is only accessible by experts, therefore security issues are not as relevant at this level. On the contrary, the analysis of the security of the PAT, and how a change in its structure impacts its security are analytical features of great value for the project even though this is not its main goal. Tutto-In-Uno has been chosen as a case study especially because it needs to be based on a detailed and well founded ontology of organizations, that, in our opinion, is the necessary step towards the security analysis.

We expect that the case studies will give a strong contribution to MOSTRO, since the organizational analysis methodology to be proposed in Deliverable 8, will be applied to these case studies and, thus, validated. The outcome of these future steps will be the possible identification of problems and limitations of the methodology. In general terms, what we can expect from the case studies is a feedback on the applicability of MOSTRO organizational analysis methodology.

The remainder of this deliverable is structured as follows. Section 2 describes the e-voting case study, describing it in general terms and going into deeper details in some areas which can be of particular interest. Section 3 introduces the Tutto-In-Uno case study in an analogous way. Section 4 ends this deliverable giving conclusions.

Chapter 2

e-Voting Case Study

Art. 84 of PAT (Autonomous Province of Trento) Law 2/2003 mandates the introduction of e-voting for the next provincial elections (to be held in 2008). To actuate the law, the Province is sponsoring the ProVotE project [1], that has the goal of providing a smooth transition to the new technologies. The project develops along different lines, among which the process/logistical line, that aims at defining the procedural, organizational, and normative framework that will regulate an electronic election.

Electoral procedures involve different organizations, several people over periods of months, and have strict security and traceability requirements. This chapter describes the work carried out within the process/logistical line of the ProVotE project, which was done in collaboration with the MOSTRO team [4], and involved applying the fragments of the methodology for organizational and security analysis developed within MOSTRO. The chapter describes the approach taken in order to provide precise models of the electoral processes, while, at the same time, providing mechanisms for documenting, reasoning on the possible alternative implementations of the procedures to support the elections of 2008.

2.1 Case Study Description

2.1.1 The ProVotE Project

ProVotE has the goal of ensuring a smooth transition to e-voting in Trentino, eliminating risks of digital divide and providing technological solutions which support, with legal value, the phases ranging from voting to publication of the elected candidates.

The project includes partners from the public administration (Provincia au-

tonoma di Trento, Regione Trentino/Alto-Adige, Consorzio dei Comuni Trentini, Comune di Trento, IPRASE), research centers and academia (ITC-irst, Faculty of Sociology of the University of Trento, Fondazione Graphitech), and local industries (Informatica Trentina) and is co-led by the Electoral Service of the Autonomous Province of Trento and by ITC-irst. Project leadership by the Public Sector, in our opinion, among other advantages, helps tackling the issue of potential conflicts of interests by private industries, see e.g. [14].

The project is multi-phased and is organized in various lines of activities which strictly interact. For instance, in the first phase of the project, some functional and non-functional requirements of the e-voting prototype, were built with a strict round-trip between the sociological and the technological line, with the normative line ensuring compatibility with the laws. See [19, 5] for more details and [16] for some considerations related to the sociological aspects of e-voting.

Various trials have been conducted to assess the results of the first phase of the project. The trials have had the goals of testing prototypes, evaluating acceptance by citizens, ease of use, etc. So far more than 11.000 citizens have tried the systems, either with experimental value (in four trials conducted in parallel to local elections) or with legal value (election of the representatives of the students in a local high school, involving about 1000 students).

For the second phase of the project, which will lead to a large-scale introduction of the new voting system, aspects related to procedures, organization, processes become more relevant, as they will serve both as the basis for the deployment of the solution and for the definition of the laws that will govern the electronic election.

With respect to scope, population, and participation, ProVotE is among the largest, if not the largest, e-voting project in Italy.

2.1.2 Voting Procedures in Italy and e-voting Experimentations

The electoral law for political elections in Italy was defined in the forties and even though it has undergone various revisions — mainly related to the algorithms for determining the elected representatives — most of the procedures applied today are those defined about sixty years ago.

Simplifying both on the law and on the procedures for the sake of presentation, voting in Italy happens as follows:

1. **Identification and registration of the voter.** At the polling station the voter is usually required to show his/her ID card and the electoral card. If the name of the voter is present in the electoral list of the polling station, the voter is registered, the electoral card stamped, and the voter is admitted to voting.

2. **Casting a vote.** The voter is given a ballot and a pencil and is shown a cabin where the vote can be cast in secrecy. Secrecy is both a right and a duty. The Italian law and procedures are aimed at ensuring that the voter cannot make his/her vote manifest to other people.

At the end of the voting day, the ballot boxes are opened and the counting procedure starts:

3. **Counting.** Votes are counted and the results tabulated in special registers. The Italian law aims at protecting the intention of the voter. Thus, even if a vote is not compliant with the definition given by the law, the vote may still be partially assigned and counted, if some its parts can be unambiguously interpreted. Representatives of the political parties monitor the counting procedure.

4. **Transmission of the results.** When all the ballots have been tabulated, the results are transcribed in various paper documents and transmitted to the offices responsible of aggregating all the data.

5. **Sum and proclamation of the elected representatives.** All the data coming from the different polling stations are counted and seats assigned according to algorithms defined by the law. Data are then made available to the general public.

Various experimentations have been conducted in Italy to introduce new technologies in the polling stations. The largest trial, so far, was sponsored by the central government, and concerned a system for automating steps 3 and 4 above. The system, operated by specially appointed technicians, was installed in 47 precincts at the last European elections and repeated at the last political elections (2006). Little, however, is known about the results of the experimentation. See [11] for some more details.

Proper e-voting experimentations (i.e. including step 2) have been conducted at the local level, usually on a small scale, in experimentations which seem to have had little continuity and/or on which information is scarce. We mention San Benedetto del Tronto (2000), trials sites in Avellino (2001), Campobasso (2001), Cremona (2002, 2006), Ladispoli (2004), Specchia (2005) [7, 8]. Other experimentations have been conducted in Valle D'Aosta, Friuli Venezia Giulia, and Milan.

2.1.3 Modelling Electoral Procedures in the UML

Electoral laws and procedures have strict security and traceability requirements aimed at ensuring that frauds are extremely difficult to be undetected. To fulfill

such goal they define chains of responsibility and delegation, mechanisms for monitoring procedures and mutual controls. Thus, for instance, in Italy, even though the results of an election are available the next day, the confirmation and officialisation of the results requires the Public Administration to perform a strict series of checks, that can take up to a month after the election day.[1]

The introduction of new technologies in the polling stations not only changes the way in which we cast votes, but also roles and responsibilities, often in subtle ways (see e.g. [15].) For instance, the introduction of voting machines may change the tools polling officers and representatives of the parties use to verify the tabulation of data (think for instance of voting machines with no printed trails, in use in some countries). In such a scenario, to maintain the same security/verifiability requirements of a paper election, it may be necessary to introduce various changes to the procedures (e.g. allow the parties and polling officers to test the machines long before the election; provide ways to verify what software is installed on the machines used during the election day).

To mitigate the risk of creeping security "holes" in the electronic procedures, the ProVotE team has decided to provide extensive modelling of processes. The model of the existing procedures provides a baseline for the definition of the new procedures, that have to be devised so that the following requirements are met:

- changes are as minimal as possible: we do not want to alter in "significant" ways how elections are conducted;
- the same security level of a paper election is maintained: that is, in an electronic election we can at least identify and mitigate the same "attacks" of a paper election;
- the new threats introduced by electronic systems are dealt with by specifically devised procedures and checks.

To do so, it was decided to provide detailed models of the current electoral processes and devised a specific methodology, based on the UML, to support functional analyst in modelling the applicative domain, while keeping uniform quality standards. The use of UML, in this case, was an essential requirement for various reasons, among which: expertise, tool support and ease of understanding by the domain experts.

The UML models of paper based voting procedures produced by the ProVotE team contain hundreds of diagrams, with more than 150 actors and entities and

[1] Usually most of the issues raised involve identifying and dealing with material errors of polling officers during the election day.

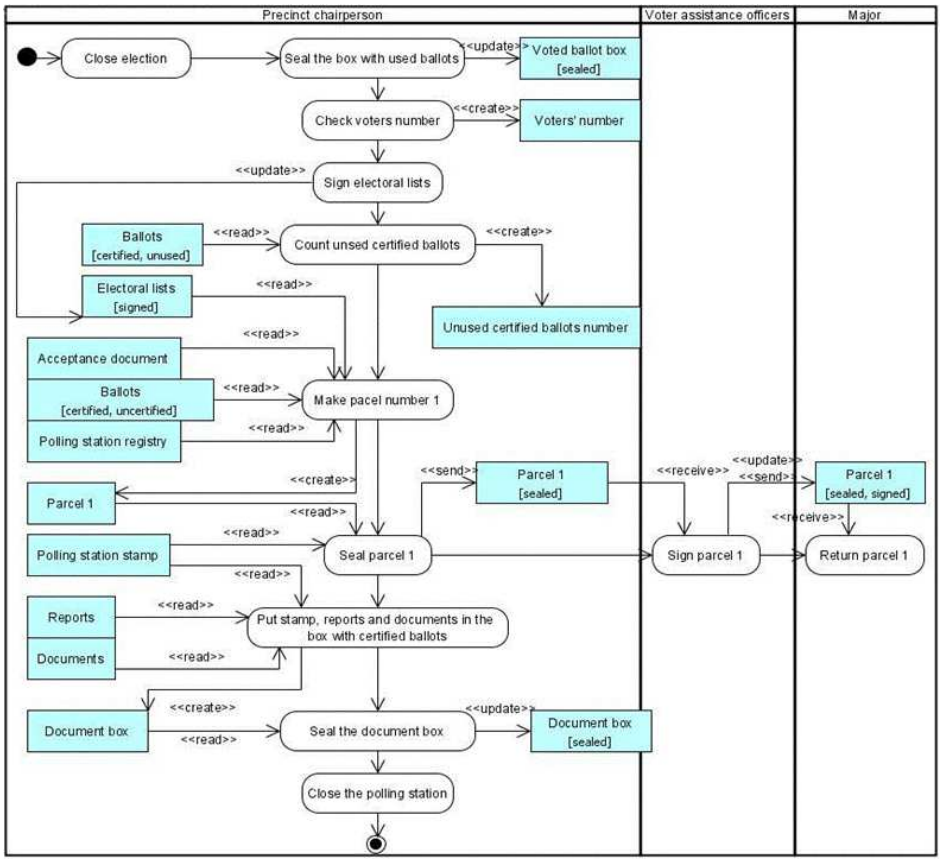

Figure 2.1: Closing procedures: activity diagram.

almost two thousands relations among them. One example, namely, the activity diagram for the closing procedures, can be seen in Figure 2.1.

Among the advantages of the definition of a methodology, we can mention the possibility of performing (semi)automated analysis on the models. In particular, VisualParadigm was used as the modelling tool and the API it offers was exploited to implement various extensions to perform custom analysis. Among the supported functions, there is the possibility of extracting information on what actors are responsible for what artifacts produced in an election (see [13] for more details).

2.2 Problems in modeling e-Voting in UML

Not surprisingly, there are different ways of modifying the existing procedures ("as is") in order to define procedures that support an electronic election ("to be") and satisfy the constraints mentioned above. Thus, the definition of the exact procedure to be followed, among all the possible alternatives, should take into consideration other requirements (e.g. economicity, efficiency, etc.) and be based on mechanisms to weigh and evaluate the different choices.

However, the UML is weak in providing means of describing alternatives and, by extension, the methodology devised by the project team falls short in providing ways to describe the *why* of the transition from the "as is" to a specific "to be". Hence the need to complement the UML modelling with some other approach more suited to face these issues. The improvement needed to fill the gap should adopt a mentalistic approach by modeling both humans and information systems as networks of interdependent actors endowed with goals.

The techniques to be adopted should cover the very early phases of requirements analysis, in which the analysis is centered on the organizational environment where the software must operate, thus on the social relations that preexist to the software and on the changes and improvements that should incur to the environment when the software is introduced. On the contrary, the UML model of the system "to be" shows how the voting scenario changes after the introduction of the information system, but it cannot explain why such changes have been introduced. Usually the rationale lies in some features that the designer wants the new system to possess, i.e. in the requirements of the new system. Particular attention has to be paid to the ability of modeling and evaluating alternatives, such as different strategies for fulfilling a specific goal of an actor: if the proposed techniques could identify the best set of alternatives, we would get a means for selecting the most appropriate solutions for the "to be" system.

Moreover, the chosen set of techniques needs to face security and trust relationships in an effective way, since the focus of MOSTRO project is on that particular kind of properties.

Some existing works, e.g. [10, 12, 20], use goal analysis (in particular, they apply Tropos methodology) to model the choice between alternatives by combining or-decomposition and contribution between goals and – by representing non-functional requirements as soft goals[2] – help understanding which choices favored more the satisfaction of a requirement.

The new approach would not replace the existing UML modeling, because

[2]Soft goals are goals for which it is not straightforward to determine whether they have been achieved or not. In this respect they are similar to non-functional requirements, like security: there is no trivial way to decide if a system is secure or not.

we are convinced of the contribution of both UML and goal-based modeling languages. Tropos [3] is the specific methodology we identify as optimal, because of the capability of managing goals and alternatives. The idea is that of keeping each modelling approach to do just what is best suited for, namely modelling *processes* on the one hand (UML), and doing *goal driven* reasoning on the other. Thus the UML models provide an exact snapshot of the procedures (independently from the motivations for which they have been devised in a specific way), while, at the same time, goal modeling helps to maintain track of the reasons for any change we had to introduce to support electronic elections.

From a technical standpoint, this translates into an approach which produces an alternating sequence of UML and Tropos models. In particular, the UML model is used at the "object" level to model processes. After the UML model is given, Tropos goal models are used at the "meta" level, to reason about design alternatives with a twofold purpose:

1. to provide a rationale for the solutions adopted for the implementation of the system "to be", by modelling possible alternative ways of accomplishing a goal;

2. to explore trust and security issues related to the e-voting process.

The results of the analysis allow, in turn, to modify the existing UML model and devise the new procedures that meet the goals stated in the Tropos goal model. The steps described above are then iterated as needed.

Ideally speaking, every solution in the system "to be" should be taken after having accurately explored all the alternative possibilities. In this case, the role of the goal modeling is that of a visual tool that gives support to the people involved in this decision-making by providing them with a general overview of the choices under consideration, so that they could explore all the available alternatives prior to choosing a solution.

In practice things are not so well methodologically applicable. Often these decisions emerge from informal discussions and are constrained by stringent legal requirements. However, even in these cases, given the involvement of different stakeholders, the goal modelling is useful as it documents and represents the motivation behind the choices. Finally, it can happen that, even in cases in which some solution has already been chosen, once that all the alternatives are represented, it comes out that some alternatives not previously considered suit better the requirements. Thus, the goal modelling can also be seen as a validation tool for the choices made.

So far we have seen the potentialities of introducing goal modeling; but, methodologically, how are the elements of the model chosen? If the main purpose

of the model is that of exploring, evaluating and eventually motivating choices between different alternative ways of accomplishing a goal with respect to a list of requirements, the methodological questions amount to the following two:

1. how are the different alternatives singled out?

2. how are the requirements that provide the reference for evaluation selected?

The first question can be rephrased as follows: how to transform well established procedures based on physical support, like pencils, sheets of paper, cardboard boxes etc. in practices based on an electronic support? The possible alternatives are constrained in many ways and these constraints come from several dimensions: technological, legal and social. The main source for the formulation of the alternatives have been the stakeholders of the project: interviews were conducted with the development team that raised technological issues, other interviews took place with the representatives of the Electoral Service of the Province, who were mainly concerned with the compliance with the provincial legislation regulating elections.

The second point, namely, the choice of the right requirements for reference during evaluation of alternative choices, was not an easy task either. Some of these requirements are strictly inherent to the information system itself, its features and functionalities, and are relative to its technical performance. Other requirements come from the interaction of the system with the environment and are more related to the effects of the system on the organizational level.

The more general requirements, as maintainability or cost concerns, have mainly been taken from the Software Engineering literature (for some references see [17] and [6]); these represent properties that are desirable for any information system. Other requirements are more specifically oriented to security issues, such as confidentiality, integrity, availability, etc. (see, e.g., [6], Chap. 7). These are particularly relevant in the e-voting scenario, since it is crucial that the system is not vulnerable, otherwise there could arise major concerns, like results being manipulated, or votes being associated to particular electors.

As mentioned above, security is not the only concern for the implementation of an e-voting system. For other more specific requirements, as non-traceability of votes or minimal change to the existing legislation, we also took inspiration from existing work, such as, for instance (see, e.g., [18, 9]).

Finally, there is a very specific requirement that is peculiar of this very project and that comes from the main objective of the project itself, namely the smooth transition from the old paper system to the e-voting. This objective brings with it a very stringent requirement, which is compliance with the existing legislation on voting in the PAT (that is contained in [2]). This is a requirement that is important

for several stakeholders (like, for instance, legislators, but also common citizens), as changing the law is a (politically and bureaucratically) complex and time consuming process. Moreover, the closer the new procedures are to the old ones, the less the people involved in such procedures have to be instructed and the lower the probability of mistakes is going to be.

2.3 Conclusions

In this chapter we have firstly presented the e-voting case study, which is derived from ProVotE project, aiming at a smooth transition towards the electronic vote in Trentino. We have shown the multiple perspectives and their strict integration. Then we listed and motivated the limits of the current modeling approach, based on the Unified Modeling Language (UML), which is capable of describing the *how* but fails in showing the *why*.

As a solution for filling this gap we proposed the integration of Tropos goal modeling to the existing methodology. The integration exploits complementary features of the two modelling approaches and allows to maintain both an operational view of the voting procedures and a visual approach to evaluate choices in designing the electronic processes "to be".

The general approach we identified has to be refined in the next deliverables (especially in Deliverable 8), but it will likely be not restricted to the application domain, since we believe it should be easily applicable to other business process re-engineering contexts.

Chapter 3

Tutto-in-Uno Case Study

3.1 Case Study Description

As already stated in the introduction, this case study originates from a project funded by the PAT and involving a company residing in the PAT's territory, namely the BPE (Business Process Engineering) company (see `http://www.bpeng.com`)[1].

The main objective of the *Tutto-in-Uno* project is that of providing the Legal Office of the PAT with a unified view on the information sources involved in the management of administrative proceedings. The motivations behind this objective lay in the difficulties and costs of controlling and organizing data for their retrieval, access and use, due to the fact that they are distributed over distinct applications.

The main sources in which these data are collected are:

- the flowchart (database) of the PAT;
- the database of the determinant judgments[2] of the structures' managers;
- the database of the deliberations of the government of the province (*Giunta provinciale*);
- a file that collects all administrative proceedings;
- the files collecting the declaratory judgments[3] and the competences;

[1] This section is based on an internal document provided by the BPE company in the scope of the mutual cooperation with the partners of the MOSTRO project.

[2] Determinant judgments (*Determine*) are acts signed by a manager.

[3] Declaratory judgments (*Declaratorie*) are juridical acts with a declaratory character that assign competences to structures of the PAT (Department, Service, Office, Special Assignment). They are expressed by a collective organ of the government.

- the database of the provincial laws;
- the matters of competence contained in the Statute for the Autonomy of the PAT;
- the decrees which assign competences to structures.

From a practical standpoint, this unified view would consist in taking each proceeding contained in the archive of the Legal Office and associating it with all the related information belonging to different sources, such as:

- the reference structures, the persons who are responsible for it, the involved laws and other attributes of the proceeding, as the code with which it has been recorded, its deadline and average time of execution, initial and final events etc.;
- the competences defined in the declaratory judgments (to each competence that is assigned to a structure one or more proceedings can be associated);
- the determinant judgments produced by the structures (each determinant judgment is referred only to a single process).

Moreover, it should also be possible to indirectly connect the competences defined in the declaratory judgments with the matters of competence listed in the Statute for Autonomy of the PAT (to each matter of competence can correspond one or more competences defined in the declaratory judgments).

Once that this information has been brought together by creating the aforementioned associations, it should be possible to perform useful analyses as for instance the following:

- finding the list of proceedings associated to a single competence. This is useful in order to know the amount of proceedings that have to be transferred in case that one or more competences are moved from one structure to another;
- given the set of determinant judgments signed by a manager responsible for a structure, subdividing them on the basis of the proceedings and/or the competences. This is useful in order to estimate the impact that the management of a proceeding/competence has on the structure that manages this proceeding/competence;
- analyzing the matters of competence of the Statute of Autonomy assigned to other institutions (like municipalities) and subdividing them based on the

proceedings and/or competences. This is useful in order to estimate the impact that the management of the matter of competence has for the institution that has it in charge in terms of proceedings and competences it has assigned.

Furthermore, to each structure in the flowchart of the PAT a number of relations will be activated, namely with a structure type (Department, Service, Office...), the responsible manager, super/sub-structures, a declaratory judgment in which are specified all the competences and the corresponding assignment's decree by the President of the Province.

3.2 Important aspects of organizations emerging from the case study

The relevance of Tutto-In-Uno in the context of the MOSTRO project is linked to the fact that some features of organizations need to be represented in order to allow the system to answer some complex queries. In particular, since Tutto-In-Uno manages legal documents concerning how the PAT is decomposed and decomposable in substructures and it relies on the knowledge about the competences and responsibilities these substructures have, a quite complex model of organizations is required. The most interesting aspect pointed out by this case study is the importance of having a model that can be quite general but must integrate different features of organizations that in the formal models present in the literature, are normally only separately, and often quite superficially, considered. In the following we will deal with some of these aspects in detail.

Hence, from MOSTRO's point of view, the scenario of the structuring process of a real and complex organization like the PAT is used in order to, on the one hand, test and refine the organizations' model and the methodology developed in MOSTRO and, on the other hand, to highlight additional aspects of organizations that have been neglected in the first phase of the analysis conducted in MOSTRO. From the Tutto-In-Uno point of view, the opportunity of relying the whole system on an ontologically well founded model allows for a more open, maintainable, and flexible system, while the formal characterization of the involved notions increases the reasoning and deductive power of the system.

Up to now we are still in the first phase, i.e. we are using the case study to drive the MOSTRO model of organizations. In particular we tried to restrict the focus of the model to some notions that are general enough to achieve both an easily operative/applicative template and a common framework in which all the specific notions studied in the MOSTRO Deliverable 1 can find a place. The latter point is particularly important for a 'modular' development of the MOSTRO ontology of

organizations. The common framework gives a unitary picture of how the specific notions involved in the organizations modeling can be 'assembled'. Therefore, these notions can be deeply characterized in a more independent way (even though they are intertwined) as long as they are necessary for representing some aspects needed in the case study.

More specifically, the Tutto-In-Uno project highlights two important features of organizations that drove the proposal presented in the MOSTRO Deliverable 2: (a) the *organization chart* and (b) the *design*.

(a) The organization chart relies on the distinction of three kinds of entities: *person*, *position*, and *structure*. Structures are organization units that can be constituted by other (sub-) structures. To each position some responsibilities of the structure this position is associated with are assigned. Persons occupy positions that are assigned to them via a formal act. While the distinction between positions (called *roles* in other approaches) and persons (or, more generally, *agents*) is quite common in organization modeling and it is present in Tropos, the idea that organizations are structured in sub-organizations without being, at least in general, reducible to them (and to the positions associated to them either) is quite new. This structural aspect heavily drove our model and indicates a possible extension of Tropos. A less important, but interesting, point concerns the fact that responsibilities are *assigned* to positions (roles) and, indirectly, to persons that occupy these positions. I.e. in addition to contracts that establish which persons occupy which positions, here we consider that persons can be assigned to positions relying on a sort of 'meta-contract' that allows the organization to operate in this way.

(b) Most of the documents considered in the case study (determinant and declaratory judgments) concern the decisions taken by some organizations about other organizations or positions. In addition to assignments of persons to positions, these report creations of sub-organizations (structures in the Tutto-In-Uno terminology), assignments of competences to sub-organizations, and delegations (to some positions) of the creation of new sub-organizations. These legal judgments contribute to the *design* of the organization, i.e. they establish how the organization is structured and who is responsible of the resultant structures. Note that, due to the fact that some documents are linked to a specific design steps, it is important to keep track of the way a specific structuring of an organization has been obtained. This means that, in addition to the final structured organization, the design process and the responsibilities involved in each design step need to be represented in the model. The knowledge about the design (that is quite neglected in the literature of organizations) makes possible, at least in principle, to deal with the *dynamic* of organizations. Changes in the structure of organizations can be due to a new design step that has been introduced and decided by (a group of) persons that occupy positions that are entitled to act at the design level. The notion of design

step is then central and it has been introduced in the proposed model. Note that in Tropos this information is not explicitly encoded in the model, and therefore this is another area on which the methodology can be extended. Much more complex is the representation of the situation in which the design of an organization envisages the changes in the design itself. This situation is quite similar to the one of norms that contain other norms that regulate how the former can be changed. This is a really challenging topic that right now has not been addressed yet.

There are other interesting aspects of organizations emerging from the case study still to be addressed yet. In particular:

- *Competences*. We already said that competences are assigned to sub-organizations, therefore they are an essential part of the definition of positions, offices, departments, etc. But which is their ontological nature is not clear, even though we can give some minimal characterizations. Differently from goals and abilities, competences have a public dimension, they are assigned from the exterior and they have deontic implications (probably a mixture of obligations, permissions, delegations, etc.). In addition, they can require or enforce some abilities.

- *Administrative proceedings*. In order to fulfill some competences, an organization has at its disposal macro-activities that are regulated by the design itself. Macro-activities are decomposable in simpler ones that need to take place in agreement with the rules established at the design level. In addition to the classical problem of representing the decomposition of complex tasks in simpler ones that can be performed in parallel or sequentially, here the laws that regulate the administrative proceedings and the application of eventual sanctions need to be taken into account.

- *Kinds of organizations*. The model proposed is very general, therefore it does not define different kinds of organizations. Nevertheless, the case study distinguishes at least three kinds of organizations in the PAT: *office*, *departement*, *service*. One basic idea that has been already followed in the literature is to characterize the different kinds of organizations on the basis of their structure. Here probably we need to add also the kinds of competences the organizations have and how they are situated in the environment (especially interesting is the relation they have with citizens). Another interesting thing is to understand whether the design process itself characterizes the organization.

3.3 Conclusions

We have seen that the Tutto-In-Uno project highlights some interesting features of the representation of organizations. This is a necessary step to deal with the security of organizations. But, even though a definition of security seems really difficult, it is at least possible to envisage what information can be used in order to analyze some aspects of organizations linked to their security.

The structure and the way competences are assigned to the components of the organizations represent an important aspect for security. This means that security, similarly to other non-functional requirements as cost, performance, etc. that in Tropos are often represented as 'soft goals', are not analyzable without considering the whole organization, i.e. they are not deducible solely from the components but the way these components are linked together needs also to be analyzed. Consequently, one of the goals of the Tutto-In-Uno project is the evaluation of the impact of change not only in terms of some macro parameters (cost, security, etc.) but also in terms of how much the actual system of laws that regulate the organization needs to be modified. In this sense the information linked to the design and the dynamic of an organization can be of great value.

Chapter 4

Conclusions

In this deliverable, we have presented the two case studies that will be used throughout the remainder of MOSTRO project. For each of them, we outlined a description of the reference context, showed the limitations of the current approach, and identified areas where the application of MOSTRO-related techniques could be useful.

The first case study concerns modeling e-voting processes: the current approach applied to this case study involves UML modeling, which can describe quite precisely *how* the process is organized, but fails in showing *why* it is organized in this way. MOSTRO can contribute by using goal reasoning techniques, such as goal modeling proposed in [3]. The second case study, Tutto-In-Uno, is related to the description of the PAT structure, showing that the proper modeling of organizations is important and, in particular, is fundamental when reasoning about security concerns. Tutto-In-Uno shows that non-functional requirements (such as security, for example) cannot be analyzed without a complete representation of the organization.

The next activities in MOSTRO will aim to face these issues by proposing a methodology for modeling and analyzing security properties in organizations, by combining in some way a complete representation of the organization and the analytical tools Tropos proposes.

Bibliography

[1] ProVotE project. http://sra.itc.it/provote/.

[2] Testo unico delle leggi regionali sulla composizione ed elezione degli organi delle amministrazioni comunali. DPReg. n.1/L, 1 febbraio 2005.

[3] P. Bresciani, P. Giorgini, F. Giunchiglia, J. Mylopoulos, and A. Perini. Tropos: An agent-oriented software development methodology. *JAAMAS*, 8(3):203–236, 2004.

[4] V. Bryl, R. Ferrario, A. Mattioli, and A. Villafiorita. Evaluating Procedural Alternatives in an e-Voting Domain: Lesson Learned. Technical Report DIT-07-005, University of Trento, 2007.

[5] L. Caporusso, C. Buzzi, F. Giolo, P. Peri, and F. Sartori. Transition to electronic voting and citizen participation. In R. Krimmer, editor, *Electronic Voting 2006*, GI Lecture Notes in Informatics, pages 191–200, 2006.

[6] L. K. Chung, B. A. Nixon, E. Yu, and J. Mylopoulos. *Non-Functional Requirements in Software Engineering*. Kluwer Publishing, 2000.

[7] Comune di San Benedetto del Tronto, MET Informatica. Preliminar report on the electronic voting expreimentation (sperimentazione sulla votazione elettronica – preliminare). In Italian. Available at http://www.comune.san-benedetto-del-tronto.ap.it/ePoll/rl00.html.

[8] E-Poll. Electronic polling system for remote voting operations. Available at http://www.e-poll-project.net/.

[9] EAC. Voting systems performance and test standards, 2002. Available at http://www.eac.gov/election_resources/vss.html.

[10] P. Giorgini, J. Mylopoulos, E. Nicchiarelli, and R. Sebastiani. Reasoning with Goal Models. In *Proceedings of the 21nd International Conference on Conceptual Modeling*, pages 167–181, 2002.

[11] Governo Italiano. *European Elections 2004, Automated Counting of the Votes (Elezioni Europee 2004, Conteggio Automatizzato del Voto)*. In Italian. Available at `http://www.governo.it/GovernoInforma/Dossier/voto_conteggio`.

[12] D. Gross and E. S. K. Yu. From non-functional requirements to design through patterns. *Requirements Engineering*, 6(1):18–36, 2001.

[13] A. Mattioli. Analisi dei Processi Elettorali in ambito di voto elettronico per le Elezioni in Provincia di Trento, 2006.

[14] M. McGaley and J. McCarthy. Transparency and e-voting: Democratic vs. commercial interests. In *the International Workshop on Electronic Voting in Europe*, 2004.

[15] R. T. Mercuri and L. J. Camp. The code of elections. *Communications of the ACM*, 47(10):53–57, 2004.

[16] A. Ostveen and P. van den Besselaar. Security as belief – user's perceptions on the security of electronic voting systems. In *the International Workshop on Electronic Voting in Europe*, 2004.

[17] I. Sommerville. *Software engineering (7th ed.)*. Addison-Wesley, 2004.

[18] Venice Commission - European Commission for Democracy Through Law (2004). Report on the compatibility of remote voting and electronic voting with the standards of the Council of Europe adopted by the Venice Commission. Available at `http://venice.coe.int`.

[19] A. Villafiorita and G. Fasanelli. Transitioning to eVoting: the ProVotE project and Trentino's experience. 2006. To appear in the Proceedings of EGOV-06.

[20] M. Weiss, D. Amyot, and G. Mussbacher. Formalizing architectural patterns with the goal-oriented requirement language. In *Proceedings of VikingPLoP 2006*, September 2006.