I. Wagner, H. Tellioglu, E. Balka, C. Simone, and L. Ciolfi (eds.).
ECSCW'09: Proceedings of the 11th European Conference on Computer Supported Cooperative Work,
7-11 September 2009, Vienna, Austria

PRODOC: an Electronic Patient Record to Foster Process-Oriented Practices

Federico Cabitza, Carla Simone and Giovanni Zorzato
Università degli Studi di Milano-Bicocca, 20126 Viale Sarca 336, Milano, Italy
cabitza,simone,zorzato@disco.unimib.it

Abstract. The paper presents PRODOC, an Electronic Document System that allows users to navigate documental artifacts according to predefined process maps. In fact in PRODOC, process models are to be considered as maps that users willingly take as guide for their decisions and actions, rather than scripts prescribed from above. The main tenet of this research is that, by integrating documents and processes, documental practices and related work practices could better align to intended models of action. The underlying concept is the result of a long empirical research in the healthcare domain, where we have deployed PRODOC as an innovative and process-oriented Electronic Patient Record. The user participation in the phase of document definition and clinical processes modeling is central in our approach and it is illustrated in three scenarios of the software informal validation that we present in this paper.

1 Health records: a challenging domain

The healthcare domain is the marketing target of many vendors that propose systems of various kinds to support different phases or activities of the healthcare process. Some of these systems receive a good acceptance since they mainly deal with administrative aspects, e.g., reimbursement accounts, or with quite streamlined and standardized activities, e.g., laboratory examinations. Other systems encounter a stronger opposition since they involve aspects of the clinical care that deal with subtle nuances of clinical work: among these systems there is still the Electronic Patient Record (EPR). EPR introduction in the hospitals is still rare and in any case problematic (Heeks et al., 1999; Hartswood et al., 2003; Berg and Winthereik, 2003;

Koppel et al., 2005). This was also manifest during our interactions with several practitioners working in hospitals that were at different stages of introduction of EPR systems. We noticed a diffuse sense of frustration in these clinicians towards this process, which in many cases was almost out of their control. In fact, decisions about the digitization of patient records (PRs) were often governed either by external forces, like the bargaining power of influent vendors, or by internal strategies, which were aimed at a uniform adoption of the same system across different departments to improve cost control and resource management.

Obviously, for us as external researchers, these processes were out of our influence. We had nice conversations with the practitioners about a matter that they must have had at heart, and we met their open availability to validate some of our ideas on small prototypes and mock-ups in sort of "loud thinking" users sessions. This gave us the motivation to capitalize the wealth of experience gained with practitioners and try to address the point whether a really *innovative* EPR could be possible. We agreed with practitioners that innovativeness was all about the challenge of fulfilling their primary needs (i.e., care and its account) without requiring them to distort their usual practices and saddle themselves with the low-level integration with the hospital information system. We decided to face this challenge since we believed that our outcome would at least provide practitioners with a tool that they could exploit in their interaction with the hospital management and the ICT vendors to spur them towards real innovativeness. This paper is a first step in this direction: after discussing critical aspects of existing EPRs, it gives the basic tenets and describes the current version of our outcome, a prototype called PRODOC, together with an account of its ongoing validation.

2 Current EPRs: a critical view

Our empirical research about the introduction of EPRs in four hospitals in Northern Italy highlighted three main critical areas that deeply influenced the design of PRODOC:

First: Standard EPRs provide their users with a sequence of electronic forms that reflect how information has been modeled to deploy the underlying database: i.e., in terms of domain entities and corresponding relations. These aggregate views propose layouts of "assembled squares" where practitioners can read or write clinical data according to predefined schema. Moreover, these forms are linked together according to a business logic that, in general, has nothing to do with the paper forms that are in use before the EPR's introduction and with the practices that doctors have built around them. Several studies (e.g., (Harper et al., 1997; Fitzpatrick, 2000)) pointed to the advantages of the paper-based forms over the electronic ones but the focus was more on the affordance of the paper medium rather than to the internal structure of these forms. Indeed, we believe that how doctors organize information in their records (a task that is facilitated in paper-based forms) is crucial: in fact, we observed that the structure of a document/record/chart is usually the outcome of a long-lasting process where the results of consolidated work practices and conven-

tions have been stratified. In particular, this structure is able to let tacit knowledge be evoked by the mutual position of information or by specific graphic cues or textual annotations that are easily juxtaposed beside or around it. All this rich combination of tacit, implicit and explicit knowledge is mainly lost when EPR is digitalized: the consequence is that practitioners are requested to abandon the practices on which they usually base the effectiveness of the care process, and to behave according to something that is out of their experience: i.e., the business workflow logic of the EPR.

Second: The business logic of EPRs are usually invariant with respect to the specific care processes that doctors are able to tailor to specific diseases and to how patients react to their interventions. These processes can be implicit medical knowledge, or made explicit in what is usually called a Clinical Pathway (CP) (Sloan and Guinane, 1999); the whole body of these processes is a sort of procedural knowledge that in any case clinicians retain and exploit to articulate their actions and those of the practitioners involved in the same clinical case. Generally, in current EPRs there is very little or no support towards these disease-related and patient-centered processes: and no wonder there is not. In fact clinical processes are defined, updated and dismissed not only according to medical evidences and guidelines but also according to very local drivers, like available resources, staff, level of education, available equipment, and even hospital topology (Lenz et al., 2007). Therefore, in the more positive case, the burden to "remember and follow" the intended pathway is left on the shoulders of practitioners; in the more negative case, they have the additional burden to cope with a contrasting logic embedded in the EPR.

Third: In specialist literature, doctors themselves propose the CP as an effective tool to decrease undesired practice variability and improve clinical performance (Campbell et al., 1998). We also, in Cabitza et al. (2008), observed the practice of adding the sheets of a reference CP into the record folder of a patient with a specific disease. In those settings, and especially when the CP is the outcome of a bottom-up collaborative effort , the integration of CPs – however represented – and the EPR is advocated to improve appropriateness and to aptly respond the increasing demands for patient safety, better risk management and reduced costs. In fact, studies have shown that IT applications can increase pathway compliance, if they embed pathways in routine work, and more precisely in routine documentation (Lenz et al., 2007). To this aim, a traditional approach is that of conceiving a set of electronic checklists that allow doctors to check the compliance of their practice against the pathway (Blaser et al., 2007). Yet this approach, even if mitigated by the principle of "charting by exception" proposed by Short (Short, 1997), usually results in documental overhead because clinicians end up by reporting more *about* the pathway than *according to* the pathway. As discussed by Berg (1999), clinical reporting follows clinical work closely and clinical work is influenced by how and when clinicians report it since they rely on the record to coordinate with each other. Even when EPRs acknowledge this mutual influence, they embed processes that are interpreted as "yet other" workflow models and scripts by "engines" that govern what tasks the application can allow users to perform, or as the outcomes of domain

knowledge representations by inference engines oriented to planning and decision making (Smart and Roux; Quaglini et al.; Aigner and Miksch, 2006).

3 Innovative EPRs: the basic tenets

To overcome the above limits, we identified basic tenets that we adopted for our proposal and express in what follows as they were formulated in frank terms by practitioners:

"Let me keep my folders!": from their introduction at the end of the 19th century, paper based PRs have evolved in very well organized bundles of documents, charts and records. What to software engineers could seem confused folders are actually sets of sheets that are grouped together according to their contents, to the time span they refer to and to the phase of the care process they are associated to (Cabitza et al., 2005). Doctors see this whole information not as a static "folder" but rather as a "web of artifacts" (Bardram and Bossen, 2005) whose ad-hoc and often unpredictable organization allows for different levels of aggregation and retrieval; this flexible folder allows doctors to continuously re-arrange its sheets so that, e.g., the peculiar proximity of pieces of information can facilitate peripheral readings that profitably complement data that are on current focus; or it can allow for a comprehensive view of data according to different criteria, like time intervals or basic indicators of patient's condition (Fitzpatrick, 2004). In the EPR design, this means that the "folder" metaphor by which to gather and present clinical data must be preserved against the omnipresent metaphor of the "dashboard", borrowed by other information-intensive domains.

"Let me do what I do on paper": our observational study confirmed that practitioners appreciate the possibility to keep browsing and skimming clinical data as they were used to with paper-based bundles as well as to add extemporary and informal annotations (Hardstone et al., 2004; Bringay et al., 2006) to the basic structure of documents: more specifically, practitioners claimed that such flexible annotations are a natural means to promote awareness and to evoke tacit knowledge pertinent to the annotated information. Practitioners also appreciated the possibility to customize, tailor and design their own forms according to their local needs as they were used to with paper-based forms, which they usually could compose and print locally. These continuous improvements always require a formal validation by the hospital management but they are on ordinary agenda if doctors comply with the core data set that has been defined at organizational level. Conversely, modifying an EPR requires much more than mere negotiation with the management since it always requires modifying the EPR DBMS and often even its internal business logic.

"Integrate data and processes but don't mix'em up": usually documents and processes are seen as independent units or, better yet, able to characterize a work domain at different levels. However, clinicians told us that when a representation of an organizational process is concerned with the clinical dimension of hospital work, they should be seen as simple maps, rather than "scripts" (Schmidt, 1997) of an application logic that prescribes and steers clinical behaviors. The term map

suggests that these representations can be used to provide a sort of loose "topology" that can promote awareness of the unfolding of the illness trajectory, of what activity is currently being performed and of what doctors should do next to coordinate with each other in a seamless way. The process can then be seen as a alternative way to organize data in terms of *when* they have to be produced and to *what* aim. Therefore, to improve awareness and coordination, clinicians suggested that there is no need to build a comprehensive model of the care process, whereas it suffices to make the crucial input/output relations between activities and specific documents (or parts of them) clearly explicit (Cabitza et al., 2007). Moreover, in the healthcare domain, process maintainability is a pressing requirement, as we reported in Cabitza et al. (2008) since strict standardization from above built into electronic systems is bound to fail (Berg, 1997); indeed, for clinical pathway to be effective aids in guiding practice, doctors must be able to continuously update and maintain them, according to both local practices, new scientific evidences and agreed guidelines based on consensus within a particular discipline.

4 The basic design choices

In order to build a prototype fulfilling the above basic tenets, we had to make choices and tradeoffs about how to represent documents and processes. To this aim, we made a survey of the main solutions reported in the literature and collected examples from the hospitals we have been in contact with in the last years. In regard to how represent clinical processes, the most used formalisms are extensions and customizations of the flowchart notation. In a minority of cases, the representation is based on matrixes, or time-grids, describing activities and responsibilities. More often, protocol-based care is formalized in terms of hierarchical networks of component tasks, that unfold over time, like Asbru (Shahar et al., 1998), GLIF (Boxwala et al., 2004) and PROforma (Sutton and Fox, 2003) (just to cite the most used). An alternative approach takes a declarative point of view, e.g., the declarative language CIGDec (Mulyar et al., 2008), to increase the flexibility of process execution: in fact, activities can be modeled without causal relations and possible precedences between them can be expressed in terms of additional constraints. In this rich panorama, our choice was to combine the advantages of an explicit representation of the relationships among activities with the flexibility resulting from releasing this representation from the engine that is usually associated to procedural languages. This suggested us to adopt the Business Process Modeling Notation[1] (BPMN), a standard notation close to flowcharts, since we observed it is familiar to most of the healthcare practitioners for historical reasons

In regard to how represent document structures, we adopted an approach that is common to other applications (see, e.g., Morrison and Blackwell (2009)) that

[1] The Business Process Modeling Notation is a graphical representation developed by the Business Process Management Initiative (BPMI) and currently maintained by the Object Management Group.

try to mimic the look-and-feel of paper-based chart quite closely. More than this, our solution aims to provide an interaction mode that makes the definition of document templates as natural as possible for the practitioners. This solution invites practitioners to see documents as just topological arrangements of data fields or "document constituents" that we call *didgets* (more details will be given in the Section 6.1). Our point is that avoiding any explicit representation of the relations between these constituents allows for a great level of flexibility in presence of changes in the forms' layout as well as of the insertion/deletion of new pieces of information. This choice has had a strong impact on how to realize the clinical data repository behind our EPR: in fact, its structure is dynamically derived from the pieces of information contained in the document templates by adopting a multidimensional approach (Pedersen and Jensen, 2001).

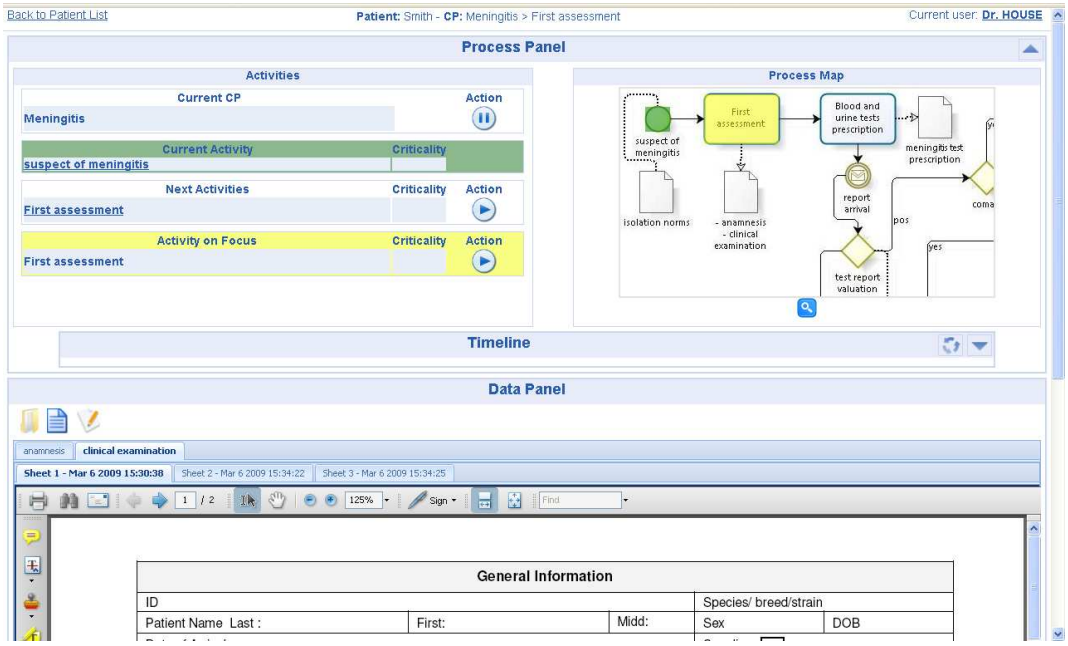

Figure 1. The main page of PRODOC (the Timeline is minimized).

5 PRODOC: description of the user interface

In this section, we outline the main functionalities of PRODOC (PROcess-oriented DOCumentation) a web-based prototype of EPR aimed at supporting clinicians in integrating clinical pathways into their records. First, we describe the main page that PRODOC displays once a doctor has selected an activity from the list of current CPs that are active on a particular admitted patient. This page is divided in two main sections (see Figure 1): the upper pane is called **Process Panel** and provides process overview and navigation functionalities. The lower pane is called **Data Panel** and provides user with access to data through paper-looking documents (currently, they are PDF forms). In its current version, the Data Panel provides also annotation

functionalities through the rich command palette provided by the Acrobat platform. The Process Panel can be reduced (or minimized) so as to give users a full-screen of the Data Panel: this can be particularly useful if PRODOC is used on a tablet or ePaper PC; in fact, this full screen view allows to simulate the traditional interaction with paper-based artifacts in all those settings where this is considered a plus by practitioners. Above the two main panels, PRODOC encompasses also a small textual section, which reports the main patient personal data, the navigation trail and information on the current user.

5.1 The Process Panel

The Process Panel allows users to have a quick glance of the process map, to assess and set the current state of the clinical process, and consult the process history. To this aim, the panel is divided into three sections (see Figure 2). Two of these sections, the **Process Map** and the **Activities** are fixed, while the third, the **Timeline**, is collapsable as the whole Process Panel.

The **Process Map** is a window where a portion of the graphical BPMN-based representation of the current CP is displayed. The map is automatically centered around either the active activity or the activity currently on focus (i.e., the activity that the user has selected to view the associated documentation) but it is also draggable, so that users can examine different regions of the process schema. The process map is an active map: this means that the diagram elements depicted therein are active links that make an activity on focus and its associated documentation be displayed in the Data Panel. PRODOC also provides a *zoom function* (the magnifying glass icon depicted in Figure 2) that allows to enlarge the map and see it in full. In the current prototype, the active activity is highlighted in green, while the activity on focus (when this does not coincide with the former) is colored in yellow. The Process Map works in combination with the Activities section.

The **Activities section** reports textual information about i) what the current activity in the process is (and its criticality); ii) what activity/ies follow/s the current one (and their criticality); iii) what activity is currently on focus: its documents are currently displayed by the Data Panel. In regard to the current activity, users can suspend it, by pressing the `Pause` button. Since the current prototype does not handle parallelism within the same CP, two activities of the same CP can not be active at the same time. Therefore, if an activity is temporarily suspended, also the overall CP is suspended and an event of temporary exit from the CP is generated. When the CP is suspended, it enters a sort of *unspecified activity* where users can get access to (and use) all the patient documentation, The suspension lasts until either a new activity is started or the current activity is resumed. When a user means to terminate an entire process and exit the CP, she can select and make one of the *end event* elements (depicted in the process map) active.

User can select any activity to put it on focus in order to read or write the associated documents. Once an activity has been put on focus (through either the Process Map or the Next Activity link) users can activate it by pressing the `Play` button, to

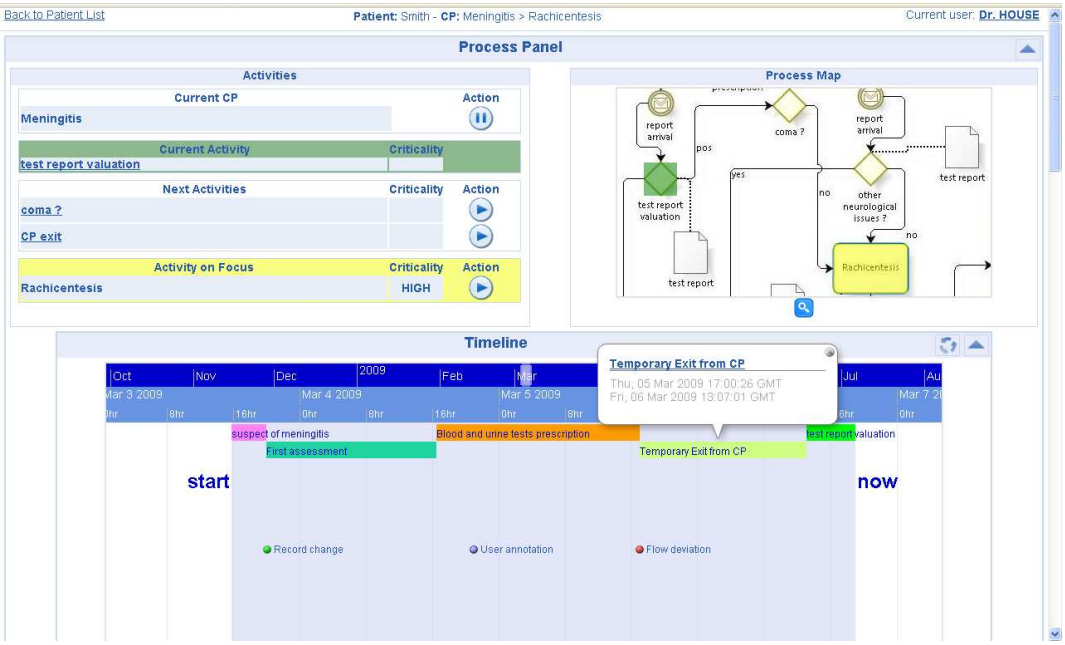

Figure 2. Process Panel with the Timeline open.

tell the system this is the activity they are currently involved in. In so doing, the system automatically terminates the current activity, makes the activity on focus active and updates the next activity links accordingly. If the activity that the user activates does not directly follow the current process activity (i.e., it is not one of the next activities), the system generates a *deviation* event that is recorded to be displayed in the TimeLine.

PRODOC does not pose significant constraints to user action and optionally can ask users to provide a written justification if they start a new activity without following the order indicated in the clinical pathway. This request can be set as either mandatory or optional by the responsible of the CP (i.e., usually who led the task force supposed to design and maintain the CP from the available guidelines) according to local conventions and if this indication is deemed useful to provide doctors of next shifts with the context to understand past decisions; in our specific case, the CP responsible proposed this functionality as a way to foster feedback from doctors on the extent the clinical pathway at hand is useful and reliable in their daily practice.

The **Timeline** is a section at the bottom of the Process Panel that can be expanded and collapsed at need every time doctors need to get a visual representation of when relevant events occured and during what activity. The timeline displays both the process history, i.e., the sequence of CP activities that doctors have actually performed till the present moment, and any relevant event that has occurred during the patient stay that far. When users open the Timeline, it is centered on the present time if the process is still in progress, or on the end of the healthcare episode, if the patient has been already discharged. When the Timeline is updated, PRODOC displays unobtrusive pop-up messages, so that users can determine whether to refresh the timeline by pushing a specific `Refresh` icon. In the Timeline, activities are

depicted as a succession of colored bars, while events as dots of different shape and color according to their predefined type (e.g., changes in the record, report arrivals, *deviations* in the CP trajectory). Users can scroll the timeline along the horizontal axis by two graduated scales to explore the process history with different time granularity: the former scale is divided in monthly intervals and allows for a quick shift upon the time axis; the latter one splits the timeline according to days and hours and allows for more accurate movements. When a user clicks on an activity bar, PRODOC shows a pop-up balloon that reports the start time of the activity, the justification given for its activation (if any), its end time (if already performed) and a direct link to the associated documentation and contextual content. This latter functionality means that doctors have got a sort of "time machine" by which to see the record's content at the selected time: pages accessed through the timeline panel are displayed in the Data Panel as usual but are read-only unless the activity on focus is the current one. In regard to the relevant events displayed in the timeline, the current implementation of PRODOC considers three event types:

- data events (green spots), which are pinpointed into the timeline whenever users insert new data in the Data Panel (i.e., save the content at sheet level);
- annotation events (blue spots), which inform users of when their colleagues have annotated a document.
- deviation events (red spots), which are displayed either when users start an activity that does not follow the current one in the CP map; when users write on a document that is not associated with the current activity; or when users suspend a CP.

If a user selects either a data or an annotation event, PRODOC displays a balloon where exact time of document saving and the author identity are indicated, as well as a direct link to the saved/annotated document.

5.2 The Data Panel

Below the Process Panel, users can see the Data Panel (see Figure 3) in which PRODOC embeds the set of the only documents associated with the activity currently on focus. If, conversely, the user has to consult the whole documentation during a specific activity, she can get access to all the sheets regarding a single patient, by pressing the `Display All` icon (the first from the left in the command bar depicted in Figure 3); PRODOC displays the whole documentation for a specific patient also after the user has temporarily suspended the CP.

In the Data Panel, users can read and write the forms that are progressively compiled during the patient stay. Usually each activity has some document templates and sheets associated; users can swap from sheet to sheet pertaining to different templates by means of tabs, so as to mimic how they are used to in the case of paper-based folders. When users complete a sheet, they can have PRODOC generate a new sheet for each template by selecting the `New Sheet` icon (the second from the left in Figure 3). By affinity with the typical constraints of the healthcare domain where each inscription is a legal report, inserted data cannot be changed nor

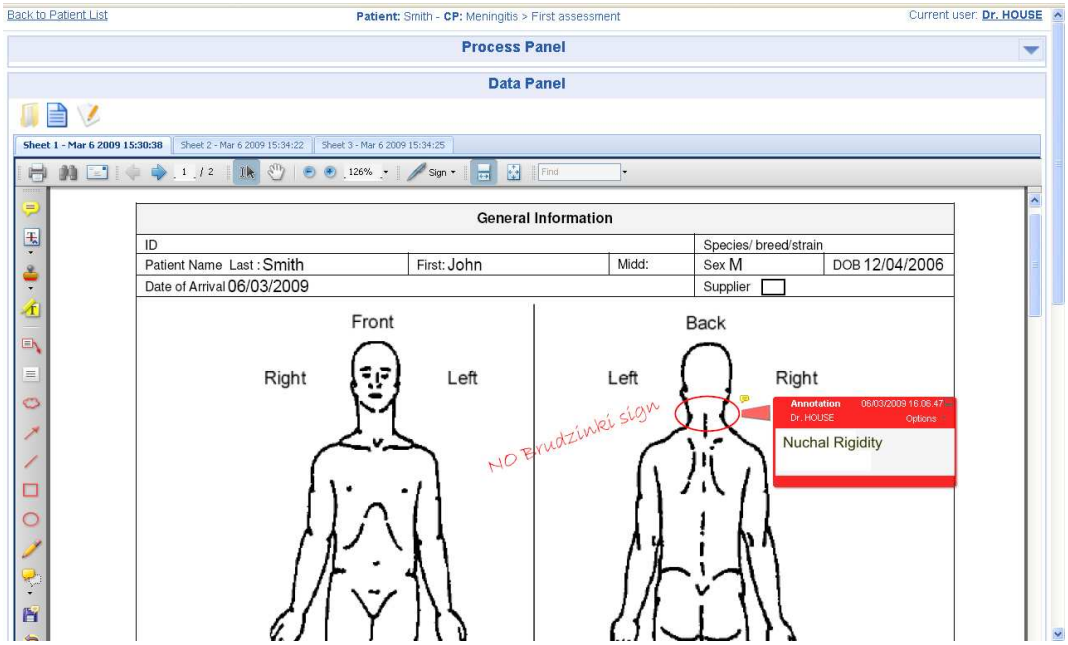

Figure 3. The Data Panel displaying an annotated form.

erased. Inscriptions can be stricken-through and new data can be juxtaposed beside the obsolete (or just wrong) ones as they would on paper[2].

Sheets are opened in read-only mode. The user that has accessed a sheet first can also turn on the edit mode (third icon from the left) and get a lock on the document for a particular patient. In this way, her colleagues can still consult the sheet in read-only mode but cannot edit it till she has released the lock by either saving the last changes or discarding them. The Data Panel provides user also with versatile annotation capabilities. Since, in the current version of the PRODOC prototype, templates are PDF forms (primarily for their strong resemblance with paper-based forms and the familiarity that generic users have with this format), each sheet is annotatable with the rich palette of drawing markups provided by the Adobe Acrobat platform: users can then add textual comments, either by keyboard or by the free-hand pencil tool; they can draw oval, circles and rectangles around portions of text or pictures and mark arrows, in a very similar way as they use to do on the paper-based record with pencil inscriptions, sketches or with post-it notes, in order to communicate with their colleagues, drop a note to recall in the next shifts (Bringay et al., 2006) and unobtrusively promote collaboration awareness and coordination. Each annotation is registered as a separate event, characterized in terms of time and author, and it is consequently reported in the Timeline; even more noteworthy, as soon as a user adds a note into a sheet, this event is immediately notified to all the users that have the same sheet open, so as to facilitate synchronous communication and make actors aware of any change in the documents that they are reading.

The main point to retain here is that PRODOC can be used in three different

[2] This constraint can be easily relaxed if PRODOC is to be used in different documental domains.

operating modes: either as a sheer electronic record where any artifact is accessible with no order constraint, if users minimize and disregard the process panel; as an interactive process map, if users have to use another third-party or legacy EPR and still want to keep trace of the current activity in the context of the adopted CP; or as an integrated tool that enables process-oriented document navigation and event-based information retrieval. In the next section, we illustrate the main functionalities of PRODOC in the context of three scenarios, by which we undertook the informal validation of the current prototype.

6 Participatory discussion through use scenarios

From the beginning, we realized that an effective evaluation of PRODOC could not be conducted in a laboratory setting. In fact, PRODOC has been designed to support the ongoing recording of (clinical) acts and the articulation of activities unfolding around this general task. Therefore, a true validation of PRODOC would require a long-term deployment and an analysis of the impact of the system on both clinical and documental practices. On the other hand, to gain an initial feedback on the effectiveness of the main functionalities of PRODOC, we undertook an informal validation according to a qualitative approach that encompasses the involvement of a small user panel. To this aim, we have so far conducted informal evaluations with clinicians in the context of three simplified use cases: one case regarding the definition of the patient record templates (to be displayed in the Data Panel); another case regarding the definition of clinical pathways (to display in the Process Panel); and one case regarding their combined use in a realistic clinical scenario.

These use scenarios were reviewed and personally experienced by selected practitioners during individual user sessions, lasting approximately forty-five minutes: a specially instrumented version of the PRODOC prototype was deployed in the hospital library so as to monitor the browsing activity, command selection and software events triggered by users during application execution. For our user panel, we invited the head doctor, a senior doctor that the former invited for his past experience with EPRs, the head nurse and one of her most experiences colleagues to use the system following the three scenarios as a sort of loose plot. The evaluation methodology we used was based primarily on usage logs and user feedback gathered in approximately one-hour long semi-structured interviews taken immediately after the user sessions. These interviews were used to support our understanding of the usage logs, to acquire feedback about how the tool was perceived and keep track of relevant suggestions. This first round of evaluation sessions, although informal, allowed us to gain insights into how PRODOC would be understood and used by clinicians to get access to their daily documentation and comply with the specifications of clinical pathways. In what follows, we will run through the scenarios we proposed to the user panel and will interpose the main remarks of the involved users while describing in some further detail the main functionalities of PRODOC's current implementation.

6.1 Document schema definition

Before PRODOC can be used, its users have to create both the process schema and the document templates it will use. This phase of preliminary definition is an important part of the innovative approach of PRODOC to process-oriented documentation. In fact, on the one hand, we wanted users be as much independent as possible in creating their own processes (i.e., clinical pathways) and correlating them to their own records. On the other hand, PRODOC also proposes a strong document-centered approach to data, i.e., we wanted users come back to thinking of data as elements of specific and "tangible" documents, rather than aggregated elements taken from underlying databases and gathered together in sort of virtual views.

To this aim, we invited our user panel to use an editor by which to re-build their own document templates and make them as similar as possible to the paper-based charts they were currently using. This was something that two practitioners we involved were already used to: in fact, they were members of the large group of hospital representatives that was supposed to compose (and maintain) the templates of the hospital patient record, have these validated by the hospital management, convert the validated templates into PDF files and then share them to their ward colleagues, so that these could print the blank charts at need to progressively feed their paper-based folders. The template editor we provided to the user panel was an augmented version of a very popular word processor, already used at the hospital: we developed a Visual Basic application that, while the word processor is open on a new document, displays a sort of small floating panel from which users can select the proper *didgets* to insert in the document. We called didget a *documental widget*, that is, a coherent set of data fields that is convenient to gather together at a certain level of description. Following the scenario, the user panel was invited to drag a *patient didget* from the editor palette and drop it on top of a chart template in order to add the usual patient data (e.g., name, surname, date of birth) in the chart header; users could also choose a *prescription didget* (which encompasses fields for the drug name, dose, the scheduled administration time, etc.) to create a new row for the table of the drug prescription form. We explained that didgets can be any element of a typical form: either simple input fields, or set of fields, multi-line text areas, check boxes, drop-down lists or combo boxes, according to the minimum data set that key users, domain experts and system analysts had collaboratively identified from the domain and document analysis. The user panel saw that the same didgets could appear in more templates and, obviously, in more instances of a single template, i.e., in more sheets. In this latter case, a didget could be *local*, if the data associated with it are not to be replicated in other sheets; or *global*, if the data must be replicated in each occurrence of the same didget all over the record.

Seeing document templates as containers of didgets, and the task of document editing as that of positioning didgets in a blank page was not a practice that fitted the habitual practices of our panel easily and immediately. Yet, at the end, the concept of didget was received quite warmly: users understood its scalability from a single text input field, e.g., a body temperature field, to a complex record, e.g., the matrix of a fluid balance charts, that is a "field macro", as it was suggestively

called by the head doctor, which could be reused in every record and chart where those data need to be reported and consulted. On the other hand, users realized that the document structure, i.e., how data are displayed in a record, and data fields could be *decoupled*. The senior doctor told us that he worked for a couple of years in a hospital ward where doctors had been using an EPR for years: he told us that after only few days from the first deployment he and some colleagues of his noticed that the body weight field was in the wrong page of the software application, i.e., associated to a preliminary phase of patient admittance where clinical data were not collected; and that the same field was not replicated in another page where it could have been useful to calcolate drug dosages. He told us that they asked the software vendor to change the user interface accordingly and that when he moved to the NICU, a couple of years later, they were still waiting for this patch. He told us this anecdote since he realized that in PRODOC he could have used a regular word processor and changed the user interface just by moving (or importing) a didget in a regular document. If this didget had not been anticipated by analysts, it could just have been added to the underlying DBMS as just another dimension related to the patient. In more general terms, the capability of changing the user interface of PRODOC (i.e., what is embedded in the Data Panel) at need and according to very local, specific and ad-hoc needs by just creating new document templates was seen as a clear plus of our approach.

6.2 Process schema definition

Once users had created the templates of their record, we told them that it was time to create their first pathway and correlate its activities with (not necessarily exclusive) sets of templates. In order to create a computational specification of a clinical pathway, we provided users with a graphical process editor that we chose because it is freeware and has got a very user-friendly graphical interface: BizAgi Process Modeler by BizAgi. This editor allows to create process diagrams in the BPMN standard format by simply dragging and dropping iconic elements: boxes for activities, diamonds for decisional branching points, circle for events and oriented arcs for flow relations. We invited the users to characterize the pathway elements by specifying their criticality (i.e., either importance or prescriptiveness) and that with this editor users could specify in terms of an extended property of either activities or flows. To associate activities with the set of templates that could be considered as either input or output of the related tasks, we instructed users to employ the BPMN constructs “association” and “artifact”. Due to the still semi-automatic integration between BizAgi and PRODOC, users had to write the name (URI) of the templates related to each task as a property of the artifact construct. For the next versions of the tool, they advocated a visual mechanism of template importing similar to the didget floating panel. While our panel showed it was proficient at modeling a pathway in terms of activities and branching points, it required some time in getting a clear comprehension of when and how to use the BPMN construct “event”. This is an additional feature with respect to traditional flow-charts, but users acknowledged

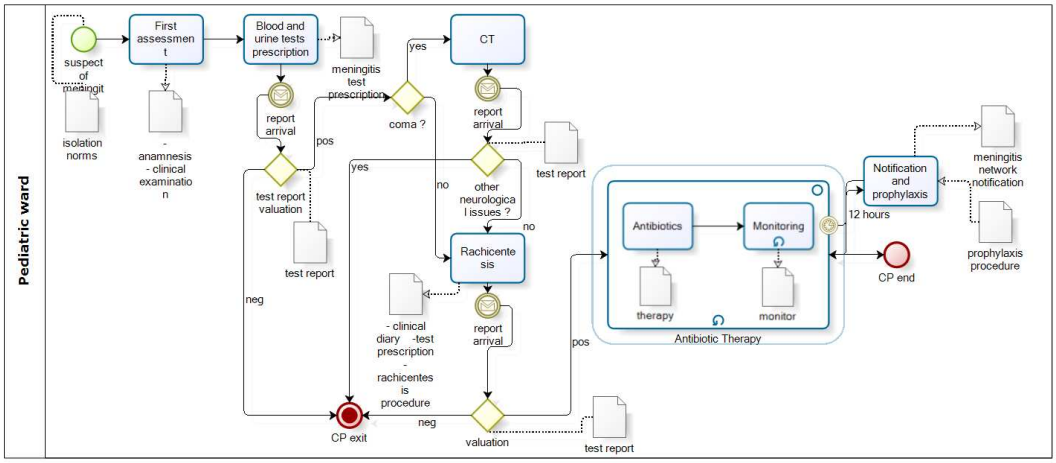

Figure 4. The clinical pathway doctors designed for meningitis cases.

its aptness to report either time- or data-related aspects of CP (e.g., “when the report becomes available”, “after twelve hours”).

In the scenario, after that users have diagrammed a pathway and indicated the relationship between tasks and documents, the editor then automatically exports the process specifications to ProDoc, in terms of both process map (as a raster image) and computational representation (as an XPDL[3] file). These resources are then used by ProDoc to enable the document navigation according to the process model.

6.3 Using ProDoc in a clinical scenario

After the two scenarios of process and template definition, we asked clinicians to consider ProDoc as their official patient record and fancy themselves involved in the management of a patient with a suspicion of meningitis. The corresponding CP, drawn by means of the editor mentioned in Section 6.2, is shown in Figure 4.

The user panel had to imagine a situation in which an infant, JD, was admitted in their pediatric ward with a meningitis suspicion formulated by the Emergency Department (ED) on the basis of clinical signs. The responsible physician of the ED had opened ProDoc and associated a *Hospital Stay* meta-process to JD so as to fill in the first pages of the corresponding patient record. A meta-process is just like a regular process but it can also contain sub-processes: i.e., activities that can be further characterized in terms of other processes. In this case, the Hospital Stay is a hospital process that encompasses very general activities in strict sequential order: the Patient Admission, the Patient Treatment and the Patient Discharge; the Patient Treatment task, in turn, can be associated with several other sub-processes, i.e., instances of different Clinical Pathways (CPs) by which doctors can manage the health problem identified in the Admission phase till discharge.

[3] XML Process Definition Language, standard developed by the Workflow Management Coalition (WfMC) to interchange process definitions between different management tools.

In the Hospital Stay process, the first activity is an Admission, and this is related to the document templates where the hospital Triage records the admission, identifies the patient and reports a first set of possible diagnoses. According to the Triage evaluation, JD is transferred to the pediatric ward. The admitting pediatrician opens PRODOC, selects the JD record, puts the Admission activity on focus and then consults the associated Triage documentation; after that, she starts the Treatment activity. When a subprocess task is selected (i.e., in this case, Treatment), PRODOC prompts for the selection of a subprocess that specifies the general task: in this case, the pediatrician decides to associate JD with the specific CP that the hospital published to cope with admitted cases of suspected meningitis.

In regard to this first part of the scenario, the user panel appreciated that a patient could be associated with several processes (and hence CPs) in parallel: although they acknowledged that they associate more CP to the same patient quite seldom, this would regard a number of complex cases when a patient is transferred from a facility to another for either complications or further investigations while she is still under the partial responsibility of the former facility: an equivocal situation that is usually difficult to manage when it is not clear who can do what. Practitioners also appreciated that a general pathway could include more specific processes: indeed, they claimed that a number of their pathways were quite "abstract" and with activities that often should need to be further characterized in terms of standard procedures, detailed diet regimens or more refined treatment protocols. Yet, at this point, the first of the main shortcomings of the current PRODOC emerged. In fact, the user panel was agreed that even if a patient is managed according to multiple pathways, a number of sheets from her patient record must be *shared* across these pathways, since they could be read and written in activities of any CP: the possibility to use the `Display All` icon (mentioned in Section 5.2) to see all the sheets related (through their father templates) to the running metaprocess (and to all its child processes) was considered a too complicated "trick".

Next in the scenario, the pediatrician consults the graphical map of the meningitis CP in the main page and that reminds her of what to do first when coping with such a case. The first activities mentioned in the CP are the *First Assessment*, which regards reporting the anamnesis and the findings of the clinical examination, and the *Test Prescription*.

Three users proceeded in this order and activated First Assessment by clicking the `Play` button. The senior doctor said that often, according to subjective impressions, diagnostic tests should come first in order to gather sound evidences for the correctness of the Triage diagnosis. In this case, users observed that the pediatrician could put Test Prescription on focus and then activate it. The point highlighted by the user panel was that PRODOC does not impose a strict order or restriction on when to use a particular document, such as it could happen in a workflow management system; rather, it only suggests users what set of documents they could need while they are performing a specific task, it gives them the possibility to create new sheets, or to open those already existent to accomplish a specific process activity. When the head nurse opened the Test Prescription form, she noticed that the pa-

tient's details in the header were already filled in: in fact, those data were coming from the global didget "Patient's Details" of the Admission form that was compiled in the first activity. Every new sheet from a template that has got that didget will have those data replicated with no additional effort from the practitioners.

When the pediatrician selects the activity Blood and Urine Tests Prescription without passing through the First Assessment, the timeline records the event as a deviation from the CP. The Test Prescription activity is associated with a Meningitis Test Prescription document template. The pediatrician marks the tests that she needs for the meningitis diagnosis (e.g., blood count, PCR and glycemia) and then saves the sheet. When the sheet is saved, PRODOC stores the data in the underlying database and records corresponding data events in the process history (timeline). Afterwards a nurse consults the Meningitis Test Prescription sheet, takes the blood samples and marks the checkbox "specimen taken" in the Prescription sheet.

After we had illustrated the part of the scenario involving the nurse, the head physician and head nurse noted that it would be important to explicitly distinguish what different roles are involved in a particular task. This was the second main shortcoming of PRODOC reported in the user sessions. In fact, role management is still preliminary in PRODOC, which currently only distinguishes between the "responsible" of the CP and simple performers. While the latter ones can change any document related to the CP, only the owner can create new process instances, terminate and activate activities or suspend a CP, i.e., change the CP status. As a matter of fact, roles can be easily represented in the process map by exploiting the BPMN elements "pool" and "lane", used to represent different participants in a process and to organize activities within pools, respectively. Users said that this knowledge about "who can do what" had to be reflected in PRODOC, in terms of transparent management of different read/write access permissions to documents according to the role associated to the user currently logged in[4].

Coming back to the scenario, while doctors are waiting for the lab to analyze the specimens and issue the report, a pediatrician can decide to prescribe an anti-inflammatory drug. This task is *not* represented in the CP schema because it represents an exceptional decision that really depends on the particular conditions of the patient. Obviously, a CP schema cannot include all the possible exceptions that may occur during a treatment (even if it should consider the most important ones and those that usually lead to aborting it). As a consequence, PRODOC allows for writing additional documentation that is not related to any activity of the CP. In the scenario, the pediatrician pauses the CP (pressing the `Pause` button beside the current activity) and creates a new Therapy Prescription sheet to order the drug. This Therapy Prescription sheet will be listed under all the activities that are associated to the same document template. In the meanwhile, the laboratory has sent the test report and this makes a doctor activate the Test Report Evaluation task. This task is related to the Laboratory Test Report document template; consequently, PRODOC lists all the Laboratory Test Report sheets that are related with the CP instance. At this point, there is only one test report sheet that has been written by the lab. The

[4] At the present moment, this feature is not implemented in PRODOC.

pediatrician selects it, reads it from within the Data Panel and select the next action accordingly.

The evaluation activity can lead to either the decision of exiting the CP, because the test results exclude the possibility of a meningitis infection (negative case); or of carrying on with the next activity according to the CP schema (positive case). In the negative case, the pediatrician selects the CP Exit event and activates it. At this point, PRODOC proposes to provide a written justification for this decision and then it will terminate the CP instance. Although PRODOC allows users to provide a justification for every deviation from the intended process, it requires a mandatory note only whenever a CP is terminated, since this kind of event is highly critical, i.e., with important consequences for the patient progress. When a CP is terminated, PRODOC comes back to the Treatment activity of the Hospital Stay meta-process related to JD. At this point, the pediatricians can either keep using the whole documentation without the navigation aid of a CP map, create a new CP instance to cope with what turned out is not a regular case of meningitis, or terminate the treatment phase to trigger the administrative tasks of the Discharge phase for JD.

7 Conclusions

In this paper we have illustrated PRODOC, a system conceived: i) To allow designers build the interface of a documental systems starting from the interactions of practitioners with their usual artifacts and not from the data model that makes these interactions meaningful; ii) To support users in browsing and using these electronic artifacts in the light of the work processes they wish to externalize. Our approach is to support clinicians in leveraging process models for what they are intended to be, i.e., as pathways, maps they can rely on to orientate themselves in a wild territory of open choices and clear responsibilities. Since documental practices and working practices are often intertwined and mutually supporting each other – as cooperative work and articulation work usually are – the main tenet of our proposal is that to make documents and processes more integrated can help making practices more aligned to intended reference models of action, a point that at least for more structured models of patient record has been shown of some use Bossen (2006). To gain proof of this tenet, we deployed a prototype in the hospital domain, where patients are the resources being documented on personalized records, and Clinical Pathways are the processes according to which doctors cure patients. That notwithstanding, we believe the concept of PRODOC is applicable to any domain where documents are used as records in order to document events, findings and interventions that regard a specific resource within a practice that actors are supposed to align to a specific protocol, procedure or process.

Irrespectively of where PRODOC can be used[5], it fosters the externalization of work processes to capture, also in a graphical and visual way, the links between the

[5] We are planning to deploy PRODOC also in the archeological domain, where the documents under consideration are the excavation records and the resources under documentation are either the stratigraphic units or found artifacts.

procedural aspects of practice and the inputs/outputs that each activity usually consumes/produces. In this way, users can leverage visual active maps to page through data and make apt use of their records. Moreover, if the integration with legacy and organizational information systems is a necessary requirement, PRODOC can be seen as a sort of process-oriented front-end to data that is architecturally "on top of" the legacy system. This integration would then require that the legacy application exposes its data to PRODOC in terms of well defined and bounded "pages" (as in the case of web-based applications). If the underlying system does not have a steady concept of "page", PRODOC can provide "input forms" to the underlying data management system. Unfortunately, modern document systems have complex DBs inside and usually do not expose them: electronic patient records make no exception. This is for at least two reasons: first, document systems are generally "jealous" of their data, due to justified requirements of data protection and security. Second, the Active Server Pages (ASP) of a web-based document system gather and show data according to the internal state of the application process (i.e., of its workflow or business logic). This means, for instance, that linking process activities to the URLs of the document system's pages would be useless, even if the navigation system on top of the document system could pass it lots of parameters.

The moral here is that document systems (e.g., EPRs) have their inner workflow, and this is a priori different from, and often irreducible to, the case-specific process that users can adopt in PRODOC. For this reason, and to provide the proof of the PRODOC concept with respect to the tenets illustrated in Section 3, the current prototype embeds an electronic patient record that closely mimics paper-based charts and lets users define and update their process maps in a decoupled manner from document templates. Due to its innovative characteristics, we are now validating PRODOC in a set of informal user sessions from which we gained the first set of findings we report in Section 6. As output of these user sessions, we are considering to improve the interaction design of the user interface, and to extend PRODOC in terms of multi-role and profile management, as well of capabilities of transparent sheet sharing between different concurrent processes.

Acknowledgements

The work presented in this paper has been partially supported by the Italian fund FAR 2008. The authors would like to thank the management and the Internal Medicine and Neonatal Intensive Care Unit personnel of the Manzoni Hospital in Lecco (I) for their kind collaboration. In particular, we would like to acknowledge the invaluable help and courtesy of Dr Bellù.

References

Aigner, W. and S. Miksch (2006): 'CareVis: Integrated visualization of computerized protocols and temporal patient data'. *Artificial Intelligence in Medicine*, vol. 37, no. 3, pp. 203–218.

Bardram, J. E. and C. Bossen (2005): 'A web of coordinative artifacts: collaborative work at a hospital ward'. In: *GROUP '05: Proceedings of the 2005 international ACM SIGGROUP conference on Supporting group work*. New York, NY, USA, pp. 168–176, ACM Press.

Berg, M. (1997): 'Problems and promises of the protocol'. *Social Science and Medicine*, vol. 44, no. 8, pp. 1081–8.

Berg, M. (1999): 'Accumulating and Coordinating: Occasions for Information Technologies in Medical Work'. *Computer Supported Cooperative Work*, vol. 8, no. 4, pp. 373–401.

Berg, M. and B. Winthereik (2003): *Health Information Management: integrating information and communication technology in healthcare work.*, Chapt. Waiting for Godot. Routledge, London, UK.

Blaser, R., M. Schnabel, C. Biber, M. Baumlein, O. Heger, M. Beyer, E. Opitz, R. Lenz, and K. Kuhn (2007): 'Improving pathway compliance and clinician performance by using information technology'. *International Journal of Medical Informatics*, vol. 76, pp. 151–156.

Bossen, C. (2006): 'Representations at work: a national standard for electronic health records'. *CSCW '06: Proceedings of the 2006 20th anniversary conference on Computer supported cooperative work*, vol. 37, no. 3, pp. 69–78.

Boxwala, A., M. Peleg, S. Tu, O. Ogunyemi, Q. Zeng, D. Wang, V. Patel, R. Greenes, and E. Shortliffe (2004): 'GLIF3: a representation format for sharable computer-interpretable clinical practice guidelines'. *Journal of Biomedical Informatics*, vol. 37, no. 3, pp. 147–161.

Bringay, S., C. Barry, and J. Charlet (2006): 'Annotations: A Functionality to support Cooperation, Coordination and Awareness in the Electronic Medical Record'. In: *COOP'06: Proceedings of the 7th International Conference on the Design of Cooperative Systems*. France, Provence.

Cabitza, F., M. Sarini, and C. Simone (2007): 'Providing awareness through situated process maps: the hospital care case'. In: *GROUP'07: Proceedings of the 2005 International ACM SIGGROUP Conference on Supporting Group Work*. New York, NY, USA, pp. 41–50, ACM.

Cabitza, F., M. Sarini, C. Simone, and M. Telaro (2005): '"When Once Is Not Enough": The role of redundancy in a hospital ward setting'. In: M. Pendergast, K. Schmidt, G. Mark, and M. Ackerman (eds.): *GROUP'05: Proceedings of the 2005 International ACM SIGGROUP Conference on Supporting Group Work*. Sanibel Island, Florida, U.S.A., pp. 158–167, ACM Press.

Cabitza, F., C. Simone, and M. Sarini (2008): 'Knowledge Artifacts as Bridges between Theory and Practice: the Clinical Pathway case.'. In: *KMIA'08: Proceedings of the International Conference on Knowledge Management In Action.* Milan, Italy, IFIP.

Campbell, H., R. Hotchkiss, N. Bradshaw, and M. Porteous (1998): 'Integrated care pathways'. *British Medical Journal*, vol. 316, no. 10, pp. 133–137.

Fitzpatrick, G. (2000): 'Understanding the Paper Health Record in Practice: Implications for EHRs'. In: *HIC'2000 Proceedings of Health Informatics Conference, Adelaide, AU, 2000.*

Fitzpatrick, G. (2004): 'Integrated care and the working record'. *Health Informatics Journal*, vol. 10, no. 4, pp. 291–302.

Hardstone, G., M. Hartswood, R. Procter, R. Slack, A. Voss, and G. Rees (2004): 'Supporting informality: team working and integrated care records'. In: *CSCW '04: Proceedings of the 2004 ACM conference on CSCW*. New York, NY, USA, pp. 142–151, ACM Press.

Harper, R. H. R., K. P. A. O'Hara, A. J. Sellen, and D. J. R. Duthie (1997): 'Toward the Paperless Hospital? A Case Study of Document Use by Anaesthetists'. *British Journal of Anaesthesia*, vol. 78, pp. 762–767.

Hartswood, M., R. Procter, M. Rouncefield, and R. Slack (2003): 'Making a Case in Medical Work: Implications for the Electronic Medical Record'. *Computer Supported Cooperative Work*, vol. 12, pp. 241–266.

Heeks, R., D. Mundy, and A. Salazar (1999): 'Why health care information systems succeed or fail'. Technical report, Institute for Development Policy and Management (IDPM), Manchester, UK.

Koppel, R., J. P. Metlay, A. Cohen, B. Abaluck, A. R. Localio, S. E. Kimmel, and B. L. Strom (2005): 'Role of Computerized Physician Order Entry Systems in Facilitating Medication Errors'. *Journal of the American Medical Association*, vol. 293, pp. 1197–1203.

Lenz, R., R. Blaser, M. Beyer, O. Heger, C. Biber, M. Bäumlein, and M. Schnabel (2007): 'IT support for clinical pathways – Lessons learned'. *International Journal of Medical Informatics*, vol. 76, pp. 397–402.

Morrison, C. and A. F. Blackwell (2009): 'Observing End-User Customization of Electronic Patient Records'. In: V. P. et al. (ed.): *End-User Development*, Vol. 5435 of *Lecture Notes in Computer Science*. pp. 275–284, Springer-Verlag.

Mulyar, N., M. Pesic, W. van der Aalst, and M. Peleg (2008): 'Declarative and Procedural Approaches for Modelling Clinical Guidelines: Addressing Flexibility Issues'. *Lecture Notes In Computer Science*, vol. 4928, pp. 335.

Pedersen, T. B. and C. Jensen (2001): 'Multidimensional database technology'. *Computer*, vol. 34, no. 12, pp. 40–46.

Quaglini, S., S. Panzarasa, A. Cavallini, G. Micieli, C. Pernice, and M. Stefanelli, 'Smooth Integration of Decision Support into an Existing Electronic Patient Record'. In: *AIME 2005, Proceedings of the 10th Conference on Artificial Intelligence in Medicine, Aberdeen,UK, July 23-27, 2005*. pp. 89–93.

Schmidt, K. (1997): 'Of maps and scripts: the status of formal constructs in cooperative work'. In: *GROUP'97: Proceedings of the GROUP Conference*. Phoenix Arizona USA, pp. 138–147, ACM Press.

Shahar, Y., S. Miksch, and P. Johnson (1998): 'The Asgaard project: a task-specific framework for the application and critiquing of time-oriented clinical guidelines'. *Artificial Intelligence In Medicine*, vol. 14, no. 1-2, pp. 29–51.

Short, M. (1997): 'Charting by exception on a clinical pathway'. *Nursing Management*, vol. 28, no. 8, pp. 45–46.

Sloan, M. D. and C. S. Guinane (1999): *Analyzing Clinical Care Pathways*. McGraw-Hill Professional.

Smart, J. F. and M. Roux, 'Medical Knowledge Representation for Medical Report Analysis'. In: *AIME'95, Proceedings of the 5th Conference on Artificial Intelligence in Medicine in Europe, Pavia, Italy, June 25-28, 1995*. pp. 53–64.

Sutton, D. and J. Fox (2003): 'The syntax and semantics of the PROforma guideline modeling language'. *Journal of the American Medical Informatics Association*, vol. 10, no. 5, pp. 433–443.