Situated Multimodal Documents

Augusto Celentano, Fabio Pittarello
Dipartimento di Informatica, Università Ca' Foscari di Venezia
Via Torino 155, 30172 Mestre (VE), Italy
{auce,pitt}@dsi.unive.it

***Abstract*—The choices made by user in processing a set of documents is related, in a broad sense, to the sum of influences coming from the documents in the user situation, which does not include only physical or technical context parameters, but also the user goal and a set of personal components related to the user goal and to the effect on the user of the document. We use the term *situated document* to denote such influence. In this paper we propose a model for situated documents and propose a functional architecture for their management.**

I. Introduction

Context-awareness, information adaptation and user profiling, are common expressions in the design of personal information systems and applications. They are the basic concepts upon which ubiquitous computer systems build their behavior: the shape and content of information and the user interaction depend on the user *situation*, defined as a combination of the user preferences and history (the profile), the user physical state (location, time, environmental conditions), the hardware/software environment (the device context), the communication capabilities (network context), etc.

The word *situation* is used in an informal way to denote a complex set of interrelationships of observable parameters whose knowledge allows a software to change the information delivered and the interaction with the user in a range of predefined or algorithmically computed variants of a global scheme.

A further step is to consider the user situation (i.e., the context) in a holistic view: it is not (only) the sum of the facets related to profile, device, location, etc., but it is related (also) as a whole to the user physical environment: the environment supports the user in her/his tasks through a set of resources that the user identifies according to the goal to be reached.

We take in this paper the juridical domain as a main case study. Juridical acts require the consultation and the preparation of a set of documents related by precise rules and regulations. Juridical documents are drawn starting from a *template*, a document schema that contains a number of clauses describing the case; for any type of act some clauses are always present, other clauses depend on the juridical situation described or modified by the document, and on the user status.

Clauses contain variable application data which describe the instances of persons, estates, goods, rights, constraints and obligations that globally define the specific juridical act, for example, the names of the parties in a selling agreement, the public registry data of a real estate, the amount to be paid, the payment deadlines, the warrants, etc.

Documents can be prepared and delivered in multimodal forms and at different levels of detail, e.g. on paper, electronically, as schematic forms or free text, with multimedia attachments such as maps, photographs, copies of authenticated documents, verbal recordings, etc.

Such a document model is not limited to the juridical domain, but is typical of any domain where the content and the format of documents does not derive from discretionary choices but from well defined rules, and extend their effect on the user personal, social and economic environment.

The paper is organized as follows: after reviewing the relevant literature in Section 2, in Section 3 we define a model for situated documents. Section 4 describes an architecture for processing and delivering situated documents. Section 5 discusses an example related to the juridical domain, and Section 6 draws the conclusions.

II. Related work

Much work has been done in the area of document adaptation and context-aware systems. The initial concept of location as "the" user context has been broadened [1]–[4] to include several facets related to the user status, to the environment, to the communication, to the device, etc.

Several proposals exist for automatic adaptation of multimodal documents, but they are generally limited to some aspects of the context (e.g., location, device features, processing resources) and do not involve the whole process of document design and delivery [5]–[7]. More complex adaptation techniques introduce a higher level of adaptation which applies to the logical structure of the multimedia document, not only to its content, while preserving the meaning of the information delivered [8].

The notion of a *virtual document* as a model for describing the semantics of a multimodal document independently from the actual media content is discussed in a previous work by one of the co-authors of this paper [9].

In [10] adaptation is approached with a different viewpoint. Assuming that different combinations of media could convey the same meaning to the user, the authors present adaptation as the selection and combination of media which have the greatest cognitive impact on the user, according to psychology studies about the human perception.

In this paper we approach the problem of adapting a suite of documents to a user situation from the *situated cognition*

perspective [11]. The theory of situated cognition has received attention in the educational field because it provides numerous prescriptions for instructional practice. It relates the external world to be understood with the individual's perceptions of the world representation, through her/his symbolic interaction with it. Quoting [12], "in this context, learning is defined as the individual's ability to construct meaning by extracting and organizing information from a given environment," a concept that can well be extended to the activity of interpreting or producing documents.

Situated cognition has been studied as opposed to the *activity theory* model [13], based on the definition of action plans to reach anticipated goals. Relations between a planned activity and situated actions have been discussed, among others, in [14], [15]. In the domain of document processing the activity theory seems suitable in cases in which the user has a precise idea of the meaning and purpose of a set of documents related to some goal. With the growth of information processing capabilities, and the increase in variety of devices and media, the understanding of a document cannot be split from the interpretation of the context in which it is delivered to a user. Plans, therefore, should be viewed as general guidelines to build an interactive experience evolving according to the evolution of the user's situation.

III. Situated documents

In this section we formally define a model for situated documents. The model defines the relationships between a meta-level in which classes of documents are defined, covering in a broad sense and with different instantiation possibilities the requirements of a domain, and the concrete final documents describing a specific case for a user in a specific situation. The model, however, does not define all the details, simply denoting the occurrences and the roles of situations, document types and user data in the relationships among the model levels and components. A more precise analysis should necessarily be anchored to a specific application domain; examples from the juridical domain will help the reader to figure out how to bridge the model with actual document processing.

Definition 1 (Conceptual document): A conceptual document is a tuple $cd = \langle N, \mathcal{DT}, \mathcal{S}, R \rangle$ where

- N is the name of the conceptual document;
- $\mathcal{DT}$ is a set of document types $\{dt_1, \ldots, dt_n\}$ that define the different ways concrete documents fulfill the requirements for the conceptual document;
- $\mathcal{S}$ is a set of situations $\{s_1, \ldots, s_m\}$ in which the conceptual document can be used;
- R is a relation on $\mathcal{DT} \times \mathcal{S}$ defining which document type can be used in which situation.

A simple yet typical example of a conceptual document is a lecture note. It can be a written document, a summary file, a video or audio recording, a multimedia presentation, etc.. Independently from the actual content, all the types share the same overall meaning, expressed through different media and at different length of detail to account for different user needs.

As a more complex example, a notary deed[1] about a real estate transaction is a form of contract that establishes the rights of sellers and buyers of a property, and the state of the property itself after the transaction. As such it has a juridical value independent from the type of the concrete document which is produced, e.g., a paper deed or an electronic document. It has a number of clauses and variable data which depend on the actual property and parties. It is accompanied by a number of related documents, generically called by-products, which contain subsets of the data contained in the deed and delivered to different offices of the public administration such as the real-estate registry office, the tax offices, the municipality archives, etc. Some of the documents that make up a real estate transaction deed must exist as paper documents physically subscribed by the contractors and by the notary, subject to precise rules of style coming from law constraints. Other documents can appear in several formats: as forms, as free text documents, as files, as paper documents, and so on, according to the general and local regulations and the procedures used in a specific town or office.

In both examples the conceptual document bears a meaning towards the real world independent from its concrete shape[2] and defines the "essence" of the document.

Definition 2 (Conceptual document suite): A conceptual document suite is a tuple $cds = \langle \mathcal{CD}, P, start \rangle$ where

- $\mathcal{CD}$ is a set of conceptual documents $\{cd_1, \ldots, cd_n\}$, among which logical relations exist in a given application domain;
- P is a partial ordering on $\mathcal{CD}$;
- $start \in \mathcal{CD}$ is the *main* document of the suite.

Logical relations exist among the document of a suite for a given application domain. The main document is the conceptual document that is first needed by a user to fulfill her/his goal. The assumption that it is unique in the suite is done for simplicity; it can easily removed to account for cases in which different conceptual documents can start of the user activity, as long as a declarative or algorithmic description, or a user choice, can decide, in a given situation s_i, which starting document is appropriate.

The partial order relation P defines a *plan schema* for cds, i.e., a set of paths among conceptual documents ordered according the temporal and causal constraints defined in the application domain.

With reference to the examples illustrated above, two conceptual suites are the set of the lecture notes for a course and the set of juridical documents that pertain to a real estate selling act.

A conceptual document may belong to several suites. For example, a property act is needed for selling a house, but also for setting a mortgage on it without changing ownership.

[1]We implicitly refer to the Italian regulations, but the differences among countries are not relevant in the context of this discussion.

[2]The word *concrete* is used in a broad sense and includes also electronic versions of the document, which are at last contained in concrete supports.

In this paper the way the documents in a suite are actually identified and correlated is left unspecified, since it depends on the application domain. It is easy to devise a set of indexes, keys and attributes to discriminate and join documents and suites in a database, but a discussion on such issues exceeds the size and the scope of this paper.

It would also be possible to augment the definition for the conceptual document with a reference to the suites it belongs to, but the augmentation is only functional to the implementation of documents and suites, and is not needed at the definition level.

Definition 3 (Virtual document): A virtual document is an instance of a conceptual document with one specific type, for one specific situation, containing the data relevant for the situation according to the application domain requirements.

Given a conceptual document $cd = \langle N, \mathcal{DT}_N, \mathcal{S}_N, R_N \rangle$, a virtual document is defined as the tuple $vd = \langle N, dt, s, \mathcal{I} \rangle$, where

- $dt \in \mathcal{DT}_N$;
- $s \in \mathcal{S}_N$;
- $(dt, s) \in R_N$;
- $\mathcal{I}$ is the *information* contained in the document, i.e., the set of data that distinguish it from other instances as pertinent to the situation s.

A virtual document suite is a set of virtual documents $\mathcal{VD} = \{vd_1, \ldots, vd_n\}$ where each vd_i is an occurrence of a conceptual document of the same suite.

A virtual document is an instantiated document, in the sense that it contains the data necessary to represent the user situation at a level sufficient to reach the user goal. It is, however, not a final concrete document because the presentation and, possibly, a part of the content are not defined, since they depend on a local user context which, in the same user situation, may vary in details such as time, device, environmental conditions, and so on.

In relation to the communication of content, we may say that the suite of virtual documents is a multimodal set that enables such communication through different channels; each virtual document belonging to this suite will be a monomodal or multimodal document targeted at specific situations.

In the juridical domain, a notary deed for a real estate property sale can be a text document written by the notary with a personal prose style, containing all the data necessary to identify the parties and the object of the sale, or a form predefined by the registry office with fields for the same data, or an electronic record of the data, electronically signed and transmitted via a secure channel to the registry office. Such documents contain all data needed to completely describe the actual juridical case (hence the user situation) but, for example, the date and place of signature could be still unspecified and filled-in when the concrete final document is produced.

The virtual document is a *partially situated* document, in the sense that it reflects the situation in which it is produced or read, except possibly for details which do not change any of its meaning and validity. The situation is however completely defined only one step further, taking into account the actual presentation and delivery to the user.

Definition 4 (Situated document): A situated document sd is an adaptation of a virtual document vd to a set of variable parameters and data, known as a *context*, which integrate the content and change the physical appearance of the document, its coding, its delivery mode, the interaction with the content, without changing its semantics in terms of the applications domain.

We do not enter into details about the definition of the context, a concept deeply investigated in the literature, but still left in an aura of uncertainty as to the way to distinguish it from other data that enter into the document processing. In our model, the context is the part of the situation not related to the user goal, mood, cultural habit, etc. It is mostly related to the user physical environment which affects operations like reading, writing, sending and receiving information and documents.

The situated document add a further level of specification to the communication modalities associated to a specific monomodal or multimodal virtual document (e.g., specifying the visual codec for a video document or the standard for character representation for a text document).

Completing the examples introduced of this section, a virtual lecture note which is an audio recording can be coded with different codecs at different audio quality, and can be delivered via a streaming server or on a CD-ROM. A text lecture note can be a PDF file, a LaTeX source, and can be delivered as a whole or one page at a time according to the capabilities of the user equipment.

The electronic notary deed can be completed and formatted according to different requirements, defined by the receiving office as a function of the operating environment of the sender.

IV. An architecture for situated multimodal documents

The goal of the architecture presented in this section is to support the user in the creation and consultation of documents in a variety of situations, influenced by different types of context. The underlying idea is that the user should express her/his needs in terms of goals to be reached and the system should be able to help her/him to access and/or to create the documents s/he needs in the specific situation. The system should be aware and take care of the evolution of the situation, which includes both the user and the environment context, suggesting to the user the most appropriate documents to follow, in the format compliant with the situation constraints.

Figure 1 shows a high-level vision of the functional architecture of the system.

Several *suites of conceptual documents*, corresponding to different user goals, are stored in a database. Access to those suites is triggered by the statement of a specific user goal. We assume that the user has a *device* enabling her/him to interact with the system; the device, whose type is not relevant to this discussion, embeds sensors that permit the capture of a set of

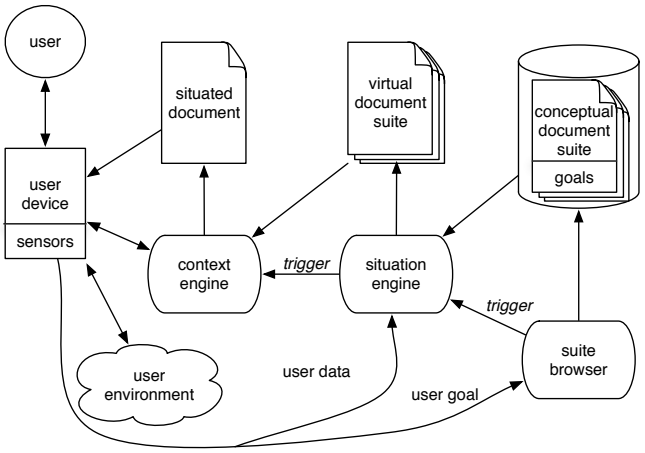

Fig. 1. A functional architecture for situated documents management

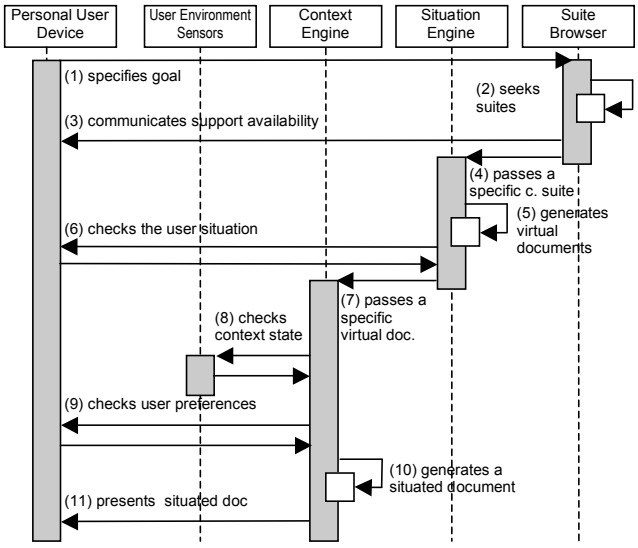

Fig. 2. A fragment of the communication protocol between the user and the system

context variables and of their changes. Additional sensors may be embedded in the environment and may send information to the user device, that acts also as a bridge for transmitting user and environment information to other system components.

At process start, the selection of a specific user goal triggers the *suite browser* component that queries the suite database in order to find a match between the collection of goals associated to the available data structures and the specific user goal. The starting conceptual document of the suite is also selected.

If a match is found, the *situation engine*, triggered by the browser, generates a *suite of virtual documents*, using as a template the conceptual documents identified by the suite browser. Although the generated structure is anchored to a specific goal and contains the data relevant to the user situation, the content and the presentation are still undefined. Data is extracted from a database, from the user device or interactively queried to the user, depending on the application and on the type of information required.

A single conceptual document may generate one or more virtual documents, each one corresponding to a different situation. The virtual documents still don't take into account the different dimensions of the context that may lead to different final concrete documents. Such transformation is operated by the *context engine*, on the basis of the information received by the sensors and loggers that monitor the user and the environment. The final result is a *situated document*, fully instantiated with application data and context information, which is presented to the user.

The requested user data are usually passed to the situation engine by the user and, in most situations, they are already stored in the user personal device and transmitted on request.

Overall, the situation and the context engine act as content and presentation filters. The engines, starting from the suite of conceptual documents and following the user in the evolution of its situation, lead to the definition of a set of situated documents whose content and presentation are the most appropriate in the specific situation constraints.

If the accomplishment of the user goal requires not only reading, but also building a document, the system guides the user to compose it with data and format suited to the situation.

The system, sensing the environment, might suggest the user to seek for available local facilities; for example, the system may be aware of the fact that the user is currently inside a public office and may suggest that the document might be available there as a printed form to fill in.

V. An example of situated document processing

The processing of a situated document is illustrated in Figure 2 by a UML sequence diagram. The labels at the top of the diagram describe the components involved in the communication process:

- the personal user device, used as a filter for communicating the user intentions to the system and for giving feedback;
- the user environment sensors, that automatically log the values of the context variables;
- the context and the situation engines, that filter and manipulate a specific conceptual document suite according to the user's goals, data and environment;
- the suite browser, that seeks the set of conceptual suites to find a match coherent with the user's intentions.

We comment the diagram with a hypothetical dialogue between the user and the system. For simplicity, the communication is humanized and expressed in the form of a natural language dialogue; it is worth to note that the way the dialogue itself is done and evolves is part of the situation. The numbers in parentheses refer to the communication protocol steps of Figure 2.

SYSTEM: Please insert your goal.

USER: To buy a house. [*The user sets a goal (1). The suite browser component seeks the conceptual document suites (2), finds a match and returns it to the user (3).*]

SYSTEM: Ok, I can support you. [*The situation engine receives the retrieved conceptual suite (4) and uses it as a template for generating a virtual document set (5).*]

SYSTEM: You have many choices: a) you want general information about the documents needed, b) you want to fill in a preliminary draft, c) you are ready to sign in the final agreement. Please, make your choice. [*The situation engine checks the specific user situation (6) to present the proper documents.*]

USER: To sign in a preliminary agreement.

SYSTEM: Ok, you can have the document in several forms: a) on the screen of your device; b) in a printed form, if you have a bluetooth enabled printer; c) as an e-mail message at a specific address. Which is your choice? [*The situation engine selects from the virtual document suite the first item complying with the user situation. The document is passed to the context engine (7), which checks the context state (8) and asks the user about the preferred format among the compatible ones (9).*]

USER: Screen output.

SYSTEM: Ok, please wait. [*The context engine generates a situated document compliant with the context and the user choice (10) and visualizes it on the user device (11).*]

USER: ...

The dialogue will proceed until the accomplishment of the user goal. The system will periodically check the state of the context variables (8) and, if any variation occurs, will generate a new set of situated and instantiated documents.

The user might also give up or suspend the current goal and make a request for a new one. In such case the system will be re-initialized and the dialogue will start again from step (1).

VI. Conclusion

In the area of context-aware systems, documents are designed and processed as collections of multimodal contents selected and/or transformed according to different properties of the user context. The chunks that compose a multimodal document are filtered from a "meta-document" base that may contain different content versions for different contexts, or transformation rules adapting content and presentation to a given context. The meta-documents are built on the rules of the application domain they address, and on the technical media suitable for their delivery.

In this paper we have presented a model for the meta-documents, that we have called conceptual documents, and for the instantiated documents, that we have called situated documents, based on a two stage processing of the user situation: a first stage considering the user goal and the relations between the user goal and the documents; a second stage considering the user local context.

The model we have discussed is a first attempt to formally define the many facets of the user situation, according to a situated cognition approach. The border between the variants induced in accessing or drawing a set of documents by the user goals and by the user local context is generally fuzzy. In the juridical domain, that we have used as a case study, the border is clear and sharp, due to the laws, regulations and rules that constraint the way documents are written and linked. The educational domain, that we have occasionally referenced, is less precise, yet habits, if not rules, exist in any educational institutions about the delivery of learning material.

The sphere of personal and social relations is very broad and variable, and a generalization of the model presented here could be hard. Nevertheless, the number of application domains in which the user situation can be described not (only) in emotional terms but as a goal dependent situation is large, and encompasses all the domains in which actions and information have some degree of "official" effect on the social relations.

Acknowledgment

We acknowledge Damiano Macedonio for helpful hints on the formalization of the document model. This work is partly supported by ACRES, Cultural Association for Economic and Scientific Research, Venice, Italy.

References

[1] A. Harter, A. Hopper, P. Steggles, A. Ward, and P. Webster., "The anatomy of a context-aware application," in *Proc. 5th Annual ACM/IEEE Int'l Conf. on Mobile Computing and Networking*, Seattle, Washington, August 1999, pp. 59–68.

[2] B. N. Schilit, N. Adams, and R. Want, "Context-aware computing applications," in *Proc. Workshop on Mobile Computing Systems and Applications*. IEEE, December 1994.

[3] A. Schmidt, M. Beigl, and H.-W. Gellersen, "There is more to context than location," *Computers and Graphics*, vol. 23, 1999.

[4] A. Dey, "Understanding and Using Context," *Personal Ubiquitous Computing*, vol. 5, no. 1, pp. 4–7, 2001.

[5] R. Steele, M. Lubonski, Y. Ventsov, and E. Lawrence, "Accessing SMIL-based Dynamically Adaptable Multimedia Presentations from Mobile Devices," in *International Conference on Information Technology: Coding and Computing (ITCC'04), Volume 2*, Las Vegas, Nevada, USA, 2004, pp. 410–415.

[6] L. Villard, "Authoring transformations by direct manipulation for adaptable multimedia presentations," in *Proc. 2001 ACM Symposium on Document engineering*. Atlanta, Georgia, USA: ACM Press, 2001, pp. 125–134.

[7] L. Villard, C. Roisin, and N. Layaïda, "A XML-based multimedia document processing model for content adaptation," in *Proc. of Digital Documents and Electronic Publishing (DDEP00)*, ser. LNCS, vol. 2023. Munich, Germany: Springer, September 2000.

[8] S. Boll, W. Klas, and J. Wandel, "A Cross-Media Adaptation Strategy for Multimedia Presentations," in *ACM Multimedia*, 1999, pp. 37–46. [Online]. Available: citeseer.ist.psu.edu/boll99cross.html

[9] O. G. A. Celentano, "Context-aware design of adaptable multimodal documents," *Multimedia Tools and Applications*, vol. 29, no. 1, pp. 7–28, 2006.

[10] N. Elouazizi and Y. Bachvarova, "On cognitive relevance in automatic multimodal systems," in *Proc. IEEE Conf. on Multimedia Software Engineering*, Miami, FL, 2004, pp. 418–426.

[11] C. Lueg and R. Pfeifer, "Cognition, situatedness, and situated design," in *Proc. of Second International Conference on Cognitive Technology*, Aizu-Wakamatsu City, Japan, 1997, pp. 124–135.

[12] N. M. Seel, "Epistemology, situated cognition, and mental models: 'like a bridge over troubled water'," *Instructional Science*, vol. 29, pp. 403–427, 2001.

[13] B. A. Nardi, *Context and Consciousness: Activity Theory and Human-computer Interaction*. Cambridge, MA: The MIT Press, 1996.

[14] L. Suchman, *Plans and Situated Actions*. New York: Cambridge University Press, 1987.

[15] J. Bardram, "Plans as situated action: An activity theory approach to workflow systems," in *ECSCW*, 1997.