J. of Ultra Comp. and I.T. Vol.1(1), 23-34 (2010).
www.compitjournal.org

ISSN 2229-3531 (Print)
2455-9997 (Online)

EHR systems to support patient care using DSS

KULVINDER SINGH MANN[1] and HARSH SADAWARTI[2]

[1]Associate Professor GNDEC, LDH.
[2]Principal RIMT, IET, Mandi Gobindgarh (INDIA)

(Acceptance Date 22nd March, 2010)

Abstract

This paper presents a guideline engineering process resulting in a number of artefacts which lead to successful decision making in healthcare delivery. EHR Architecture is described in terms of reference model and archetype model describing the concepts of transaction, folder, data structures and data types. Existence of the components of Computer Interpretable clinical Guidelines (CiGs) in the EHR allows the point-of-care application to better promote the guideline with workflow and decision support. EHR also supports workflows by identifying the process of carrying out the actions. Instruction reference model is explained with a purpose to provide detailed information about instructions, the states through which instruction execution proceeds and how instruction connectors helps to specify workflow patterns.

Key words : Electronic Health Records(EHR), artefacts, Computer Interpretable clinical Guidelines(CiGs), Workflows, reference model, archetype model, Instruction Reference Model(IRM).

Introduction

Undertaking a **guideline engineering process**, we can develop a set of artefacts (specifications) from a source guideline document. Guideline engineering process is shown in Figure 1.

The steps followed to perform guideline engineering process are:

Analysis: Process begins by an analysis of clinical guideline document and abstracts from it the required set of actors, interaction between them, actions to be performed, decision points and data or information flows to develop scenarios. A **scenario** is an account or synopsis of a possible course of action or events and provides a description of interactions between identifiable entities, actors or participants. It is a type of case study or a use case. Scenarios are "walkthroughs" of the use case using real world data. Each scenario should document the :

a) Business processes

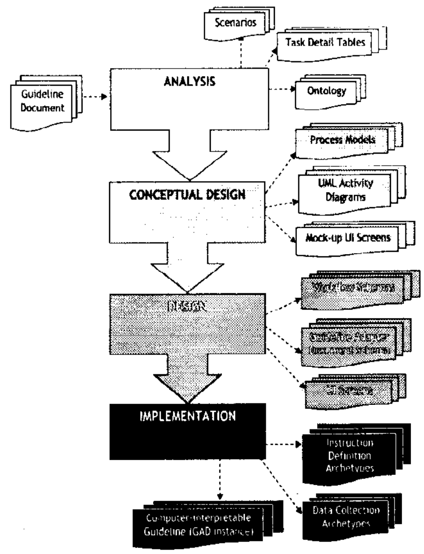


Fig. 1. Guideline Engineering Process and Associated Key Deliverables

b) Business and technology environment
c) Set of people and system components
d) Mapping of actors to activities
e) Mapping of environment to activities
f) Data flow throughout the scenario between actors.

Conceptual design: After developing scenarios conceptual design begins by producing process models represented as data flow diagrams. In this phase Mock-up interface screens are also developed to simulate the interactions between providers and clinical information systems.UML activity diagrams are also produced to illustrate the system component interactions when DSS or WfMS(workflow modeling using scenarios) are used.

Design : In this Phase, UML sequence diagrams are produced to formalize the interactions in guideline usage and workflow scenarios. Required workflow schemas are also produced from actions recommended in the guideline. The workflow schemas provide formal specifications of the process models developed during conceptual design. A computer interpretable specification of the guideline is also developed based on ontology created in analysis phase where ontology is an explicit specification of a conceptualization. The last step is to create the finalized version of the mock-up user interface screens which may undergo several iterations and actual usability testing.

Implementation : It is the final phase in which we produce the corresponding instruction definition archetypes from the workflow schemas and other EHR archetypes needed for data collection as a result of performing the tasks in the workflow.

Current guideline models use different modeling formalisms to express the different processes:

a) Flowcharts for algorithmic problem-solving processes.

b) Disease-state maps relating decisions made during patient care.

c) Sequencing of activities in care plans.

Engineering of a guideline in electronically decision supported clinical information systems produces a number of artifacts: a) EHR content (what is to be recorded) b) CiGs, computer interpretable clinical guidelines (when to record and how to make decisions) c) Workflow schemas (clinician and system dependent actions) d) Hypermedia (human readable electronic version of a guideline) as shown in Figure 1. A CiG may reside within a workflow expression of a larger-scale workflow whose activity executes CiG. E.g. a workflow activity of ordering a medication executes a CiG to recommend the specifics of the drug (name, dose ,quantity etc.) or referring the patient to a specialist. A clear relationship among these artefacts leads to successful computerized support in evidence-based CDM.

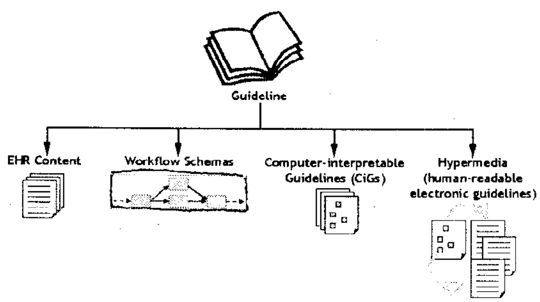


Fig. 1. Guideline Engineering Artefacts

Role of CiGs and Workflows in EHR :

CiGs specify decision support rules and recommend actions to be taken for patient care. They reside within knowledge bases and are executed by decision support systems or applications generating recommendations in the form of alerts and reminders.

Workflow is modeled in terms of a single clinician's decision-making process along with the action.

In healthcare, two aspects are supported

1) Explicit recording of rationale (process of arriving at clinical decisions).
2) How to carry out the actions.

This second aspect constitutes the workflow. The timely flow of information is critical in workflows between organizations. It is often required that clinical observations are recorded at granular level at specific points in the workflow. At higher granularity level, the control flow within workflows is dictated by recording of documents rather than recording of data items within documents. e.g. action of ordering a wheelchair to a patient cannot be performed without receiving the provider's documentation of assessment and recommendation. It is required to explicitly specify what to record at specific points in a workflow, who will record and enact the activities and to have explicit constructs in the EHR that retains detailed information about the actual steps to be taken. The OpenEHR entries have a direct relationship to components of CiGs as shown in Figure 2.

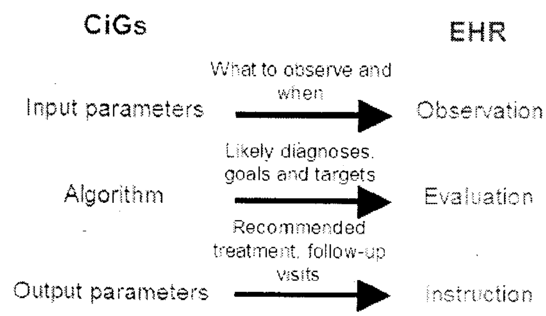


Fig. 2. Relationship of the CiG to EHR content

EHR Architecture :

OpenEHR architecture forms the basis of EHR approach having two-level modeling framework: reference model and archetype model.

OpenEHR reference model: It has a number of concepts that it represents with regards to the EHR:

a) Transaction, a durable, atomic, consistent, isolated, modifiable unit of information corresponding to the interaction of a healthcare agent with EHR. Transactions are either **persistent** containing data which remains pertinent over a long period of time such as family history or **event** containing data at a particular instant of time such as biochemistry test.

b) Folder, organizing a set of related transactions (event, persistent or demographic) as they accumulate over time helping in logically navigating the contents of EHR.

c) Organizer, provides headings to sections within the transaction content and is archetyped as a tree structure.

d) Entry, contains all information occurring in a clinical statement. It may be an **observation** (measurable or subjective statements as BP), **evaluation** (results of analysis of observations as diagnosis), or **instruction** (actions to be carried out as medication order).

e) Data structure, captures historical data. It is in the form of trees, lists or tables. It may be single having only one data item (e.g. patient's weight) or composite having a group of data items (e.g. systolic and diastolic BP values).

f) Data type, represents the type of EHR data. These are derived from other data types used in GEHR, HL7 v3 RIM. Some OpenEHR data types are : BASIC, DV, DV_URI, DV_TEXT,DV_QUANTITY,TIME_SPECIFICATION, ENCAPSULATED, DV_EHR_URI.

OpenEHR archetype model: Archetypes are formal structured constraint definitions of clinical concepts which allow guideline-specific and case-specific information to be recorded in a general and extensible EHR framework. An archetype defines a collection of concepts which when aggregated from a higher level concept, thus defined at different degrees of granularity. E.g. BP measurement is an archetype at higher level and consists of other concepts such as systolic and diastolic blood pressure, the position of the patient during measurement, patient's cuff size which is aggregated to give BP value.

Archetypes can be specialized for local use. A specialized archetype conforms to and contains all the relevant parts of its parent archetype. It includes additions and modifications for specialized use. For Example: 'Problem' (consisting of text, clinical description) is specialized to 'Diagnosis'(consisting of terms, grading, diagnostic criteria) which is further specialized to 'Diabetes Diagnosis' (local specialization containing diagnostic criteria constrained to fasting > 6.1, GTT2hr > 11.1) where 'Problem' is the parent of 'Diagnosis' and 'Diagnosis' is the parent of 'Diabetes Diagnosis'.

The identifier of the specialized archetype is usually equal to its parent's archetype identifier, with a 'dot' notation to delimit a further name to refer to the specialized archetype. In the case of our example, the identifiers would be as in Figure 3.

openehr.evaluatoin.problem.v1

↓

openehr.evaluatoin.problem.diagnosis.v1

↓

openehr.evaluatoin.problem.diagnosis.diabetes.v1

Fig. 3. Example of Archetype Specialization

Thus, archetypes represent the concepts used across healthcare domain in a standardized format that maintains their semantics such that they can be shared between information systems and provide interoperability, extensibility and flexibility in EHR systems via archetype specialization. Whilst archetypes place constraints on the reference model to define domain-specific concepts, openEHR **templates** describe the high-level constraints on the actual data that can be collected by specifying the set of archetypes that can be used and other local system constraints such as how archetypes are organized to form EHR document, the ordering, cardinality, obligation and data item constraints such as default values.

OpenEHR Paths: In openEHR data structure, any node or leaf item can be reached via a path mechanism which is generated by the concatenation of the values of two attributes, 'name' and 'meaning' of each element from a particular point to the specific node to be referred. These attributes are inherited from LOCATABLE class of OpenEHR model. The path that uses the 'name' values is called **runtime path** which is unique in data and used to locate data items of EHR. The path that uses 'meaning' values is called **archetype path** which is unique within archetypes but not in EHR. These are used to match sub compositions to their generating archetype structures or to assist Graphical User Interface (GUI) display. The values for 'name' and 'meaning' may or may not be same.

Other OpenEHR components:

1) EHR Extract Model, which represents an extract of all of some version of EHR which helps to transmit selected parts of EHRs between multiple EHR systems.

2) Demographics Model for capturing aggregated EHR data for secondary purposes.

3) Common Model, which describes archtyping features for the openEHR models and external identifiers.

4) Support Concepts for the OpenEHR reference models.

Rationale Construct : There exists a problem to link the decisions made by the healthcare providers back to the guidelines. But it can be solved using rationale links which track the relationship of a series of clinical encounters to a guideline decision rule representation. Using *open*EHR's archetypes, we refine the detailed information recording options for specific classes of encounter. The rationale construct allows the clinician or the electronic DSS to record justification for decision points made during the patient's care and includes a) an optional free-text justification statement from the clinician b) identifier for the guideline used c) set of indications or links to justify the decision chosen.

It can reside within any entry (observation, evaluation or instruction) to explicitly record and link back to specific steps in the CiG which request input parameters, arrive at decisions and indicate output parameters.

Figure 4 shows an example of a rationale for an instruction within a GP encounter or contact note, where the instruction is to prescribe an ACE inhibitor due to the presence of proteinuria and diabetes type 1.

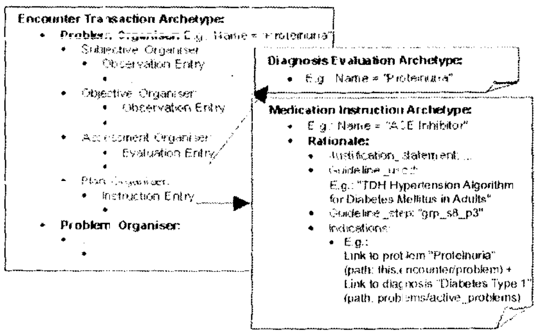


Fig. 4. Example of a Rationale for an instruction within an Encounter

Instruction Reference model (IRM)

OpenEHR Instruction Reference Model supports recording of relatively simple instructions which are the statements describing the actions to be enacted and are detailed enough to be enactable without further details. Instructions are either atomic (one activity) or composite (two or more activities). Activities are linked via connectors. These connectors represent valid types of ordering and execution of activities within an instruction. The various connectors used in IRM to link various activities are sequence, split, conditional_loop and choice_joins which imply pre and post conditions for workflows. e.g., for a sequence connector, an activity will be started only if preceding activity has completed. OpenEHR specifies only the future action to be performed but there is no explicit way to specify what data is to be recorded for the action or how the resulting recorded data is linked back to the instruction that initiated it. This all is done by the archetype author and the recording of links is specified by the EHR system itself. Data is recorded as a result of an instruction activity being executed and the rationale for that data includes a link to the instruction that initiated it but still it is needed to know exactly where to look for that EHR data in the first place and where it is recorded. The execution entry allows this and enables the relevant instruction related data to be grouped and reside in a particular place for a particular care process and activity. Figure 4 shows the relationship among guidelines, archetypes and instructions.

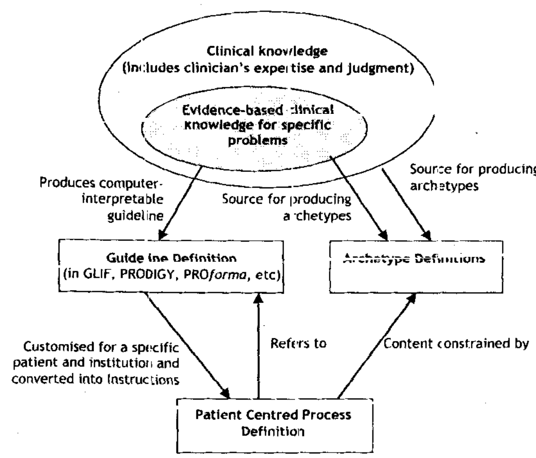


Fig. 4. Relationship between Guidelines, Archetypes and Instructions.

Role of persistent and event transactions in IRM: Persistent transactions containing historical data carry out "Instruction Executions" and Event transactions containing new data carry out "Instruction definitions".

The scope of event and persistent transactions allows to differentiate between past, current and future work and what recording to be done, for whom, by whom and when to be done can be determined. **Instruction Definition entry** acts as process definition to indicate that some activity is to be undertaken and has some attributes such as data, protocol and rationale to capture notes about an instruction. It resides as event transaction and is referred to by a single instruction execution entry within a persistent transaction. An 'activity' requires one or more work items (the representation of the work to be processed within a process instance). An activity may be structured (contains composite activities) and some activities (proxy activities) may refer to other instructions or they may be atomic. The work item of activities describes the work to be done once an atomic activity instance has been assigned to a party or role. **Instruction Execution entry** acts as process instance to record the state of each activity as activity instance identified in instruction definition. It resides in EHR as persistent transaction and is referred to by a single instruction definition entry within an event transaction that initiated it. Figure 5 shows the sub-classing and scope of composite instructions and various questions addressed by subclasses.

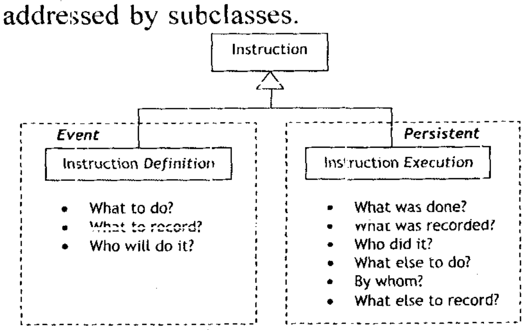


Fig. 5. Sub-Classing and scope of composite instructions.

This model supports simple instructions. e.g., "take this medicine 3 times per day". It has the notion of "do this action". It records the acts that have taken place as a kind of observation entry. It presents and uses OpenEHR Action Specification which adds the notion of Action to consider the idea of Acts and Participants and includes the following:

a) Profile: configuration of data mappings from archetyped model of action.

b) Action: description of acts to be performed.

c) Data: execution data as a result of executed actions.

d) State: current state of action according to state machine model.

e) Guideline Identifier: identifier of guideline which initiated action.

Modeling Instructions: In healthcare delivery, modeling of instructions requires an understanding of the types of semantics that can be placed on the concept of instructions themselves. Thus, there are two semantics of action: a) system executed actions such as alerts and reminders generated by Clinical DSS and b) real world actions performed by humans such as a nurse administering a drug to a patient. Instructions result from evaluations or clinical decisions being made. Thus, decisions and actions have direct relationship.

Modeling Workflows: Action specifications do not allow complex linkage of actions to support more complex type of instructions that model workflow. Modeling business processes in healthcare does not take into account conditional branching and parallel execution of activities supported by workflows. Workflows are modeled in terms of composite instructions (WfMC) which have explicit constructs for various types of branching and temporal sequencing of activities in a workflow allowing close correspondence between workflow and EHR, integrating EHR with WfMS. EHR itself contains clinically oriented workflows which need to be coordinated with other workflows.

Workflow is considered at three levels:

1) Generalized process definition: It consists of a network of activities, their relationships, criteria to indicate the start and end of a process and information about the individual activities such as participants, data etc. It is a generic template for a whole class of process definitions and stem directly from clinical guidelines.

2) Patient centered process definition: It is a tailored process definition for a specific patient. It is captured as an instruction definition entry which is stored in a specific patient's EHR. For e.g. a chained medication order for a patient as shown in Figure 5.

3) Process Instance: It is the representation of a single enactment of a process and is created, managed and terminated by workflow management system. For e.g., a surgery performed for a specific patient. It uses its own process instance data and capable of independent control and audit as it progresses towards completion or termination.

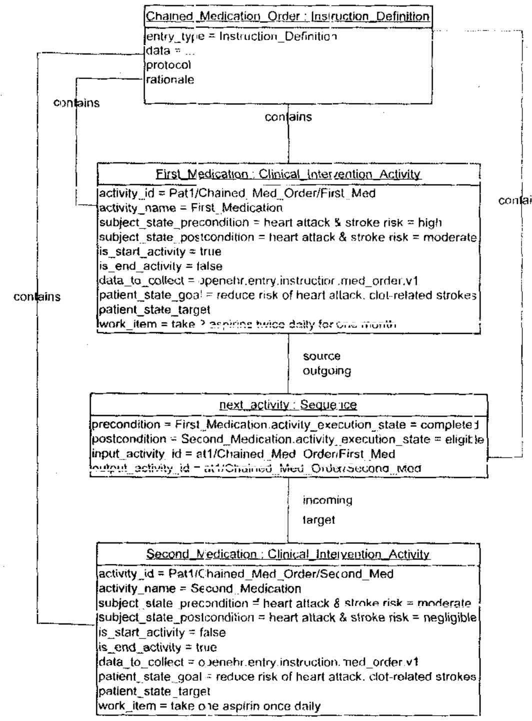
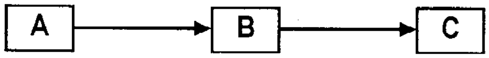


Fig. 6. Chained Medication Order in an Instruction definition

IRM Data Types: The data types used by IRM are COORDINATED_TERM, DV_EXPRESSION:DV_CHOICE_JOIN_TYPE, DV_SPLIT_TYPE, DV_STATE, DV_EXECUTION_STATE, DV_EXECUTION_STATE_CHANGE.

Equivalence of IRM connectors and workflow patterns: Few of the workflow patterns such as Multi-Merge, the Discriminator and the N-out-of-M-join are not explicitly representable in the WfMC's XML Process Definition Language (XPDL).Therefore, they are mapped from XPDL to Instruction models. e.g.

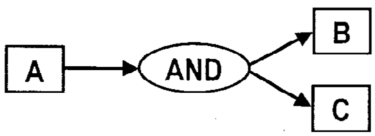
Sequence



The enabling of one activity after the completion of another activity in the same process.

IRM Connector Equivalent: Sequence connector.

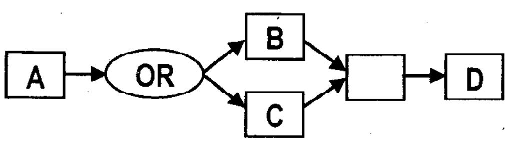
Parallel Split



A point in the process where a single thread of control splits into multiple threads of control and can be executed in parallel (at the same time or in any order).

IRM Connector Equivalent: Split connector with attribute split_type = AND_split.

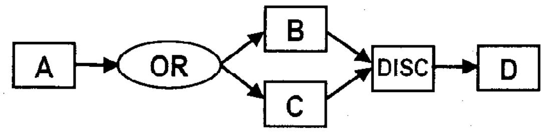
Multi-Merge



A point in the process where one or more threads reconverge without synchronization. That is, if more than one branch is executed then they are executed in parallel, but are then reconverged asynchronously such that activity "D" is started for every incoming branch.

IRM Connector Equivalent: Split connector with attribute split_type = OR_split, and Choice_Join connector with attribute choice_join_type = OR_join, and is_synchronous = false.

Discriminator



A point in the process where the process waits for one out of one or more threads to complete before starting the subsequent activity. It then waits for all remaining threads to complete and 'ignores' them. Once all incoming branches have been triggered, it resets itself so that it can be triggered again.

IRM Connector Equivalent: Split connector with attribute split_type = OR_split, and Choice_Join connector with attribute choice_join_type = XOR_join, and is_synchronous = true.

Setting synchronization to 'true' ensures that the subsequent activity can be started when it receives an incoming branch that was triggered, but does is not considered 'complete' until all branches that were triggered have completed.

State Machine Model for Instruction Execution :

Instructions specify actions occurring in the real world. Thus, they include conditions for starting, delaying, repeating, stopping or cancelling which are all influenced by real world events. Instructions and activities within instructions progress through meaningful execution **states** (Figure 7) due to occurrence of events.

a) Ineligible: service object is disabled and cannot be started.

b) Eligible: service object is waiting to be executed.

c) Executing: service object is running.

d) Completed: execution of service object is completed.

e) Suspended: execution of service object is temporarily suspended.

f) Aborted: executing service object is terminated due to an exception.

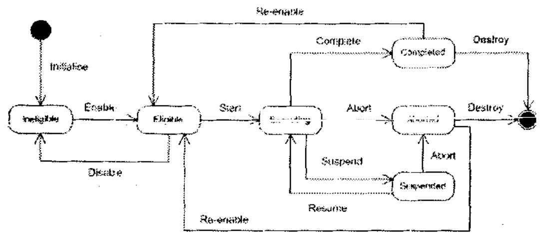
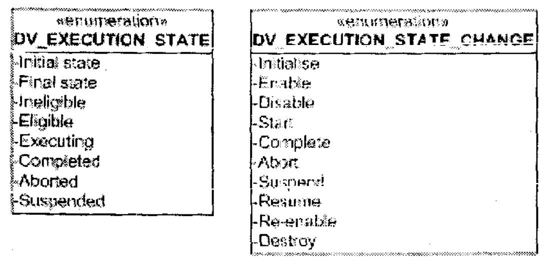


Fig. 7. State Machine Model for Instruction Execution

State machine model may be too granular to be executed by the system itself for some instances of instruction itself which are largely executed in the real world. The progression of such states largely depends on the incoming events.

Conclusion

From this paper, we have concluded that Guideline Engineering process produces various artefacts which help in patient care. From the perspective of using the DSS, a computer interpretable guideline provides the logic/reasoning, recommendations, hypermedia information of the clinical guideline being considered. OpenEHR reference model is introduced with respect to the EHR, data structures and data types as they provide the building blocks of workflow-integrated EHR. OpenEHR reference model is constrained by an archetype model to represent specific information recording



requirements or concepts used across healthcare domain. e.g. to enroll a patient in a "post-stroke rehabilitation program" and describing all the activities of the workflow. Workflow models the work to be carried out, by whom, when and how and information about the instructions that initiate the actions constituting workflow can be collected using rationale links.

References

1. G. H. Brundtland, 'Press-release from the Director-General of the World Health Organisation', 16 May (2003).

2. "World Health Report 2002: Reducing Risks, Promoting Healthy Life," World Health Organization (WHO) (2002).
3. *(WHO)*, World Health Organization (WHO), viewed 30 Jun 2003, <http://www.who.int>.
4. "World Cancer Report," World Health Organization (WHO), Geneva 3 Apr (2003).
5. *Agency for Healthcare* 2003, *Research and Quality*, Agency for Healthcare Research and Quality, viewed 30 Jun 2003, <http://www.ahcpr.gov>.
6. *To Err is Human: Building a Safer Health System*. Institute of Medicine, Washington, DC: National Academy Press, (1999).
7. E. G. Holland and F. V. Degruy, 'Drug-induced Disorders', *Am Fam Physician,* vol. 56, no. 7, 1 Nov, pp. 1781-8, 1791-2 (1997).
8. "PRODIGY Phase Two: Report on the Results of PRODIGY Phase Two," Sowerby Centre for Health Informatics at Newcastle (1998).
9. S. Boyages, Ma, J., Stennett, J. and Penny, R. 2001, 'Chronic and Complex Care Reform for the NSW Health System', *Healthcover Article*, Apr/May (2001).
10. L. Dazzi, C. Fassino, R. Saracco, S. Quaglini and M. Stefanelli, 'A patient workflow management system built on guidelines', *Proc AMIA Annu Fall Symp*, pp. 146-50 (1997).
11. S. Abbott, Johnson, L. and Lewis, H., 'Participation in Arranging Continuing Health Care Packages: Experiences and Aspirations of Service Users', *Journal of Nursing Management,* vol. March, 9(2), (March 2001), pp. 79-85 (2001).
12. R. Summers, Cloke, A. P., Nurse, D. and Kay, J. D. S., "Workflow Models of Hospital Discharge Communications," IEEE International Conference on Information Technology Applications in Biomedicine (1998).
13. T. Kindberg, Bryan-Kinns, N. and Ranjit, M., "Supporting the Shared Care of Diabetic Patients," International ACM SIGGROUP Conference on Supporting Group Work, Arizona (1999).
14. P. Dadam, M. Reichert and K. Kuhn, "Clinical Workflows-The Killer Application for Process-oriented Information Systems?," 4th International Conference on Business Information Systems, Poznan (2000).
15. P. F. Brennan, 'Health informatics and community health: support for patients as collaborators in care', *Methods Inf Med,* vol.38, no. 4-5, Dec, pp. 274-8 (1999).
16. M. Schoop and D. G. Wastell, 'Effective multidisciplinary communication in healthcare: cooperative documentation systems', *Methods Inf Med,* vol. 38, no. 4-5, Dec, pp. 265-73 (1999).
17. *Enhanced Primary Care - Medicare Benefits Items*, updated 24 Apr (2002), Commonwealth Department of Health and Ageing (2002), <http://www.health.gov.au/epc/>.
18. *Assessing the New MBS Items - Group 18 & 19*, Mid North SA Division of General Practice, viewed 28 May (2003), <http://www.mndgp.org.au/Accessing%20new%20MBS%20items.pdf>.
19. J. Grimshaw, N. Freemantle, S. Wallace, I. Russell, B. Hurwitz, I. Watt, A.

Long, and T. Sheldon, 'Developing and implementing clinical practice guidelines', *Qual Health Care,* vol. 4, no. 1, Mar, pp. 55-64 (1995).

20. I. Sim, Gorman, P., Greenes, R., Haynes, R., Kaplan, B., Lehmann, H. and Tang, P., 'Clinical Decision Support Systems for the Practice of Evidence-based Medicine', *Journal of the American Medical Informatics Association,* vol. Nov /Dec, 8(6), pp. 527-34 (2001).

21. D. L. Hunt, Haynes, R.B., Hanna, S.E. and Smith, K., 'Effects of Computer-based Clinical Decision Support Systems on Physician Performance and Patient Outcomes', *Journal of the American Medical Association,* vol. 280, no. 15, 21 Oct, pp. 1339-46 (1998).