# Wiki-based conceptual modeling:
# an experience with the Public Administration

Cristiano Casagni[1], Chiara Di Francescomarino[2], Mauro Dragoni[2], Licia Fiorentini[3], Luca Franci[3], Matteo Gerosa[2], Chiara Ghidini[2], Federica Rizzoli[3], Marco Rospocher[2], Anna Rovella[5], Luciano Serafini[2], Stefania Sparaco[4], and Alessandro Tabarroni[3]

[1] Polo Archivistico Regionale, Istituto per i beni artistici, culturali e naturali della Regione Emilia-Romagna, Viale Aldo Moro 64 - Bologna, Italy
[2] FBK-irst, Via Sommarive 18 Povo, I-38123,Trento, Italy
[3] SCS Consulting, Via Marco Emilio Lepido182/3, I-40132, Bologna, Italy
[4] Servizio sviluppo amministrazione digitale e sistemi informativi geografici, Regione Emilia-Romagna, Viale Silvani 4/3, I-40122, Bologna, Italy
[5] Dipartimento Sistemi di Produzione, Consiglio Nazionale delle Ricerche, Italy

**Abstract.** The dematerialization of documents produced within the Public Administration (PA) represents a key contribution that Information and Communication Technology can provide towards the modernization of services within the PA. The availability of proper and precise models of the administrative procedures, and of the specific "entities" related to these procedures, such as the *documents* involved in the procedures or the *organizational roles* performing the activities, is an important step towards both (1) the replacement of paper-based procedures with electronic-based ones, and (2) the definition of guidelines and functions needed to safely store, catalogue, manage and retrieve in an appropriate archival system the electronic documents produced within the PA. In this paper we report the experience of *customizing* a semantic wiki based tool (MoKi) for the modeling of administrative procedures (processes) and their related "entities" (ontologies). The tool has been *used* and *evaluated* by several domain experts from different Italian regions in the context of a national project. This experience, and the reported evaluation, highlight the potential and criticality of using semantic wiki-based tools for the modeling of complex domains composed of processes and ontologies in a real setting.

## 1   Introduction

In the last few years, the Public Administrations (PA) of several countries around the world have invested effort and resources into modernizing their services, in order to improve labor productivity as well as PA efficiency and transparency. The recent contributions and developments in ICT (Information and Communication Technology) can boost this modernization process, as shown by the dematerialization of documents produced within a PA. The availability of proper and precise models of the administrative procedures of the PA and of specific "entities" related to these procedures, such as the *documents* involved in the procedures or the *organizational roles* performing the activities, is a a key factor towards both (1) the re-design of the administrative procedures in

order to replace paper-based documents with electronic-based ones, and (2) the definition of guidelines and functions needed to safely store, catalogue, manage and retrieve in an appropriate archival system the electronic documents produced within the PA. The definition of these models requires the collaborative interplay of several actors with different competencies:

– *specific knowledge of the administrative procedures and their related documents in different domains*. Examples are the administrative procedures for business-to-government purchase and sale of goods and services, the ones for the management of personnel, those for the services that the PA offers to individual citizens, and so on. This knowledge is provided by domain experts working in the PA.
– *specific knowledge in archival science*. This knowledge is needed to identify what aspects of the administrative procedures have to be modeled in order to design the functionalities of an appropriate document management system. This knowledge is provided by experts in archival science.
– *specific knowledge in conceptual modeling (including process modeling)*. This knowledge is necessary to help the construction of proper, precise and unambiguous models that can facilitate the analysis of the procedures, and of the "entities" related to these procedures. This knowledge is provided by knowledge engineers.

In this paper we report the experience of making these three groups of actors share their competences and collaborate in the modeling activities using the wiki-based MoKi tool in the context of the ProDe Italian national project. The reasons behind the choice of MoKi were its ability to involve domain experts in the modeling process as well as its ability to model both procedural aspects (the administrative procedures) and ontological aspects (the "entities" related to the procedures) in an integrated manner. While the general version of MoKi enables people to model generic processes and ontologies, a customization of the tool (ProDeMoKi) was developed for ProDe, in order to guide the domain experts working in the PA in modeling precisely the elements of the domain at hand. Thus, the entire modeling process we report in this paper consists on:

– an identification of the main entities to be modeled and their relations (conceptual schema). This activity was driven by experts in archival science, with the help of knowledge engineers;
– a customization of MoKi for building models coherent with the conceptual schema proposed, which led to the development of ProDeMoKi. This activity was driven by knowledge engineers, with feedback from experts in archival science;
– the final modeling activity, which was performed by domain experts from the PA with some supervision from knowledge engineers and experts in archival science.

The contribution of the paper is therefore twofold: (1) it provides an empirical evidence of how to customize a generic wiki-based modeling tool for a specific complex scenario (Section 4) on the basis of its conceptual description (Section 2.1); and, (2) it provides an evaluation of the main features of the tool (Section 5) and an analysis of the lessons learned. This experience, and the reported evaluation, highlight the potential and criticality of using semantic wiki-based tools for the modeling of complex domains composed of processes and ontologies in a real setting, and can provide the basis for future customizations of wiki-based modeling tools to specific complex domains.

## 2 The ProDe project

ProDe[6] is an Italian inter-regional project with the aim of defining a national reference model for the management of electronic documentation (dematerialized document) in the Public Administration. This reference model follows an archival science perspective, and can be used for the identification of guidelines and functions needed to safely store, classify, manage, and retrieve, electronic documents produced within the PA in an archival system. The project has a duration of 30 months, from May 2010 to October 2012, and is composed of 11 tasks assigned to 11 teams (*task-teams*) coming from 10 regions (Piemonte, Lombardia, Liguria, Emilia-Romagna, Marche, Abruzzo, Campania, Puglia, Sicilia and Trentino). The 11 tasks are divided in 4 **central tasks**, that represent the core part of the project and are in charge of guiding the activities on specific topics such as document management and digital preservation; and 7 **peripheral tasks**, that provide the specific expertise on specific sectors of the PA (e.g., the administrative procedures for business-to-government purchase and sale of goods and services, the ones for the management of personnel). Thus, the central tasks provided the main expertise in archival science, while the peripheral tasks provided domain expertise in different fields of the PA.

### 2.1 Modeling flows of documents: the Conceptual Schema

In this section we illustrate the conceptual schema that was used in the customization of the ProDeMoKi and was proposed to the domain experts to guide the modeling of their administrative procedures. This conceptual schema, whose simplified version is graphically depicted in Figure 1 using an Entity-Relationship notation, was developed by the experts in archival, computer, and organizational sciences working in the central tasks of the ProDe project.
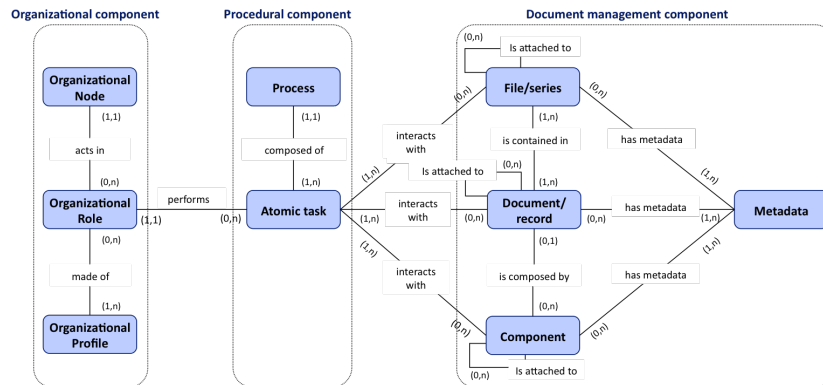


Fig. 1: The conceptual schema.

---

[6] http://www.progettoprode.it/Home.aspx

As shown in Figure 1, the entities of the model can be clustered in three different components, each of which plays an important role in the production and management of documents in administrative procedures:

1. **Document management** component. This component describes the archiving-related aspects of the domain and constitutes the central part of the model.
2. **Procedural** component. This component describes the activities which produce (manage, consume) documents.
3. **Organizational structure** component. This component describes the structure of the offices involved in the dematerialized procedures and the different profiles which interact with documents, possibly with different permissions.

*The document management component* The entities *document* and *record*[7], hereafter named document/record, or, for the sake of simplicity only document, constitute the central entities of the conceptual model. In ProDe, the description of the document/record entity is mainly devoted to the life-cycle of the document, which consists of the 5 following actions: **Create** (how the document is created); **Capture** (how the document is acquired in the document management system); **Manage** (how the document is managed inside the document management system); **Store/preserve** (how the document is stored in the document management system and preserved in the long run); and **Deliver** (how the document is distributed and made available by the system). These actions are used to identify the services and functionalities needed in the document management system to handle documents and records in a correct and appropriate manner. Furthermore, the description of document/record is characterized by a set of attributes such as the name of the document/record, the type of document, the origin and destination of the entity, and so on. The entity *component* identifies the set of bits which (possibly together with other components) composes a document/record. Moreover, each document/record is classified according to a filing plan and inserted in an appropriate *file/series*. According to the type of file/series, different criteria of management, storage, preservation and access can be granted to the document/record. Finally, *Metadata* are used to provide information about the context, content, structure, and management of the different entities described so far. To support the construction of an homogeneous model and the compatibility with the main standards for metadata such as Moreq2[8] and EAD[9], a common dictionary of metadata was also provided by the central teams of the project, and inserted in ProDeMoKi.

*The procedural component* Usually a business process is composed of a set of related activities, which happen inside an organization and transform resources in products or services for an (internal or external) customer. Within the ProDe project the entity *process* has been used also to include all those (complex) activities carried out by document

---

[7] By *record* we refer to an archival document, in a final and correct state, registered into a document management system and not modifiable or deletable. The only operation possible on a record is the modification of its metadata. By *document* we instead refer to an artifact which still requires modifications and is amenable to cancellation.

[8] http://www.dlmforum.eu/

[9] http://www.loc.gov/ead/

management systems. *Atomic task* is instead used to describe an atomic action within a process. The relations with other entities of the document management component emphasize the fact that activities can perform actions over these entities (such as the creation of a document or the modification of a file). The relations with the organizational role entity in the organizational structure component emphasize the fact that actions are performed by specific roles within the organization.

*The organizational structure component*  The main entity of the organizational structure is the *organizational role*. A role refers to an *organizational node*, which defines the atomic component of an organization (e.g., finance office), and is associated to an *organizational profile*, which instead defines the different profiles of permissions available within the organization.

## 3   The MoKi Architecture and Tool

MoKi[10] is a collaborative MediaWiki-based [8] tool for modeling ontological and procedural knowledge. The main idea behind MoKi is to associate a wiki page, containing both unstructured and structured information, to each entity of the ontology and process model. From a high level perspective, the main features of MoKi[11] are:

- the capability to model different types of conceptual models in an integrated manner. In particular the current version of MoKi is tailored to the integrated modeling of ontological and procedural knowledge;
- the capability to support on-line collaboration between members of the modeling team, including collaboration between domain experts and knowledge engineers.

These features have been proved extremely important in the context of the ProDe project. In fact, as we can see from the ER model depicted in Figure 1, the scenario addressed in the ProDe project required the modeling of administrative procedures, usually better described using a business process modeling notation, enriched with knowledge which typically resides in an ontology, such as the classification of document types, organizational roles, and so on. Moreover, the modeling team was composed by an heterogeneous group of domain experts and knowledge engineers situated in different Italian geographical regions. In the following we illustrate how these features are realized in the generic MoKi architecture. In Section 4 we illustrate the ad-hoc customization we performed for the ProDe project and how these features were realized for the specific MoKi used in ProDe.

*Modeling integrated ontological and procedural knowledge*  The capability of modeling integrated ontological and procedural knowledge is based on two different characteristics of MoKi. First of all, MoKi associates a wiki page to each *concept*, *property*, and *individual* in the ontology, and to each (complex or atomic) *process* in the process model. Special pages enable to visualize (edit) the ontology and process models

---

[10] See http://moki.fbk.eu

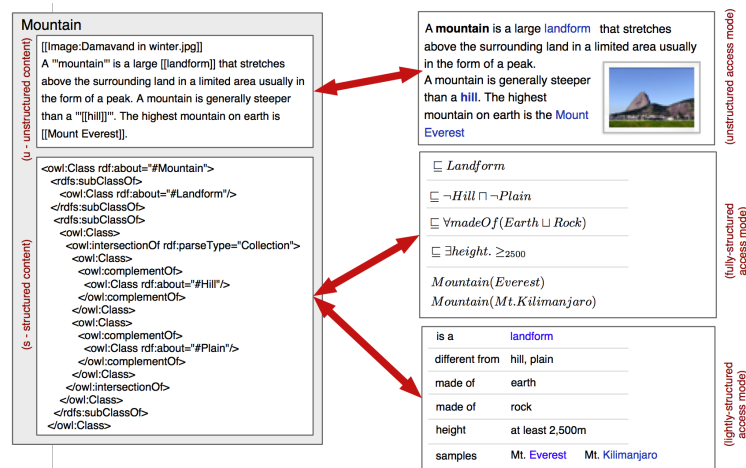[11] A comprehensive description of MoKi can be found in [6].

Fig. 2: Multi-mode access to a wiki page.

organized according to the generalization and the aggregation/decomposition dimensions respectively. The ontological entities are described in Web Ontology Language (OWL [11]), while the process entities are described in Business Process Modeling Notation (BPMN [9]). Second, MoKi has extended the functionalities of the BPMN Oryx editor [3], used in MoKi to represent BPMN diagrams, to annotate tasks with concepts described in the ontology, or to incorporate data objects formalized in the ontology. The integrated procedural and ontological knowledge is then exported in a comprehensive OWL model following the approach described in [4].

*Supporting collaboration between domain experts and knowledge engineers* MoKi is an on-line tool based on MediaWiki, thus inheriting all the collaborative features provided by it. In addition MoKi facilitates the collaboration between domain experts and knowledge engineers by providing different access modes to the elements described on the model, as illustrated in Figure 2 for the ontology concept "Mountain".

MoKi allows to store both *unstructured* and *structured* descriptions of the elements of the models, as shown on the left hand side of Figure 2. The unstructured part contains a rich and often exhaustive description of knowledge better suited to humans, usually provided with linguistic and pictorial instruments. Instead, the structured part is the one which is used to provide the portion of knowledge which will be directly encoded in the modeling language used to describe the specific element (OWL in the case of the concept "Mountain"). The advantage of storing the unstructured and structured descriptions in MoKi is twofold. First, informal descriptions are usually used to provide the initial description upon which the formal model is built, and to document the elements of the model (e.g., for future access and revisions). Storing the unstructured and structured descriptions in the same tool can facilitate the interplay between these parts. Second, domain experts, who usually create, describe, and review knowledge at a rather informal/human intelligible level, may find the unstructured part their preferred portion

of page where to describe knowledge, while knowledge engineers should be mainly focused on the descriptions contained in the structured part. Nevertheless, by using the same tool and accessing the same pages, all of them can be notified of what the others are focused at. Moreover, the discussion facilities of wikis, together with special fields for comments, can be used by both roles to discuss on specific parts of the model.

The organization of a page in an unstructured and a structured part is a first important collaborative feature, but may not be enough in the case of complex conceptual modeling languages, such as OWL or BPMN. In this case the structured part of the page will contain very precise, and often logic based, descriptions of the knowledge, preventing domain experts from accessing the domain knowledge encoded in the conceptual model. To overcome this problem, MoKi associates different *access modes* to each part of the page, as depicted in the right hand side of Figure 2. The current general version of MoKi is based on three different access modes:

– an *unstructured access mode* to view/edit the unstructured content;
– a *fully-structured access mode* to view/edit the complete structured content; and
– a *lightly-structured access mode* to view/edit (part of) the structured content via simple templates.

As shown in Figure 2, the access mode to the unstructured part can be provided by means of the regular view/edit facilities of wikis, while the access to the structured content can be provided by means of two different modes: one based on a translation of the OWL content in, e.g., DL axioms or in the Manchester OWL syntax, and another based on a structured, but semi-formal rendering of the OWL content in a pre-defined template. This way, the knowledge engineers can formally describe the ontology concept "Mountain" in OWL by using a highly formal access mode, while the domain experts can access a simplified version of the same content using a different, simpler, mode. A similar structure is provided also for the description of BPMN elements. By providing distinct modalities to access the structured content of a wiki page domain experts can not only have access to the knowledge inserted by knowledge engineers, but can also comment or directly modify part of it. Therefore, the design of appropriate access modes, according to the conceptual modeling language used and the degree of complexity handled by the domain experts, is a key aspect of the implementation of a wiki-based tool for conceptual modeling.

## 4   The MoKi Customization for the ProDe Project

ProDeMoKi[12] is the customization of MoKi that has been developed for the ProDe project. The general version of the tool has been adapted to support both the representation of the conceptual models needed in the project, and the skills of the modeling actors, who had a good expertise in the design of business processes, but had no experience in ontology design. Therefore, a first personalization of MoKi consisted in using only the unstructured and the lightly-structured access modes for the definition of ontological entities (the ones in the Document management component, and Organizational

---

[12] Available at https://dkmtools.fbk.eu/moki/prode/tryitout/index.php/Main_Page

Fig. 3: The template used to insert document information.

structure component in Figure 1), and only the unstructured and the fully-structured access modes for the modeling of the procedural component.

The second, important, personalization involved the templates used in the lightly-structured access mode of the ontological entities. In particular we have created an ad-hoc template for each entity shown in 1. In this paper we focus on the template for the document entity, whose main parts are shown in Figure 3, as it provides a representative and exhaustive example of the customizations implemented in ProDeMoKi.

The *Attributes* box in Figure 3 allows the insertion of general information through the first four text areas. Besides this general information, the user is able to insert the relations between a document and the organizational nodes who perform the actions which involve a document. The bottom part describes the relations between a document and the file/series that contain it. The *Document life-cycle* box in Figure 3 is focused on the steps which involve a document during a process, from its creation to its preservation in the document management system; for example a document may be classified in file/series during an "Acquisition" task and it may be electronically signed during a "Management" task. The user is able to describe these actions in the text boxes contained in the template, and to define the metadata that are required in each task via the "has-metadata" relation. Finally, the *Relations* box in Figure 3 is used to specify additional relations between the document and other entities in the ER model, namely, the "is-a" relation with other documents; the "is-attached-to"relation which expresses the fact that a document may be an annex of another document; the "has-metadata" relation which is used to express metadata that hold for the document in general, independently
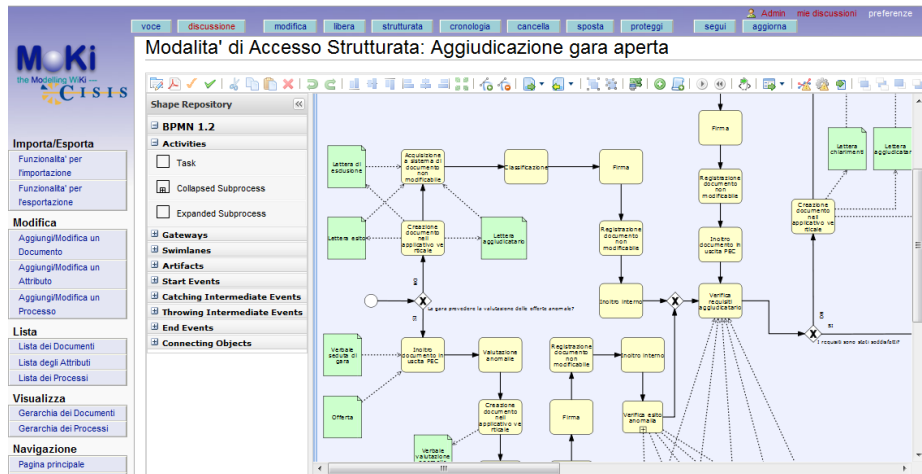
Fig. 4: The BPMN diagram enriched with documents.

from the specific phase in the lifecycle; and the "is-composed-by" which expresses the fact that a document is composed of a certain set of bits in a certain electronic format (e.g., a pdf file).

By combining the general features of MoKi and the customized ones discussed in this section, ProDeMoKi enables the following macro-functionalities: *model overview*, *model navigation*, and *entity modeling* (i.e., creation, revision, deletion and renaming). In detail, the global view of the model is provided both in the form of an unstructured list and of a hierarchical taxonomy of documents and processes. Similarly, the model can be navigated starting from an unstructured list, a hierarchical view, and also directly from the graphical representation of BPMN diagrams, enriched with data objects representing documents taken from the ontology, as shown in Figure 4[13]. Such a graphical representation can also be exploited for modeling processes and documents, together with the side-bar "add" commands (for both documents and processes) and the "add"commands available in the hierarchical view (only for documents).

The customization of MoKi required about 1 person-week for the definition of the conceptual model and 3 person-weeks for the implementation and testing of ProDeMoKi.

## 5   Evaluation

With the aim of evaluating the usage of ProDeMoKi for supporting domain experts in the collaborative modeling of specific knowledge, we investigated the following research questions:

– *(RQ1)* Is ProDeMoKi easy to use for domain experts?

---

[13] This feature has been added in ProDeMoKi in January 2011. Documents are represented by means of green data objects.

- (*RQ2*) Is ProDeMoKi useful for collaboratively modeling domain knowledge?
- (*RQ3*) Are all the provided views useful or is there a "best" view among the different interface views provided by ProDeMoKi for: (a) getting the model overview? (b) navigating the model? (c) creating new entities?

In order to answer these questions we performed two types of analysis: a quantitative and a qualitative one. In the former, data about tasks performed by ProDeMoKi users in a use case have been analyzed, while in the latter, ProDeMoKi users have been asked to answer a questionnaire in order to capture their perceptions about the ease of use and usefulness of the tool according to their experience.

*The Use Case* The considered use case consists in the actual usage of ProDeMoKi in the first phase of the ProDe project, where participating regions have been asked to produce two types of models: the "Modello di riferimento" and "Modello di gestione". While the "Modello di riferimento" is an abstract model of administrative procedures and documents, i.e., a kind of meta-model of concrete models, the "Modello di gestione" refines processes of the "Modello di riferimento" into concrete and more specific tasks, thus creating a link with the software applications to be used in the process realization. The ProDeMoKi users involved are PA employees distributed across the 7 peripheral tasks of the project. All of them are domain experts, i.e., they have knowledge of the PA domain and had previous experiences in the analysis of administrative procedures and documents as well as in documentation drafting. However, not all the users had the opportunity to model processes before and none to use ontologies. Before the beginning of the modeling activities (February 2011), the PA employees have been trained with a learning session, in which all the features of ProDeMoKi have been illustrated, and hands-on exercises have been proposed.

*The Questionnaire* The PA employees have been asked to fill an on-line questionnaire, requiring about 15 minutes and including 31 questions[14]. The questionnaire mainly aimed at investigating the users' background, their perception about three macro-functionalities of the tool (model overview, navigation and entity modeling), and their overall subjective evaluation about ProDeMoKi. Some of the questions were provided in the form of open questions, while most of them were closed questions. The latter mainly concern the user evaluation of the tool on a scale from 1 to 5 (e.g., 1 = *difficult to use*, ... , 5 = *intuitive to use*) or the direct comparison between two alternatives (e.g., 1 = *I prefer A*, 2 = *I prefer B*, and 3 = *I equally evaluate A and B*).

## 5.1 Quantitative Evaluation Results

We analyzed the data on the usage of ProDeMoKi (October 2010 - June 2011), that have been obtained by combining the information stored in the ProDeMoKi database, and the access logs of the web server hosting the tool. Table 1 shows the overall number of page creations, revisions, deletions and renaming performed by the different task-teams in the

---

[14] Collected data are available at https://dkm.fbk.eu/index.php/ProDeMoKi_Evaluation_Resources.

| task-team | Page Creations | Page Revisions | Page Deletions | Page Renaming |
|---|---|---|---|---|
| t1 | 171 | 542 | 71 | 0 |
| t2 | 171 | 799 | 37 | 54 |
| t3 | 135 | 534 | 13 | 33 |
| t4 | 217 | 548 | 55 | 2 |
| t5 | 384 | 1065 | 58 | 7 |
| t6 | 103 | 316 | 16 | 0 |

Table 1: Usage of ProDeMoKi by the different *task-teams* per type of activity on pages



(a) Activities on pages per month     (b) Documents and processes per task-team
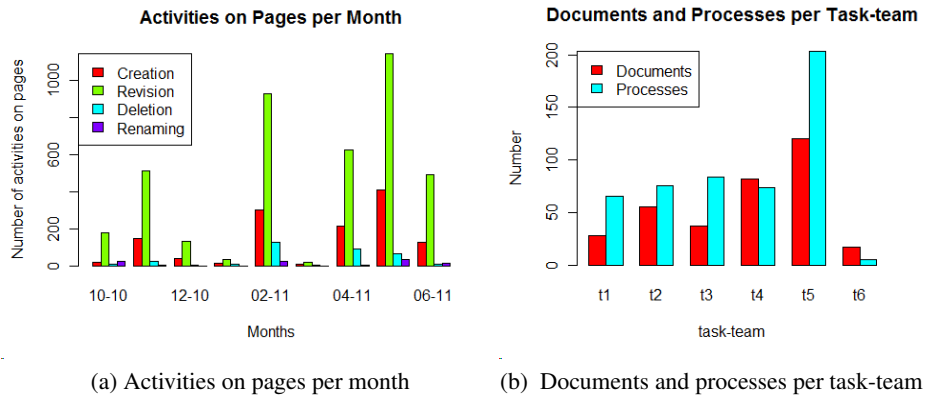
Fig. 5: ProDeMoKi usage

considered period[15]. The table shows that, though all the typologies of page activities have been exercised by users, there exists a trend in their distribution: as expected, the most frequent one is the page revision activity, followed by page creation and deletion. By looking at the distribution of the activities per month (Figure 5a)[16], we found that: (i) the general trend of activity typologies on pages is also monthly confirmed; (ii) there have been peaks of work in November 2010, February 2011 and May 2011. These can be partially justified by users' autonomous ProDeMoKi training after the beginning of the modeling activities (October 2010) and the learning session (February 2011), and by project internal deadlines (May/June 2011).

Currently, the two reference models contain overall 342 documents and 506 processes modeled with ProDeMoKi. Figure 5b shows their distribution per task-team. Most of the task-teams (4 out of 6) produced more processes than documents, thus remarking the importance of the procedural knowledge in domains such as the PA one. Furthermore, the number of documents used in processes' diagram (on average 4.2[17]) and the number of processes in which a document is used (on average 1.32), also reveals the

---

[15] The task-teams that are not required to use ProDeMoKi in this phase of the project are not reported in the table.

[16] The average number of daily activities on page is reported.

[17] The average number decreases to 0.6 documents per process if we consider also atomic processes, i.e. with an empty flow.

| Entity | Model Overview | | Model Navigation | | | |
|---|---|---|---|---|---|---|
| | List | Hierarchy | List | Hierarchy | Diagram | Textual Search |
| Doc.s | 503 | 239 | 341 | 155 | 11 | - |
| Proc.s | 939 | 626 | 758 | 478 | 55 | - |
| Total | 1042 | 865 | 1099 | 633 | 66 | 11 |

(a) Use frequency of model overview and navigation

| Document and Process Creation | | |
|---|---|---|
| | Sidebar | Diagram | Hierarchy |
| Doc.s | 113 | 275 | 61 |
| Proc.s | 90 | 580 | - |
| Total | 203 | 855 | 61 |

(b) Use frequency of entity creations

Table 2: Use frequency of ProDeMoKi views

strength of the relationship between these two types of knowledge, thus confirming our intuition about the importance of providing adequate means for their integration.

Moreover, by looking at the usage of (the commands available in) the different views provided by ProDeMoKi, it is possible to have an insight of their suitability for specific purposes, as well as of users' preferences. In detail, Table 2a summarizes the results obtained by investigating the different views available for the model overview and navigation, while Table 2b for the creation of new documents and processes. While in case of documents, the number of model visualizations and document accesses from the unstructured list is more than double with respect to the corresponding numbers in the hierarchical view, for processes also the hierarchical visualization plays an important role. On the other hand, while the graphical representation of both documents and processes in BPMN diagrams has not been extensively exercised by users for the model navigation (probably because of its late introduction in the tool, in January 2011), it represents the most used way for creating new documents and processes: it in fact allows to introduce high level entities directly in the process modeling phase and to detail them later on. Both these results seem to confirm our intuition about the importance of abstraction layers in process modeling and visualization.

Finally, we investigated the usage of a special functionality offered by the tool (provided by MediaWiki), the log history. Although it has not been frequently used (52 times in total), the 40% of times it has been exercised by the most productive (120 documents and 203 processes) and among the most numerous (4 modelers) task-teams, thus remarking its usefulness in case of large models and of collaborative works.

### 5.2 Qualitative Evaluation Results

We analyzed the data collected by means of the on-line questionnaire proposed to a total of 14 ProDeMoKi users in order to investigate their perception about the tool. Table 3 reports the information about their background knowledge, as well as about the time spent and the approach followed to learn the ProDeMoKi use. All the participants are pc habitual users: they use the pc almost everyday, mainly for writing and reading documents, for job (using office suites) and for navigating the Internet. They also had frequent occasions to visit wiki pages, while only half of them had the opportunity to (on average rarely) edit wiki pages. 93% of the employees had previous experience in analyzing administrative documents and procedures, while 21% had never modeled diagrams before. On average, the time spent for learning ProDeMoKi has been 1-2 days and the preferred learning approach the autonomous training.

| Property | Value | % | Property | Value | % |
|---|---|---|---|---|---|
| **Pc use frequency** | never | 0 | **Pc use purpose** | document reading and editing | 27.66 |
| | rarely | 0 | | office suites | 27.66 |
| | sometimes | 0 | | Internet browsing | 25.53 |
| | often | 0 | | programming | 17.02 |
| | always | 100 | | testing and customer care | 2.13 |
| **Wiki page consultation frequency** | never | 0 | **Wiki page editing frequency** | never | 50 |
| | rarely | 21.43 | | rarely | 42.86 |
| | sometimes | 28.57 | | sometimes | 7.14 |
| | often | 14.29 | | often | 0 |
| | always | 35.71 | | always | 0 |
| **Experience in domain analysis and documentation** | none | 7.14 | **Type of experience** | domain analysis and textual documentation drafting | 21.43 |
| | bad | 0 | | procedure analysis and diagram creation | 21.43 |
| | medium | 35.71 | | document analysis and documentation production | 0 |
| | good | 42.86 | | domain analysis as well as documentation and diagram production | 57.14 |
| | very good | 14.29 | | | |
| **Time spent for learning** | < 1 day | 36.36 | **Learning approach** | autonomous training | 72.73 |
| | < 2 days | 36.36 | | learning session | 9.09 |
| | 1 week | 27.27 | | tutorial | 0 |
| | 10 days | 0 | | talking with and asking colleagues | 18.18 |
| | > 10 days | 0 | | | |

Table 3: Users' background knowledge and learning approach

| **Model Overview** | | | | **Model Navigation** | | | |
|---|---|---|---|---|---|---|---|
| **Factor** | **View** | **Avg. rank** | **Std. dev.** | **Factor** | **View** | **Avg. rank** | **Std. dev.** |
| Ease of use | Hierarchy | 1.14 | 0.23 | Usefulness | Hierarchy | 2.00 | 0.74 |
| | List | 1.86 | 0.23 | | Diagram | 2.14 | 0.78 |
| Usefulness | Hierarchy | 1.14 | 0.23 | | Textual Search | 2.73 | 0.68 |
| | List | 1.86 | 0.23 | | List | 3.14 | 0.92 |

Table 4: Users' ranking of ProDeMoKi views for the model overview and navigation

In the questionnaire, we investigated the users' evaluation with respect to the three main ProDeMoKi macro-functionalities, with a special focus on the different alternative approaches provided by ProDeMoKi for their realization.

Table 4 (left) reports the results related to the users' average ranking of the two alternative views provided by ProDeMoKi (where 1 and 2 denote respectively the first and the second position in the ranking) for the model overview with respect to ease of use and usefulness[18]. Similarly, Table 4 (right) reports the average value (and the standard deviation) of the ranking provided by subjects about the usefulness of the four alternative views for the model navigation, where 1 represents the first position in the ranking and 4 the last one. The table shows that the hierarchical view is perceived by users as the most useful for both the model overview and navigation, while the list is considered as the least useful. A similar result is obtained also in the case of the comparison between the ease of use of the list and the hierarchical views. Moreover, the quality of the overall model navigability support provided by ProDeMoKi is also investigated: on average, it is perceived by users as more than reasonable.

Table 5a reports, instead, the subjective user perception, on a scale from 1 (very difficult) to 5 (intuitive), about the ease of performing modeling activities (i.e., creation, revision, deletion and renaming) on model entities. The table shows that on average users found entity creation easy, deletion between reasonable and easy, and revision and renaming more than reasonable.

---

[18] More details about the reliability of the agreement among users can be found in [5]

|  | Avg. | Std. dev. |
|---|---|---|
| **Entity Creation** | 4 | 0.89 |
| **Entity Revision** | 3.45 | 0.69 |
| **Entity Deletion** | 3.55 | 0.69 |
| **Entity Renaming** | 3.27 | 0.65 |

(a) Perceived ease of use of modeling activities

| **Document Creation** | | | | |
|---|---|---|---|---|
| **View** | **Ease of use** | | **Usefulness** | |
|  | Avg. rank | Std. dev. | Avg. rank | Std. dev. |
| Sidebar | 1.68 | 0.46 | 1.68 | 0.6 |
| Hierarchy | 2.05 | 0.27 | 2.14 | 0.45 |
| Diagram | 2.27 | 0.41 | 2.18 | 0.6 |
| **Process Creation** | | | | |
| **View** | **Ease of use** | | **Usefulness** | |
|  | Avg. rank | Std. dev. | Avg. rank | Std. dev. |
| Sidebar | 1.32 | 0.25 | 1.41 | 0.3 |
| Diagram | 1.68 | 0.25 | 1.59 | 0.3 |

(b) Users' ranking about ease of use and usefulness of alternative views for new entity creation

Table 5: ProDeMoKi entity modeling

|  | Average | Std. dev. | p-value |
|---|---|---|---|
| **Ease of use** | 3.36 | 0.67 | 7.442e-06 |
| **Usefulness** | 3.36 | 0.81 | 3.739e-05 |

Table 6: Perceived ease of use and usefulness of ProDeMoKi

Among the different activities, we focused in details on the entity creation, and we investigated the users' subjective ranking of ease of use and usefulness of the different alternatives for creating new documents/processes. Table 5b reports the average value and the standard deviation of the users' rankings, where 1, 2, and 3 are the three possible values provided by subjects for denoting the first, the second, and the third ranking position in case of documents, and 1 and 2 for processes. According to the subjective evaluation, the most useful and easy to use way for creating new documents and processes is by means of the sidebar commands, while the least useful and easy to use is the creation starting from the diagrammatical view.

Finally, Table 6 shows the results related to the subjective evaluation about the overall ease of use (resp. usefulness) of the tool reported by ProDeMoKi users on the base of a 5-point scale, where 1 means very difficult (resp. completely useless) and 5 intuitive (resp. absolutely useful). According to the users' answers, ProDeMoKi is easy to use and useful (the provided evaluation about ProDeMoKi is, on average, more than easy and more than fairly useful for the collaborative modeling). These positive results are also statistically relevant at 5% confidence level[19] (i.e., we have the 95% of confidence that the fact that the tool is perceived as easy to use/useful is not due to chance). Moreover, we found that there exists a strong positive correlation[20] between the subjective answers with respect to the ProDeMoKi usefulness for the collaborative work and the number of people involved in the team, thus strengthening the result related to usefulness of the tool for collaborative purposes.

---

[19] We applied a one-tailed Wilcoxon test and we obtained in both cases p-value<0.05.

[20] We performed a correlation analysis applying the Spearman's coefficient at 5 percent confidence level.

### 5.3 Discussion and Lesson Learned

The quantitative data collected demonstrate a concrete and almost continual experience of actual users, distributed all over Italian territory, in the use of the tool, thus making us confident about the reliability of their answers.

By looking at the results obtained with respect to the ease of use of the creation, editing and deletion on entities and at the overall ease of use of the tool, we can state that the users perceive the tool as more than easy to use. This result is also strengthened by the amount of time spent and the approach exploited for learning how to use the tool: 72% of employees spent only less than two days to learn how to use ProDeMoKi, and the same percentage learned it autonomously. We can hence positively answer *RQ1*.

Moreover, we observed that users positively perceive the overall usefulness of the tool for the collaborative modeling of documents and processes. The validity of this result is also confirmed by the fact that such a usefulness is perceived more strongly by employees working in teams having more than two persons (on average 3.8 for teams with more than two persons versus 2.8 for those with less than three). There exists, in fact, a correlation between the size of the subject's team and his/her feedback about the ProDeMoKi usefulness for collaborative purposes. We can hence positively answer *RQ2* too.

With respect to the alternative views provided by the interface we found that, while the hierarchical one is perceived by users as the most useful (and the list as the least useful one) both for the model overview and navigation, their frequency of use shows that the list is actually the most used for both the macro-functionalities. Similarly, the creation of documents and processes starting from the process view, though perceived as difficult to use and less useful with respect to the commands available in other views (i.e., the classical ones on the sidebar and in the hierarchical view), has been extensively used. A possible explanation to this is that more mechanical actions (as the model overview through the list of its artifacts) are usually the first to be executed and only later, if not useful, they are replaced by more specific ones. While these results suggest to better investigate the usability of those views that, although perceived as more useful and easy to use, are less used in practice, they also reveal the importance of having all the available alternative views. In conclusion, for all the three sub-questions of *RQ3*, we can state that all the views have their own usefulness.

Finally, some interesting suggestions for improving ProDeMoKi and making it more suitable to users' needs came from the answers to the open questions. While quite satisfied with the functionalities related to the model overview and navigation, users found some space of improvement for the functionalities related to the modeling, in particular to the document modeling (almost 50% of the users found the document modeling a weakness of the tool). Some of the proposed suggestions have already been implemented in a new prototypical version of the tool, while we plan to implement others (e.g., a command for duplicating entities) in the next versions. However, the tool has been positively judged with respect to its process modeling capability (identified as the ProDeMoKi major strength), as well as to its support for collaboration and modeling of complex domains, and its broad accessibility (only a web browser is needed).

# 6 Concluding Remarks

Several works have focused on the application of Semantic Web technologies to the PA, for instance in the areas of semantic services [12], reference models [10], data integration [1], and collaborative knowledge sharing [7]. Similarly several efforts aim at using semantic wikis for the collaborative construction and visualization of conceptual models [8, 6, 2]. In this paper we report our experience in applying semantic-based wiki technology for the specific modeling needs of a complex PA domain, in which administrative procedures and related "entities" are tangled. The concrete use of the ProDeMoKi tool by real domain experts and their subjective evaluation revealed that the tool is easy to use and useful for the collaborative work, though still open to improvements. In the future, we plan to enhance ProDeMoKi by implementing the users' suggestions and further investigating how to improve the usage of the tool (e.g., strengthening the training, enhancing or adding functionalities). Moreover we aim at validating the overall MoKi customization approach by extending the MoKi customization to other specific domains.

# References

1. Harith Alani, David Dupplaw, John Sheridan, Kieron O'Hara, John Darlington, Nigel Shadbolt, and Carol Tullo. Unlocking the potential of public sector information with semantic web technology. In *6th International Semantic Web Conference (ISWC)*, Busan, Korea, 2007.
2. S. Auer, S. Dietzold, and T. Riechert. Ontowiki - a tool for social, semantic collaboration. In *5th Int. Semantic Web Conference (ISWC)*, volume 4273, pages 736–749. Springer, 2006.
3. Gero Decker, Hagen Overdick, and Mathias Weske. Oryx — an open modeling platform for the bpm community. In *Proceedings of the 6th International Conference on Business Process Management*, BPM '08, pages 382–385, Berlin, Heidelberg, 2008. Springer-Verlag.
4. Chiara di Francescomarino, Chiara Ghidini, Marco Rospocher, Luciani Serafini, and Paolo Tonella. Semantically-aided business process modeling. In *8th Int. Semantic Web Conference (ISWC 2009)*, volume 5823 of *LNCS*, pages 114–129. Springer, 25-29 October 2009.
5. Mauro Dragoni, Chiara Ghidini, Marco Rospocher, Luciano Serafini, and Chiara Di Francescomarino. On the use and evaluation of a wiki-based tool. Technical report, FBK-IRST, Italy, 2011. https://dkm.fbk.eu/index.php/ProDeMoKi_Evaluation_Resources.
6. Chiara Ghidini, Marco Rospocher, and Luciano Serafini. Moki: a wiki-based conceptual modeling tool. In *ISWC 2010 Posters & Demonstrations Track: Collected Abstracts*, volume 658 of *CEUR Workshop Proceedings (CEUR-WS.org)*, pages 77– 80, Shanghai, China, 2010.
7. Bernhard Krabina. A semantic wiki on cooperation in public administration. In *7th International Semantic Web Conference (ISWC2008). Poster & Demo session.*, 2008.
8. Markus Krotzsch, Denny Vrandecic, and Max Volkel. Wikipedia and the semantic web - the missing links. In *Proc. of the 1st Int. Wikimedia Conference (Wikimania 2005)*.
9. OMG. Business process modeling notation, v1.1. www.omg.org/spec/BPMN/1.1/PDF.
10. Vassilios Peristeras and Konstantinos Tarabanis. Reengineering the public administration modus operandi through the use of reference domain models and semantic web service technologies. In *AAAI Spring Symposium on The Semantic Web meets eGovernment*, 2006.
11. Michael K. Smith, Chris Welty, and Deborah L. McGuinness. Owl web ontology language guide. W3C Recommendation, 10 February 2004.
12. Tomas Vitvar, Mick Kerrigan, Arnold van Overeem, Vassilios Peristeras, and Konstantinos Tarabanis. Infrastructure for the semantic pan-european e-government services. In *AAAI Spring Symposium on The Semantic Web meets eGovernment*, 2006.