Published in Methods of Information in Medicine, vol. 44, no. 1, 2005, 14-24

Design and Development of a Mobile System for Supporting Emergency Triage

Wojtek Michalowski, Ph.D.
School of Management, University of Ottawa
136 Jean-Jacques Lussier St. Ottawa,
Ontario K1N 6N5, Canada,
tel. (613) 562-5800 (x. 4955), fax (613) 562-5164,
e-mail: wojtek@management.uottawa.ca

Roman Slowinski Ph.D., Szymon Wilk* Ph.D.
Institute of Computing Science, Poznan Technical University

Ken J. Farion MD FRCPC, John Pike MD FRCSC, Steven Rubin MD FRCSC FRCSE
Children's Hospital of Eastern Ontario, University of Ottawa

* Research described in this paper was conducted while Dr. Wilk was a postdoctoral fellow in the School of Management, University of Ottawa.

Summary

Objectives
Our objective was to design and develop a mobile clinical decision support system for emergency triage of different acute pain presentations. The system should interact with existing hospital information systems, run on mobile computing devices, and be suitable for operation in weak-connectivity conditions (with unstable connections between mobile clients and a server).

Methods
The MET (Mobile Emergency Triage) system was designed following an extended client – server architecture. The client component, responsible for triage decision support, is built as a knowledge-based system, with domain ontology separated from generic problem solving methods and used for the automatic creation of a user interface.

Results
The MET system is well suited for operation in the Emergency Department of a hospital. The system's external interactions are managed by the server, while the MET clients, running on handheld computers are used by clinicians for collecting clinical data and supporting triage at the bedside. The functionality of the MET client is distributed into specialized modules, responsible for triaging specific types of acute pain presentations. The modules are stored on the server, and on request they can be transferred and executed on the mobile clients. The modular design provides for easy extension of the system's functionality. A clinical trial of the MET system validated the appropriateness of the system's design, and proved the usefulness and acceptance of the system in clinical practice.

Conclusions
The MET system captures the necessary hospital data, allows for entry of patient information, and provides triage support. By operating on handheld computers, it fits into the regular Emergency Department workflow without introducing any hindrances or disruptions. It supports triage anytime and anywhere, directly at the point of care, and also can be used as an electronic patient chart, facilitating structured data collection.

Keywords
Decision Support Systems, Clinical; Triage; Computers, Handheld; Emergency Service, Hospital

Published in Methods of Information in Medicine, vol. 44, no. 1, 2005, 14-24

1. Introduction

A clinical decision support system (CDSS) is defined as "any program designed to help health-care professionals make clinical decisions" [1]. This broad definition covers several categories of computer-based systems, including:

- *Systems for information management*. They provide general data and knowledge for clinicians. Depending on the type of provided information, they are further categorized into medical information retrieval systems for managing and extracting medical knowledge, and electronic patient record systems (EPRS [2]) for managing patient data.
- *Systems for focusing attention*. They are used to remind clinicians about actions that might require attention. Such systems are used in the ICU or laboratories to flag abnormal readings, in pharmacies to check and notify about possible drug interactions, and in clinical practice to remind physicians about certain tasks (e.g., immunizations) and alert about potential risks (e.g., patient drug allergies).
- *Systems for providing patient-specific recommendations*. They provide assessments or advice based on patient-specific clinical data. Systems from this category use diversified techniques to make decisions or to provide support in clinical decision making, from direct implementation of approved clinical practice guidelines [3] to advanced techniques of artificial intelligence [4].

CDSS from the first two categories have been well accepted and used in clinical practice for more than three decades [5] to deal with the constantly growing amount of clinical information that has to be collected, processed and analyzed. Increasing interest in systems from the third category is driven by the necessity to make evidence-based and cost effective decisions [6], and the efforts to improve healthcare processes and patient outcomes [7]. Patient-specific systems help clinicians make two types of decisions – diagnostic (what is the underlying health condition of the patient) and management (what to do for the patient). Although it is rather artificial to separate the diagnostic process from the management one, many clinicians believe that it is for the management process that they would most often seek support [1].

Emergency triage is the first stage of the patient management process and it takes place as soon as the patient arrives at the hospital's Emergency Department (ED). It establishes the timing and type of care that needs to be given to the patient. CDSS supporting triage are rarely used in clinical practice, with few reported successful applications [8]. This situation can be attributed to the specificity of the ED workflow where clinicians work under time constraints and pressure, constantly moving from one patient to another. There is usually no place in the ED examination rooms to set up a clinical workstation that could host a CDSS, and entering data or consulting the triage decision on a workstation located elsewhere interrupts the workflow, causes delays and constitutes a significant hindrance [9]. Thus, despite apparent advantages, triage-supporting CDSS that rely on conventional computing platforms seem to have no place in an ED setting.

The need to overcome the shortcomings and limitations of conventional CDSS has led to the recent concept of m-health [10]. M-health, being an extension of e-health [11], and understood as providing required support in an ubiquitous manner [12], favors support systems that are available directly at the point of care, and thus can be easily integrated into the workflow.

Published in Methods of Information in Medicine, vol. 44, no. 1, 2005, 14-24

Because of these advantages, mobile solutions have stimulated the interest of healthcare professionals and become accepted in clinical settings. The majority of currently available m-health CDSS are drug databases and patient tracking systems [13]. There are few mobile patient-specific systems [14, 15], with most of them being implemented as simple medical calculators or scoring procedures [13].

Responding to the challenge of supporting ED triage, and taking advantage of the emergence of m-health, we developed the Mobile Emergency Triage (MET) system [16] to facilitate the triage of various acute pain presentations. MET belongs to the third category of CDSS – patient-specific systems, and it provides a triage suggestion on the basis of collected clinical information, using techniques based on artificial intelligence. Specifically, MET is a knowledge-based mobile CDSS with clinical knowledge presented explicitly in the form of decision rules. The system works as a consultant – it facilitates the patient's triage by presenting possible management outcomes together with the strength of their recommendation, reflecting the uncertainty inherent in clinical reasoning. MET has a modular and open architecture with separate clinical modules associated with various acute pain problems, so it can be easily extended to handle new presentations. The system offers triage support anytime and anywhere (directly at the point of care). Moreover, it is integrated with EPRS and other hospital information systems (e.g., laboratory systems) to extract data already available in electronic form (e.g., information collected during patient's registration and results of the laboratory tests) and to allow clinicians to transfer the results of the ED examination to the EPRS after finishing with the patient. Thus, MET can be also considered as an electronic patient chart, extending the EPRS.

The paper is organized as follows. In the next section we discuss the workflow in the ED, and how MET fits into it. Next, we present the requirements that should be satisfied by any patient-specific CDSS, and illustrate how we designed the MET system to adhere to them. We then describe a clinical trial, designed and conducted to verify the usefulness of MET in clinical practice. Finally, we end with a discussion.

2. Triage of Acute Pain Presentation and ED Workflow

Patients presenting to the ED with acute pain problems undergo triage, which consists of two assessment phases that respectively answer the following two questions: how quickly does the patient need to receive medical attention, and what type of management is necessary? Figure 1 illustrates the workflow associated with the triage process, and represents the basis for the MET system design.

The first assessment phase, called *prioritization*, is done by a triage nurse who evaluates the severity of the patient's clinical condition. The patient is assigned an appropriate priority level that determines how quickly the ED physician should see the patient. Patients assigned a high priority are seen immediately by the physician, while lower priority patients may wait for prolonged periods of time.

The second phase, *medical assessment and disposition,* is aimed at establishing the management path for the patient's problem. It involves the physician evaluating the patient's history, conducting a physical examination, and ordering diagnostic tests. This results in one of the following disposition decisions (triage recommendations):

- *Discharge*, which implies that the patient has a benign and/or resolving presentation and can be sent home with possible follow-up with the family physician.
- *Observation/further investigation*, which means that continuing in-hospital observation and/or investigation for a potentially serious cause of a presentation is required.
- *Consult*, which indicates that a request for urgent specialist consultation (e.g., surgical consult for appendicitis, or urology consult for testicular torsion) and further management are required.

In the research reported in this paper, we are concerned with the *medical assessment and disposition* phase, and for simplicity, we further refer to this phase as the triage, and to the disposition decisions as the triage decisions.

There are two general roles that a CDSS may play in supporting the clinical workflow [1]:

- *Consultant*, when the system serves as an advisor that accepts patient-specific data, asks additional questions and generates advice.
- *Critique*, when the system acts as a "sounding board" that evaluates clinicians' preconceived management options, evaluates them, or suggests reasoned alternatives.

We designed MET to be a consultant that can be used anytime during the triage of a patient. We believe that the experience of the healthcare professional working with the system is a key factor influencing the usage of MET. Experienced physicians are likely to consult MET when the triage is almost completed in order to verify the likely triage recommendation. In the hands of less experienced physicians, MET may be used early in the triage process to help with structured data collection, and to assure that the disposition is not based on incomplete or irrelevant data. This flexible *modus operandi* of MET in the ED does not require any changes in the management process or in the ED workflow, thus improving chances of the system's being accepted in the ED practice.

3. The MET System

3.1. Requirements for a Patient-Specific CDSS

CDSS are used in different settings and by different users. Despite such diversity, there is one common need that they have to fulfill in order to be successful in clinical practice – the fit into the clinical workflow [1, 5]. The use of CDSS should not introduce any significant changes into the regular clinical routine or into well-established procedures (i.e., clinicians cannot be forced to leave patients in order to consult a system running on a remote workstation); otherwise, the system will be rejected, despite its functionality [5]. Ideally, a CDSS should seamlessly integrate with the workflow and eventually become a "byproduct" of the physicians' regular work practice [1].

Patient-specific CDSS, apart from fitting into the workflow (which also means not trying to replace the clinicians [1, 17]), have to satisfy two equally important additional needs [1, 4]:

- They should offer reliable and comprehensive advice. Clinicians are the ultimate decision makers and this is not likely to change for a variety of reasons [4]. Therefore, they will

Published in Methods of Information in Medicine, vol. 44, no. 1, 2005, 14-24

only accept systems acting as consultants that provide additional information and advice, but allow them to make the final decision. Such systems not only have to offer acceptable reliability (specificity and sensitivity), but also need to be able to explain and justify their recommendations.

- They should integrate with the EPRS and other hospital information systems. Such integration allows the users to access data that are already stored in electronic form (e.g., the details of presentation collected during registration and prioritization) and saves users from re-entering available information. This is why the integration is often considered as another factor (together with the fit into the workflow) for the successful use of advising systems in clinical practice [2].

When designing the MET system, we had to transform these three high-level needs (fitting into the workflow, offering reliable advice, and integrating with the EPRS) into low-level requirements and specific design specifications. We started with the fit into the workflow. The medical personnel in the ED have to make clinical decisions under stress and time constraints [18] directly at the point of care. The physician doing triage does not have time to consult a remote workstation (most examination rooms in the ED do not have enough space to accommodate a computer), and therefore should be equipped with a "handy" support tool that could be used when and where it is needed. Such support at the point of care has only recently become possible with the technological advances [12] that have led to the concept of m-health [10]. This concept has just started being used in the design of CDSS [13-15], but considering that such systems have the capability of providing seamless integration with the workflow [19], we opted for the m-health design of MET.

Mobile devices offer limited computing capabilities in comparison with desktop systems. They are not capable of storing large amounts of data and performing computationally demanding tasks (this is why they are called "lean" platforms). When designing MET, we addressed those issues by distributing the system's functionality between clients and a server that would store the data and be responsible for intensive processing (e.g., exchanging data with other clinical systems). The most convenient method of communication between the server and the clients is through wireless networking [20]. However, its availability in a hospital cannot be taken for granted, due to the varied adoption of this technology by hospitals over security concerns and possible interference with other medical equipment. Therefore, when designing MET, we followed the extended client-server paradigm [21] that is suited for weak connectivity, where permanent connection between clients and a server cannot be maintained, and where extended clients provide crucial functionality without contacting the server. In the case of MET, the functionality on the client side included entering and storing locally data that were collected during the patient's examination, and supporting triage decisions.

The MET system needs to communicate and exchange data with the EPRS and other hospital information systems (e.g., laboratory systems) to relieve clinicians from entering data already available in electronic form, and to provide access to the most recent and complete information. We ensured such a system's interoperability by making the MET server capable of receiving, processing, and sending messages (trigger events [1]) generated by the EPRS and other systems. This implied making the server aware of standard protocols used in a hospital for exchanging messages and terminologies used for coding their content.

In order to be accepted in practice and gain the confidence of clinicians, MET has to mimic a trustworthy consultant by offering a reliable support function, and the capability to explain its recommendations. When designing the support function of the system we decided to use decision rules extracted from historical data to construct decision models for triaging specific acute pain presentations. Decision rules represent clinical knowledge similarly to practice guidelines, and thus they are easy for clinicians to understand and to accept (e.g., the Ottawa Ankle Rule [22]). Medical experts can also verify such knowledge before it is embedded into the system. Moreover, the use of decision rules is transparent [4] and it offers extensive capabilities of explaining and justifying suggested decisions (e.g., specific rules that led to certain recommendations can be presented to the clinician). Finally, our earlier research on developing rule-based clinical decision models demonstrated that such models achieved satisfactory accuracy of triage recommendations [23].

In the following subsections we present the detailed design of MET from two perspectives – triage support available on the client, and synchronization and integration provided by the server.

3.2. Design of the Triage Support Component

Design of the triage support component was a challenging task. On the one hand, it had to support various acute pain presentations, but on the other hand, it had to operate on a "lean" mobile device, with limited memory and computational power. The easy solution would have been to design a dedicated support system for each presentation of pain, and to upload these systems on the client when required. However, it would limit the flexibility and reusability of the solution. Moreover, it could easily lead to the critical mass issue (when users would not be able to navigate between applications supporting very specific tasks) [24]. To avoid those problems we employed the design inspired by the research on decision support systems with separated domain ontologies and domain independent, generic solving methods (also called solvers) [25, 26].

Domain ontology is enumeration and organization of concepts in a problem area. It defines classes of concepts (e.g., patient, visit) and the relationships among these classes (e.g., there is at least one visit for a patient). Domain ontology is used for building knowledge bases with instances of described concepts (using database terms, an ontology can be viewed as a schema of a database, with a knowledge base as its content [25]). A solver is a general strategy that can be applied to any domain ontology without the need for customization.

Building a knowledge-based system using independent blocks, domain ontologies and generic solvers, results in reusability of these blocks and flexibility of created systems, because the same ontology can be used by many solvers, and one solver can utilize many ontologies. Moreover, this approach allows one to easily decompose complex systems into much simpler ones that could be assembled together on request. The ontology – solver separation is not only important from a systems engineering perspective, but it flows directly from the research in medical informatics. Ontologies are the ways of presenting conceptualizations of health-related application areas in machine-understandable formats, and generic solvers represent fundamental algorithms for processing clinical data and knowledge [26].

In the MET system we use a common basic ontology that is specialized for each type of acute pain, and thus each clinical module. The basic ontology contains four classes of concepts:

Published in Methods of Information in Medicine, vol. 44, no. 1, 2005, 14-24

patient, *visit*, *presentation*, and *rule*. A patient has at least one visit associated with a single presentation of a clinical problem that describes the patient's clinical attributes (symptoms, signs and tests). *Presentation* remains an abstract concept (no entities of this concept are created) that is specialized to specific acute pain presentations by adding appropriate attributes. Figure 2 presents the ontology specialized for presentation of abdominal pain, and we have also developed ontologies for presentations of scrotal pain and hip pain, thus proving the extensibility and scalability of our design principles. Finally, *rule* represents a decision rule used by the solver to establish the recommended triage decision. It relies on the concept of presentation to characterize the patient's condition, and becomes usable when the basic ontology is specialized.

The knowledge base in the MET system contains decision rules that were created from data describing past cases using automatic knowledge discovery techniques. The scope of the knowledge base is determined by a specialized presentation concept and clinical attributes used in the ontology. An automatic approach to constructing the knowledge base allowed us to avoid the Feigenbaum bottleneck [27] associated with acquiring knowledge from experts and from medical literature. The data used to generate decision rules come from patients' ED charts, evaluated and transcribed as part of a retrospective chart study, and augmented with verified patients' outcomes. This approach to knowledge discovery, though having some drawbacks [28], allows for an encapsulation of clinical knowledge that attempts to capture the clinical acumen of experienced physicians. The decision rules used in MET were generated using knowledge discovery techniques based on rough set theory [29-32]. The usefulness of rough set theory comes from its capability to process symbolic and numeric values, to deal directly with inconsistent and incomplete information [33] without diminishing the robustness of the results. Clinical data usually include a mixture of numeric and symbolic values (e.g., temperature and the type of pain), describe patients characterized by the same values of clinical attributes but having different disposition decisions (there might be some information that was applied to make the final decision, but it is not included in the charts, thus differentiating between patients is impossible), and usually have a large number of not specified (missing) values (especially for laboratory tests that are not ordered for every patient).

The MET system relies on decision rules to triage a patient, and it currently uses one solver, the similarity-based inference engine [34], that proved to be effective in previous applications of triaging acute pain [16, 35]. The solver can arrive at a decision even if the patient's data are incomplete or inconsistent with decision rules from the knowledge base (it fires partially matched rules that are most similar to the available data); however, the strength of the recommendation would then be decreased. The solver can be easily updated or expanded if a better solving method is found, and it is also possible to add new solvers, providing that the domain ontology is extended accordingly. We have plans to modify the domain ontology to include clinical pathways, and to add an appropriate solver to handle this functional extension.

Domain ontologies are not only the building blocks for knowledge-based systems; they can also be used for the automatic construction of user interfaces [36]. Such interfaces reflect the ontologies on which they are based, and self-adapt whenever the ontologies are modified. To generate an interface, the ontology has to be expanded by additional information: for example, what selectors (high-level representations of user-interface components [37]) to use for specific attributes of defined concepts. Although the idea of automatically generated interfaces was developed and applied in developing the knowledge acquisition tools for creating and updating knowledge bases [38], we decided to employ this methodology to create the interface for entering

Published in Methods of Information in Medicine, vol. 44, no. 1, 2005, 14-24

and presenting the values of clinical attributes collected during the patient's assessment. The fact that MET is intended to be used in a specific, well-defined domain (i.e., the triage of acute pain), simplifies the creation of the interface, as most of its components are common for different types of acute pain presentations, and only a few have to be customized. Limited customization does not require extensive computations and can be done directly on a mobile device whenever a patient with a specific presentation is evaluated.

The ontology for the automatically created user interface of the MET system is illustrated in Figure 3. It is composed of the interface objects belonging to two basic classes: forms and selectors. The form object is used to present information, while the selector object is used to enter and modify values of clinical attributes. Customization is required for attributes forms, as they have to present the values of attributes considered for a specific pain presentation, and for selectors utilized for entering data, because they have to be adjusted to the domain of an attribute (a set or a range of possible values). Table 1 presents the specification of the customized interface for the presentation of abdominal pain.

The form and selector objects were designed with the specificity of the clinical task, the needs of the clinicians [39], and the specificity of mobile computing devices in mind. MET has to support interactions that are timely and easy to follow, allowing the user to focus on the task at hand and to perform the triage with no increase in the cognitive burden associated with use of the system (imposed, for example, by its technical specificity). At the same, time it has to work on a mobile device with limited means of interaction (much smaller displays, data entry techniques limited to touch sensitive screens and handwriting recognition). This makes many of the modalities routinely employed in desktop systems useless here. Therefore, the interface was designed according to user-oriented [40] and task-oriented [41] design methodologies, and following the specific guidelines for clinical systems [42] and mobile devices [43]:

- Presentation format should enforce clinicians' focus on the key data,
- Terms should be familiar to the clinicians and should relate to triage,
- Data entry should minimize the chances for transcription mistakes,
- Access to and support for routine tasks should be straightforward,
- Navigation cues and the system's options should be easily accessible.

Sample implementations of form objects are given in Figures 4 and 5. They present the navigation form providing access to the different MET functions, and the attributes form customized for specific pain presentation with the values of attributes and with navigation cues. All terms and shortcuts on these forms (e.g., Hx, PE or Ix) come from medical practice, and clinicians were consulted regarding their use. Figures 6, 7, and 8 illustrate the implementation of different selector objects that are used for entering the values of different attributes (with a finite set of values, with numerical values, and with values represented graphically on a pictogram). The selectors were designed to make the data entry more efficient (faster) and less prone to transcription errors, and to minimize the amount of information entered through handwriting recognition (it is still supported for expert users; however, less advanced ones can operate MET by clicking on buttons and images). Moreover, whenever possible, we used images, pictograms, and symbols specific for the triage task and familiar to clinicians. It was reported in the literature that such a structured data collection alone may improve physicians' clinical performance [44-46].

Published in Methods of Information in Medicine, vol. 44, no. 1, 2005, 14-24

The architecture of the MET triage support component is given in Figure 9 and it illustrates the idea of separating the domain ontologies from generic solvers and using the same solver with many ontologies. The functionality of the triage support component of the MET system derives from the clinical modules and the shell. The clinical module contains the domain model with specialized ontology, and the knowledge base with the decision rules that form the decision model. The domain model and the knowledge base are consulted by the shell to customize and execute the interface, and to give the triage recommendation.

The shell consists of the interface repository (with interface objects constructed according to the interface ontology), the solver, and the executor, which cooperate together, but are independent. They communicate with one another by using a well defined set of services, and thus they can be easily modified or replaced, provided that their services remain unchanged. The executor is responsible for the majority of tasks performed by the shell while handling a patient with a specific presentation of pain. First, it invokes the clinical module specific to the presentation. It then uses the information stored in the domain ontology to customize the interface, runs it, and interacts with the clinician. When the clinician decides to enter or modify the values of the clinical attributes, the executor retrieves and customizes the appropriate interface object from the interface repository, and when the clinician requests a triage recommendation, the executor invokes the solver. The solver operates on the decision model from the clinical module to produce a recommendation and passes the result to the executor, which in turn communicates it to the clinician. To reflect the uncertainty inherent for clinical reasoning, and to give a better insight into the recommendation, the executor does not present the unequivocal outcome, but instead it reports all possible dispositions together with their strengths (see Figure 10), letting the clinician make the final decision (initially MET also presented rules that led to a specific disposition, but we had to give up that idea because of the difficulties associated with displaying the rules in a readable format on the small display of a mobile device).

Separation of the clinical modules and components constituting the shell supports easy modifications (updates are bounded to a single module) and deployment of new modules without modifying the whole triage support component. Moreover, linking the interface creation task with the functionality of the executor, and limiting the required customization to a few elements, allows the creation of the customized interface during run time when the appropriate clinical module is being invoked. This approach gives the clinician the impression of working with a single application. In fact, he/she works solely with the executor, and there is no need to manually switch between different clinical modules by quitting an active one and activating another, as would be required in the case of stand-alone specific applications that are compiled in advance and installed on a handheld. Thus, the danger of the critical mass problem is successfully addressed. Finally, careful design of the interface with the specificity of the task, the clinician, and the computing platform in mind, leads to a solution that is convenient, usable and does not deter clinicians from routine management.

3.3. Design of the Synchronization and Integration Component

Weak connectivity (i.e., infrequent connections between a client and a server) calls for an extended client-server architecture [21], also known as flexible client-server architecture [47], where the client takes over some functions of the server when a connection is not available. The architecture of the MET system is presented in Figure 11. The MET server integrates with the

EPRS and other clinical systems via the integration subsystem. It also communicates with clients, using the sync subsystem to exchange patients' data, and to send the necessary clinical modules associated with specific presentations (the modules are stored on the server and executed on the client side). The MET client provides facilities to collect clinical data, it retrieves required clinical modules, supports triage decisions, and synchronizes collected information with the server.

The ability to provide support without connection with the server requires the triage support component (i.e., the clinical modules and the shell) and the local patient database (used for storing data received from the server and collected during examination) to be located on the client side. Unlike classical knowledge-based systems, where knowledge supplied by the experts and case data entered by the users are stored in the same knowledge base, the MET system maintains the case data in a separate database to simplify their management and facilitate data synchronization with the server.

Weak connectivity and the need to integrate with the EPRS necessitate the use of the temporal database on the server side. It plays the role of a buffer that stores information before it can be transferred to the clients or back to the EPRS, thus minimizing the number of interactions and eliminating the need to query the EPRS whenever clinical data are to be sent to the client.

The MET server communicates with the EPRS and other hospital information systems using the HL7 (Health Level 7) protocol [48]. HL7 is the standard of exchanging information between medical applications. It specifies the format and content of the exchanged data; however, it defines neither how messages are passed between applications nor how applications process this information. This flexibility allows one to use various network protocols and to integrate diverse systems without extensive modifications. Although HL7 ensures that information is passed between various clinical systems, it does not ensure its interoperability (different systems may use different terms referring to the same piece of information). The latter is achieved by relying on standards for medical terminologies, such as SNOMED-CT [49] and ICD-10 [50] for diagnoses or LOINC [51] for observations and results of laboratory tests.

When a new patient is registered in the EPRS, an admission message (ADT trigger event) with demographic information and presenting complaint is generated and transmitted by the HL7 interface engine to other clinical systems, including MET. The integration subsystem on the MET server decodes it and stores received data in the temporal database, where they await the next synchronization with any of the mobile clients. Moreover, when new data are reported to the MET server by any other system using an observation report (ORU trigger event), the server makes necessary updates and keeps the recent values. As in the case of admission data, the new results are transmitted to clients during the subsequent synchronization. Whenever data in the temporal database are updated by any of the clients, the integration subsystem sends messages with observation reports to the interface engine, which in turn passes them to the EPRS and remaining systems, thus maintaining consistency of shared information. At present, the MET server sends messages when the examination and triage of a patient are completed. Exchange of HL7 observation reports is limited only to these clinical attributes that are defined in domain ontologies of acute pain presentations handled by the system, so the server does not accept and store redundant information.

Although there are medical terminologies for precise specification of diagnostic evaluations, interventions, laboratory results and observations or drugs, there is still no unified terminology to capture a medical history or findings on physical examinations [1]. As the majority of information processed by MET comes from the examination in the ED and describes the patient's history and physical findings, we follow the coding standard established at the Children's Hospital of Eastern Ontario (CHEO) in Ottawa, Canada[†]. Only results of laboratory tests are presented according to LOINC. Such an approach ensures that transferred information is correctly interpreted and processed by other clinical systems. Moreover, if the appropriate terminology is introduced, the system can easily accommodate it by updating the integration subsystem on the MET server.

The server and the client can synchronize their data using wired or wireless connection. Synchronization involves sync subsystems of the client and the server. It consists of two phases. In the first phase, patients' records are synchronized and the process is bi-directional (i.e., data are transferred from the client to the server and then from the server to the client). After obtaining data from the client, the server updates the temporal database and keeps the most recent values. Then the server sync subsystem sends data to the client. To limit the amount of transmitted data, only records of active patients are transferred – records are active for a specified time window (by default it is 6 hours) after a patient is admitted and registered in the EPRS, and when the active time window elapses, the record is deactivated and data are no longer synchronized. The client sync subsystem updates the contents of the local database by overwriting existing records with received ones and removing inactive records.

During the second synchronization phase, the server sends the necessary clinical modules to the client. First, the server sync subsystem sends those modules that are required to handle patients' records transferred in the first phase, and that are not currently available on the client side. A presenting complaint is stored with the demographic data sent by the EPRS, so it is known what clinical modules are necessary to handle these records. The server sync subsystem then sends the modules explicitly requested by the client. It may happen that the initial complaint given during registration is not correct, and another clinical module is necessary to triage the patient. If the module is stored on the client side, the clinician may switch immediately and continue evaluation. However, if it is not available, the client sync subsystem requests it during the next synchronization. After receiving the requested clinical modules, the client sync subsystem removes all unnecessary ones. A module becomes unnecessary if there are no records in the local database that require processing by a given clinical module.

By integrating with the EPRS, the MET system saves clinicians' time. Moreover, by sending back values collected during the examination in the ED, the MET system can be also considered as an electronic extension of the patient's chart, allowing for structured and convenient data collection and storage in an electronic format. We should stress that the patients' data stored on handheld computers and transferred to the MET server require solutions for protecting privacy and security. The access to the local database is authenticated by a password that must be supplied when the MET client starts. Moreover, the content of the database is encrypted, and it cannot be read without supplying the encryption key. Finally, when the clients synchronize information with the server, a secure protocol (Transport Layer Security [52]) is used.

[†] CHEO is a teaching hospital that participated in the research described in this paper.

Published in Methods of Information in Medicine, vol. 44, no. 1, 2005, 14-24

4. Clinical Evaluation

The MET system with the abdominal pain module and its fit into the ED workflow were prospectively evaluated during a clinical trial at CHEO. Because of ethical concerns regarding the possible influence of the system's recommendations on the patient management process, clinicians were blinded to the triage recommendation provided by MET. Instead, they were able to enter their own triage decisions. These were later reviewed and compared to the MET recommendations and to the gold standard triage disposition (validated during a follow-up and a chart audit). Thus, the functionality of the system was confined to fit the workflow, and the support function was invoked in the background and verified off-line, after completing the management of the patient.

The design of the clinical trial is presented in Figure 12. After the patient was registered and prioritized, the system checked to see if the presenting complaint was abdominal pain. Patients with other presentations were automatically excluded from the trial (the MET server filtered out admission messages generated for such patients) and those with abdominal pain were subsequently asked to consent to the trial participation (consent was requested from the child's guardian). After obtaining consent, the eligibility of the patient was verified against a set of inclusion/exclusion criteria. If verification was positive, the patient was included in the trial, and the clinician used MET to enter clinical data collected during assessment, and to record his/her triage dispositions. In all cases, documentation of the patient's visit was also completed on the paper ED chart.

The follow-up and chart audit were conducted about two weeks after the patient's visit to the ED to determine the patient's final diagnosis. This included a telephone call to check if other medical assistance was required after visiting the ED. The hospital chart was reviewed for the results of any further visits, tests or consultations. The final diagnosis, established during the follow-up, determined the gold standard for the triage recommendation that was used while calculating and comparing the accuracy of the ED physicians and the MET.

The trial at CHEO lasted for 8 months, and patients were recruited 24/7. During that period 2255 patients with abdominal pain visited the ED. Of 1098 patients eligible to participate, 632 were approached to participate, and only 38 did not consent. Finally, 574 patients were verified during the follow-up. Patients and parents did not have problems participating in the trial, as the use of MET did not introduce additional examinations or invasive tests, and did not lengthen their time spent in the ED.

The MET system was used by over 100 members of the ED clinical staff (physicians and residents) who had not been involved in the design and development of the system. This group had diverse prior experience with handheld computers, ranging from complete novices to advanced users experienced with medical applications. All users participated in short orientation sessions (half an hour) and were able to operate the MET client without any difficulties.

The overall accuracy of the MET system was slightly lower but not statistically different from the accuracy of the ED physicians (70.2% for physicians vs. 67.2% for MET). Though the utility of a tool with only 67% accuracy may be questioned, we maintain that the performance of the MET system is satisfactory, considering that it mirrors that of experienced physicians in a teaching hospital, and is used at an early stage of the management process. It is also important to stress that MET's triage accuracy was calculated considering only the recommendation that acquired the highest strength, while the clinician using the system would be able to see all triage recommendations with relative strength recommendations.

5. Discussion

The management of patients with acute pain presentations in the ED requires repeated evaluation of the patients' history, physical findings and investigations. On the basis of this assessment, patients are triaged to an appropriate management path. Thus, any CDSS deployed in an ED needs to have the ability to mimic this process by having information-gathering capabilities, the ability to provide a triage recommendation, and it must be available anytime and anywhere at the point of care. The MET system follows the general framework proposed for medical aid tools for patient management support [53]. At the same time, it significantly expands the scope of these tools by allowing for structured data capture, storage and transmission to and from the EPRS, and providing the triage support function anytime and anywhere. Due to the fact that it runs on a handheld computer that is familiar to many clinicians, and since the system requires no special networking setup, it can be easily introduced and used in a variety of ED environments. The system's modular design makes it very easy to modify its clinical functionality, or to expand its scope by adding new clinical modules. This design solution allows us to extend the system's list of possible mobile clients to other mobile devices, such as tablet computers or advanced cell phones (so called, smart phones), and this research is currently underway. Our clinical evaluation of the MET system in the ED of a teaching hospital is one of very few cases where a computer-based support system has been actually tested in clinical practice. The 24/7 trial design implied that MET had to operate flawlessly without any external support, and the clinicians well received the CDSS implemented on a handheld very well. Their comments and the relatively high enrollment of eligible patients in the trial confirmed that the system's design fits the ED workflow. Following this positive experience, a multi-center trial of MET with full functionality is planned in the future.

Acknowledgments

The authors acknowledge the financial support of the Natural Sciences and Engineering Research Council of Canada and the Physicians' Services Incorporated Foundation of Ontario. The authors would like to thank Marta Kersten for work on designing the user interface, and Bernard Plouffe for work on implementing the MET server.

The authors would like to thank anonymous reviewers for helpful comments and suggestions, and Anne Burgess for editorial assistance.

The MET system was created using AppForge Crossfire and iAnywhere SQL Anywhere Studio.

References

1. Musen MA, Shahar Y, Shortliffe EH. Clinical Decision Support Systems. In: Shortliffe EH, Perreault LE, Wiederhold G, Fagan LM, editors. Medical Informatics. Computer Applications in Health Care and Biomedicine. 2 ed: Springer-Verlag; 2001. p. 573-609.

2. Shortliffe EH. The Adolescence of AI in Medicine: Will the Field Come of Age in the 90's? Artificial Intelligence in Medicine 1993;5: 93-106.

3. Séroussi B, Bouaud J, Antoine ÉC. OncoDoc: a successful experiment of computer-supported guideline development and implementation in the treatment of breast cancer. Artificial Intelligence in Medicine 2001;22: 43-64.

4. Hanson CW, 3rd, Marshall BE. Artificial intelligence applications in the intensive care unit. Critical Care Medicine 2001;29 (2): 427-35.

5. Anderson JG. Clearing the way for physicians' use of clinical information systems. Communications of the ACM 1997;40 (8): 83-90.

6. de Dombal FT, Leaper DJ, Staniland JR, McCann AP, Horrocks JC. Computer-aided diagnosis of acute abdominal pain. British Medical Journal 1972;2 (5804): 9-13.

7. Hunt DL, Haynes RB, Hanna SE, Smith K. Effects of computer-based clinical decision support systems on physician performance and patient outcomes: a systematic review. JAMA: Journal of American Medical Association 1998;280 (15): 1339-46.

8. Kinney WC. Web-based clinical decision support system for triage of vestibular patients. Otolaryngology and Head and Neck Surgery 2003;128 (1): 48-53.

9. Gertner AS, Webber BL. TraumaTIQ: Online decision support for trauma management. IEEE Intelligent Systems and Their Applications 1998;13 (1): 32-39.

10. Daou Systems Inc., Going Mobile: from eHealth to mHealth. A Daou Systems White Paper, Avilable at http://www.daou.com/emerging/pdf/mHealth_White_Paper_April_2001.PDF, 2001 (accessed on March 25th, 2004).

11. Tachakra S, Wang XH, Istepanian RS, Song YH. Mobile e-health: the unwired evolution of telemedicine. Telemed J E Health 2003;9 (3): 247-57.

12. Ammenwerth E, Buchauer A, Bludau B, Haux R. Mobile information and communication tools in the hospital. International Journal of Medical Informatics 2000;57 (1): 21-40.

13. Fischer S, Stewart TE, Mehta S, Wax R, Lapinsky SE. Handheld computing in medicine. JAMIA: Journal of the American Medical Informatics Association 2003;10 (2): 139-49.

14. Roth AC, Leon MA, Milner SM, Herting RL, Jr., Hahn AW. A personal digital assistant for determination of fluid needs for burn patients. Biomedical Sciences Instrumentation 1997;34: 186-90.

15. Zupan B, Porenta A, Vidmar G, Aoki N, Bratko I, Beck JR. Decisions at hand: a decision support system on handhelds. Medinfo 2001;10: 566-570.

16. Michalowski W, Rubin S, Słowiński R, Wilk S. Mobile clinical support system for pediatric emergencies. Journal of Decision Support Systems 2003;36 (2): 161-176.

17. Ramnarayan P, Britto J. Paediatric clinical decision support systems. Archives of Disease in Childhood 2002;87 (5): 361-2.

18. Brown DS, Motte S. Device Design Methodology for Trauma Applications. In: SIGCHI conference on Human factors in computing systems; Los Angeles; 1998. p. 590-594.

19. Shiffman RN, Karras BT. Pen-Based, Mobile Decision Support in Healthcare. ACM SIGBIO Newsletter 1999;19 (2): 5-7.

20. Essex D. Wow! Wireless works!! Healthcare Informatics 1999;16 (10): 38-42, 44, 46.

21. Jing J, Helal AS, Elmagarmid A. Client-server computing in mobile environments. ACM Computing Surveys 1999;31 (2): 117-157.

22. Rae F. The Ottawa ankle rule. Journal of Emergency Medicine 2001;18 (2): 147.

23. Michalowski W, Rubin S, Słowiński R, Wilk S. Triage of Acute Abdominal Pain: Clinical Use of a Palm Handheld in a Pediatric Emergency Department. In: The 37th Annual Hawaii International Conference on System Sciences (HICSS'04); Los Alamitos: IEEE; 2004. p. 60161a.

24. Berner ES, Ball MJ, Hannah KJ, editors. Clinical decision support systems: theory and practice. New York: Springer Verlag; 1999.

25. Musen MA. Domain ontologies in software engineering: Use of Protege with the EON architecture. Methods of Information in Medicine 1998;37 (4-5): 540-550.

26. Musen MA. Scalable software architectures for decision support. Methods of Information in Medicine 1999;38 (4-5): 229-238.

27. Steel L. Second generation expert systems. Future Generation Computer Systems 1985;1 (4): 213-221.

28. Puppe B, Ohmann C, Goos K, Puppe F, Mootz O. Evaluating four diagnostic methods with acute abdominal pain cases. Methods of Information in Medicine 1995;34 (4): 361-8.

29. Pawlak Z, Słowiński R. Rough Set Approach to Multi-attribute Decision Analysis. European Journal of Operational Research 1994;72: 443-459.

30. Słowiński R. Rough Sets Approach to Decision Analysis. AI Expert Magazine 1995;10: 8-25.

31. Stefanowski J. On rough set based approaches to induction of decision rules. In: Skowron A, Polkowski L, editors. Rough Sets in Data Mining and Knowledge Discovery. Heidelberg: Physica-Verlag; 1998. p. 500-529.

32. Stefanowski J. Rough set based rule induction techniques for classification problems. In: Proceedings of the 6th European Congress on Intelligent Techniques and Soft Computing (EUFIT'98). Aachen; 1998. p. 109-113.

33. Greco S, Matarazzo B, Słowiński R. Dealing with missing data in rough set analysis of multi-attribute and multi-criteria decision problems. In: Zanakis SH, Doukidis G, Zopounidis C, editors. Decision Making: Recent Developments and Worldwide Applications: Kluwer Academic Publishers; 2000. p. 295-316.

34. Stefanowski J. Classification support based on the rough sets. Foundations of Computing and Decision Sciences 1993;18 (3-4): 371-380.

35. Wilk S, Słowiński R, Michałowski W, Greco S. Supporting Triage of Children with Abdominal Pain in the Emergency Room. European Journal of Operational Research 2004 (in press).

36. Eriksson H, Puerta AR, Musen MA. Generation of Knowledge-Acquisition Tools from Domain Ontologies. International Journal of Human-Computer Studies 1994;41 (3): 425-453.

37. Johnson J. Selectors: Going beyond user-interface widgets. In: Proceedings of the ACM Conference on Human Factors in Computing Systems (CHI '92). New York: ACM; 1992. p. 273-279.

38. Gennari JH, Musen MA, Fergerson RW, Grosso WE, Crubezy M, Eriksson H, et al. The evolution of Protégé: An environment for knowledge-based systems development. International Journal of Human-Computer Studies 2003;58 (1): 89-123.

39. Michalowski W, Kersten M, Wilk S, Słowiński R. Designing Man-Machine Interactions for Mobile Clinical Systems: MET Triage Support Using Palm Handhelds. European Journal of Operation Research 2004 (in press).

Published in Methods of Information in Medicine, vol. 44, no. 1, 2005, 14-24

40. Rector AL, Horan B, Fitter M, Kay S, Newton PD, Nowlan WA, et al. User Centered Development of a General Practice Medical Workstation: The PEN&PAD Experience. In: Proceedings of the SIGCHI conference on Human factors in computing systems. Monterey; 1992. p. 447-453.

41. Raskin J. The Human Interface. New York: Addison Wesley; 2000.

42. Sittig DF, Kuperman GJ, Fiskio J. Evaluating physician satisfaction regarding user interactions with an electronic medical record system. JAMIA: Journal of the American Medical Informatics Association 1999: 400-404.

43. Ostrem J, Palm OS ® User Interface Guidelines, Avilable at http://www.palmos.com/dev/support/docs/uiguidelines.pdf, 2003 (accessed on June 15th, 2004).

44. Korner H, Sondenaa K, Soreide JA, Andersen E, Nysted A, Lende TH. Structured data collection improves the diagnosis of acute appendicitis. British Journal of Surgery 1998;85 (3): 341-4.

45. Guerlain S, LeBeau K, Thompson M, Donnelly C, McClelland H, Syverud S, et al. The Effect of a Standardized Data Collection Form on the Examination and Diagnosis of Patients with Abdominal Pain. In: Proceedings of the Human Factors and Ergonomics Society 45th Annual Meeting. Minneapolis: Human Factors and Ergonomics Society; 2001. p. 1284-1289.

46. de Dombal FT. Computer-aided decision support in clinical medicine. International Journal of Bio-Medical Computing 1989;24 (1): 9-16.

47. Satyanarayanan M. Fundamental Challenges of Mobile Computing. In: Proceedings of the 15th ACM Symposium on Principles of Distributed Computing: ACM Press; 1995. p. 1-7.

48. Quinn J. An HL7 (Health Level Seven) overview. Journal of AHIMA / American Health Information Management Association 1999;70 (7): 32-34.

49. SNOMED International, SNOMED Clinical Terms, Avilable at http://www.snomed.org/snomedct, 2004 (accessed on June 16th, 2004).

50. WHO Statistical Information System, IDC-10. The International Statistical Classification of Diseases and Related Health Problems, tenth revision, Avilable at http://www.who.int/whosis/icd10/, 2003 (accessed on August 4th, 2004).

51. Regenstrief Institute Inc., Logical Observation Identifiers Names and Codes, Avilable at http://www.loinc.org/, 2004 (accessed on August 4th, 2004).

52. IETF: The Internet Engineering Task Force, TSL: Transport Layer Security, Avilable at http://www.ietf.org/html.charters/tls-charter.html, 2004 (accessed on June 17th, 2004).

53. Sutton GC. Computer-aided diagnosis: a review. British Journal of Surgery 1989;76 (1): 82-5.

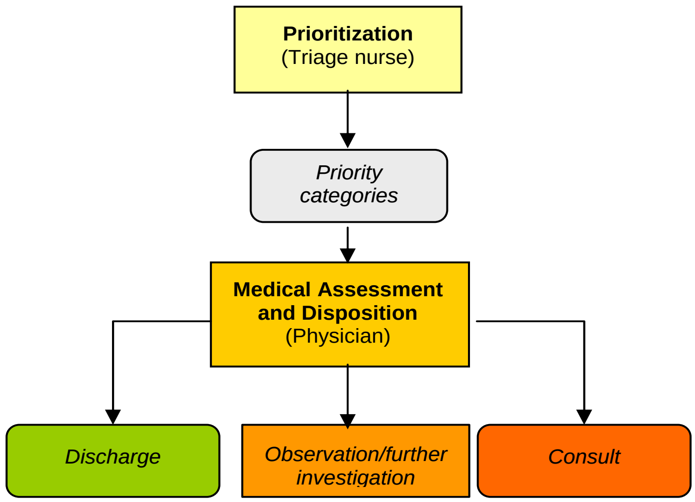
Published in Methods of Information in Medicine, vol. 44, no. 1, 2005, 14-24

Table 1. Interface specification for the presentation of abdominal pain

Attribute	Form	Selector
Site of pain	Symptoms	Pictogram
Duration of pain	Symptoms	Numeric
Type of pain	Symptoms	List
Shifting of pain	Symptoms	List
Previous visit	Symptoms	List
Vomiting	Symptoms	List
Site of tenderness	Signs	Pictogram
Rebound tenderness	Signs	List
Localized guarding	Signs	List
Temperature	Signs	List
WBC	Tests	Numeric

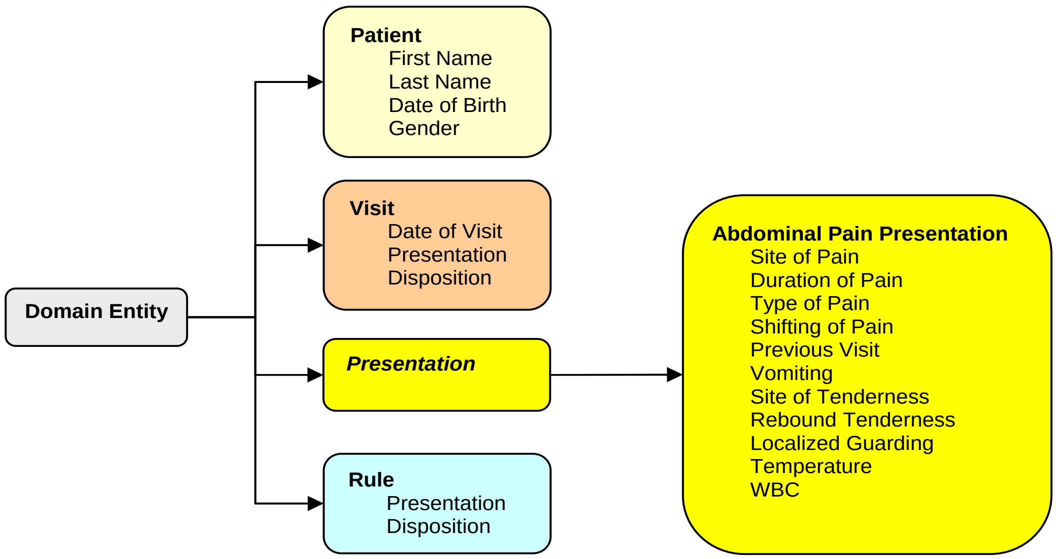
Published in Methods of Information in Medicine, vol. 44, no. 1, 2005, 14-24

Fig. 1. The ED workflow for triage of acute pain presentation



Published in Methods of Information in Medicine, vol. 44, no. 1, 2005, 14-24

Fig 2. Domain ontology for triage of abdominal pain presentation



Published in Methods of Information in Medicine, vol. 44, no. 1, 2005, 14-24

Fig. 3. MET interface ontology

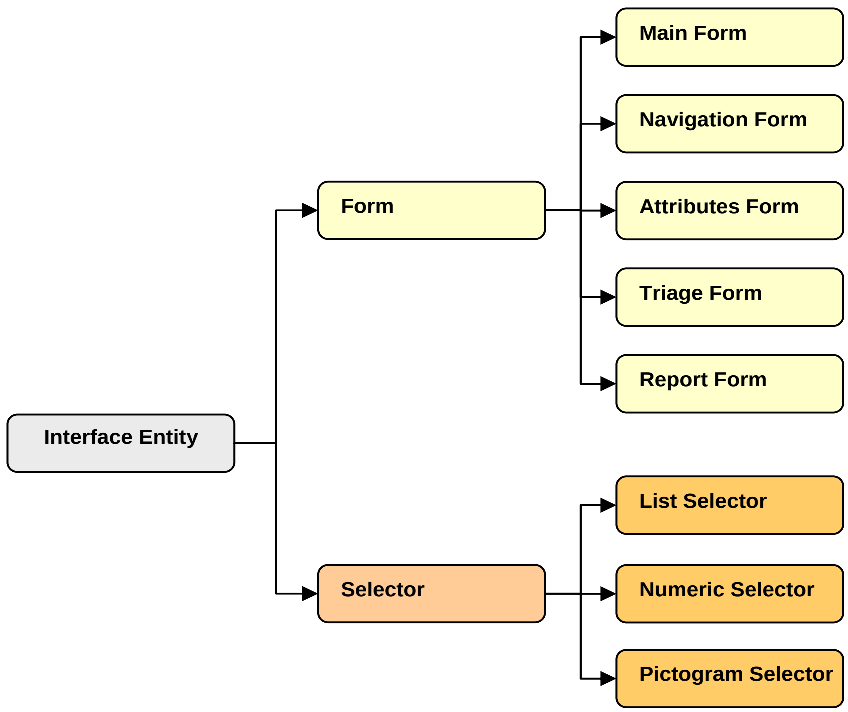
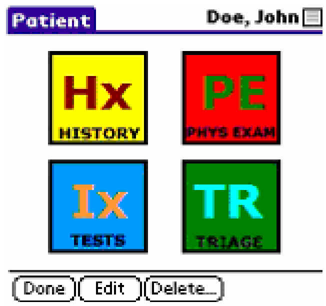
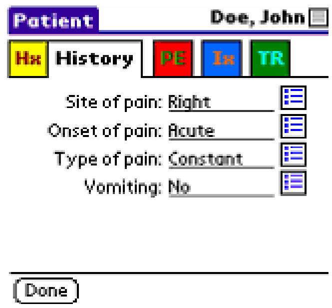


Fig. 4. Navigation form



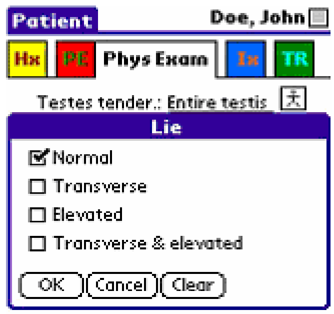
Published in Methods of Information in Medicine, vol. 44, no. 1, 2005, 14-24

Fig. 5. Attributes form



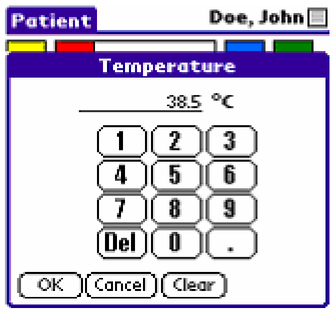
Published in Methods of Information in Medicine, vol. 44, no. 1, 2005, 14-24

Fig. 6. List selector



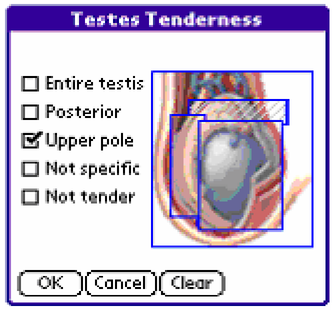
Published in Methods of Information in Medicine, vol. 44, no. 1, 2005, 14-24

Fig. 7. Numeric value selector



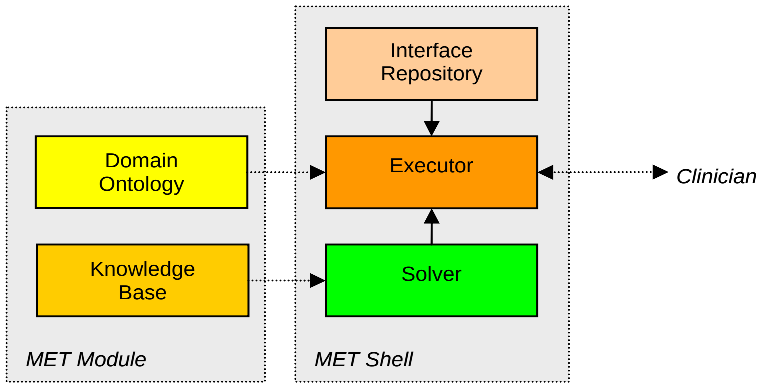
Published in Methods of Information in Medicine, vol. 44, no. 1, 2005, 14-24

Fig. 8. Pictogram selector



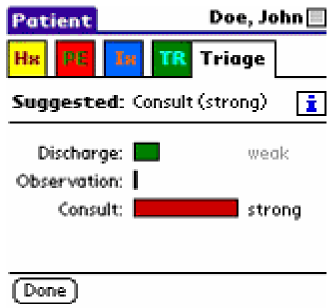
Published in Methods of Information in Medicine, vol. 44, no. 1, 2005, 14-24

Fig. 9. Architecture of the triage support component of the MET system



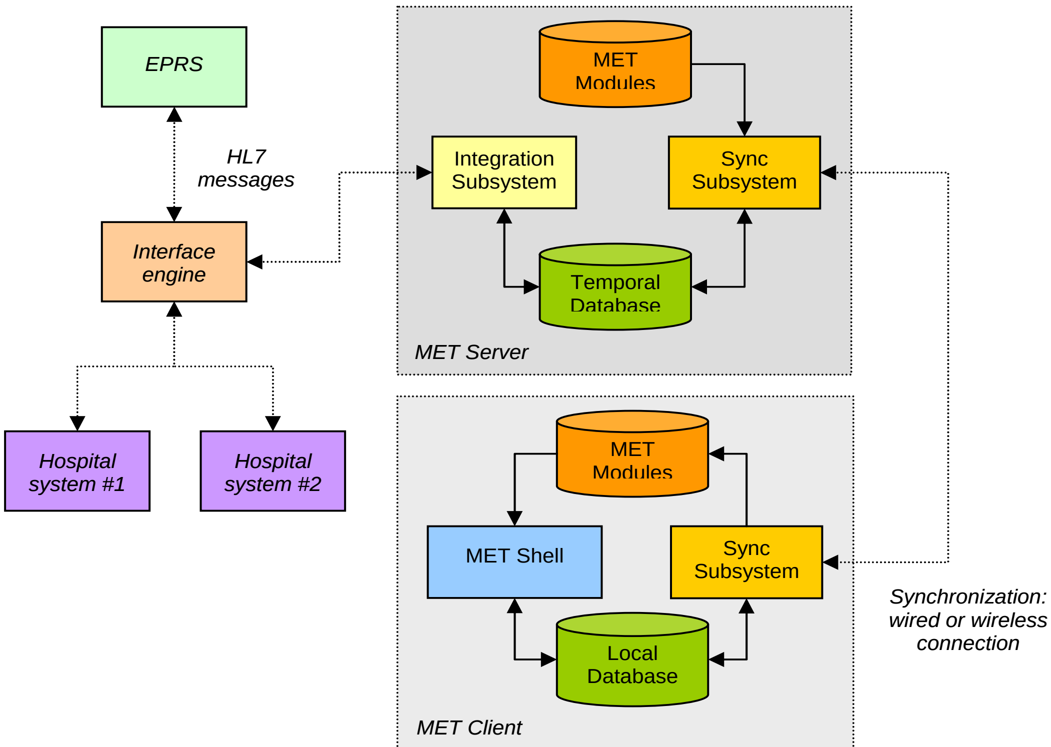
Published in Methods of Information in Medicine, vol. 44, no. 1, 2005, 14-24

Fig. 10. Triage recommendation form



Published in Methods of Information in Medicine, vol. 44, no. 1, 2005, 14-24

Fig. 11. Extended client-server architecture of the MET system



Published in Methods of Information in Medicine, vol. 44, no. 1, 2005, 14-24

Fig. 12. Design of the clinical trial