A MEDIATOR-BASED APPROACH FOR THE INTEGRATION OF DISTRIBUTED ELECTRONIC HEALTHCARE RECORDS

J. A. Maldonado, P. Crespo, A. Sanchis, M. Robles

Bioengineering, Electronics and Telemedicine Group, Technical University of Valencia, Spain

jamaldo@upvnet.upv.es

Abstract: To support cooperative work among health professionals and institutions it is necessary to share healthcare information about patients in a meaningful way. But, nowadays, in most hospitals health data are distributed across several information systems whose interconnection is difficult to achieve, this leads to the so called islands of information. This paper briefly describes the architecture, design and implementation of the PANGEA mediator system. PANGEA allows healthcare professionals to access patient information stored in heterogeneous autonomous information systems through a set of formal aggregates of health data based on the European pre-standard of Healthcare Record Architecture ENV13606 from CEN/TC251. ENV13606 is also used as canonical model for the representation of healthcare information, therefore the overall system can also be considered as a system for publishing legacy relational data as XML-Electronic Healthcare Records compliant with ENV13606.

Introduction

Integration is still central for health information systems. Most modern hospitals have computerized records. However, these systems are usually proprietary and often only serve one specific department within the hospital. Hospitals may have dozens of individual systems that do not interoperate with each other. This leads to fragmented and heterogeneous data resources and services, which contain health data about patients. The best-of-breed approach can be very suitable for large organizations letting departments meet their custom business needs more easily and allowing a greater flexibility within decentralized organizations such as hospitals. The challenge is finding how these systems can efficiently and meaningfully exchange health information about patients with one another.

Nowadays, the healthcare sector is undergoing a change. One new context in which a team of healthcare professionals from different disciplines and institutions is responsible for patient health is replacing the traditional single doctor-patient relationship. This new context requires a high level of interoperability and data sharing among professionals and institutions involved in the healthcare of a patient. This crucially depends on the ability to exchange information about patients while preserving its original meaning. Briefly, what is required is that everyone involved in the delivery of healthcare to a patient should be able to access all the relevant patient's healthcare information.

Materials and Methods

Mediators. The mediator/wrapper approach [1] is used to integrate data from both traditional database sources and non-database information sources such as structured or semi-structured files, multimedia files or other proprietary system data. In this architecture a mediator, which can be seen as a virtual database, is introduced between the data sources and the application using them, the mediator is capable of answering queries about the underlying data, for this purpose the mediator uses the sources (suitable interfaced by a wrapper), and/or other mediator to answer the queries. The virtual database offered by a mediator must be based on a specific data model, called the canonical or common data model. The schemas of the virtual databases are schemas of this model.

Electronic HealthCare Record Architecture (EHCRA). Basically, an Electronic Healthcare Record Architecture (EHCRA) is an information model or framework for the construction of electronic healthcare records. It models the generic features necessary in any electronic healthcare record for it to be communicable, complete, a useful and effective ethic-legal record of care, and may retain integrity across systems, countries, and time. It neither does it prescribe or dictate what should be stored in a healthcare record, nor how any electronic healthcare record system has to be implemented [2]. Standardization of Electronic Healthcare Record Architecture (EHCRA) is vital if the clinical information has to be transferred outside the organization or department where it was created.

Much work has been done in the field of EHCRA. In Europe, Work group I of CEN/TC251 (European committee of Normalization, Technical Committee 251), has developed a European pre-standard known as prEN13606-1 [2][3].

Most healthcare data continue to be stored in relational databases system. It is not probable that this situation will change in the foreseeable future due to the high reliability and performance of relational databases. Consequently, some sort of semi-automatic mechanism is needed to publish legacy relational data in the form of EHCR extracts compliant with some EHCR architecture standard. PANGEA is based on ENV13606.

Archetypes. The archetypes along with a reference model are the basis for the dual-model approach to the design of the EHR communication information architecture [3][4].

The reference model is used to represent the generic properties of health record entries, how they are aggregated, and the context information required to meet ethical, legal and provenance requirements.

On the other hand, an archetype defines valid configurations of the building-block classes defined in a reference model for particular clinical domains, organizations, and operational context by specifying particular record component names, data types and prescribed value ranges, and values. In our case, this reference model is ENV13606. Two other similar experiences in the use of archetypes to model EHCR can be found in the literature [5][6]. Since the components of ENV13606 have been defined at a high level of abstraction they give us a flexible model able to represent any entry in a healthcare record. Thus, they can be easily used to represent terms or concepts from the medical domain such as a discharge report, GP record, protein S level, etc. Specialist in the domain should define archetypes, for instance pathologists may define archetypes to represent biochemical results but a college of general practitioners might define one for physical examinations.

Archetypes in PANGEA. The integration approach of PANGEA is based on the concept of archetype. The middleware data model used in PANGEA is a labelled directed graph data model (underlying model of XML) where the nodes are basically objects (complex or atomic) and the edges represent different kinds of relationships between the nodes. In this setting an archetype is just a schema based on this model and the clinical data is instance of a database that complies with this schema.

The archetypes constitute the core of our integration solution; their purpose is to make public the information stored in the underlying databases and, at the same time, to hide technical details, location and heterogeneity of the data repositories. They constitute a semantic layer over the underlying databases associating them with domain specific semantics. Therefore, the mediator schema is designed based on the semantics of the domain, rather than on the actual organization and partitioning of data in the external data sources.

Linking archetypes to data. Since the health data resides in the underlying databases, some kind of mapping information relating archetypes to database schemas should be defined. In database theory, views provide a userdefined subset of a large database. Thus, in PANGEA an archetype can be considered as a view that provides sharing and abstraction in interfacing between the underlying data (mainly relational) and constructs in the middleware data model. We also want to use the views for instantiating extracts of EHCR that are compliant with ENV13606, , i.e. XML documents, from the data stored in the underlying databases. To achieve this, and due to the heterogeneity between the relational model and XML, views should provide apart from a set of queries that extract the relevant information, a mapping between the archetype and the underlying database schemas that should allow the future structuring and tagging of the resulting XML document. Such a mapping is not a trivial task, because the two data models differ significantly. Relational data is flat, normalized into many relations, and its schema is often proprietary. By contrast, XML data is nested, unnormalized, and its schema is public [7], for example, the one proposed by ENV13606. Publishing XML data involves joining tables, selecting and projecting the data that needs to be exported, mapping the relational attributes names into XML elements and attribute and finally creating XML hierarchies.

In PANGEA the mapping is done by linking archetype attributes to a set of table fields and constants through a set of attribute mapping functions, external functions defined in Java or tables holding value mappings. Figure 1 shows as an example two simple mapping functions. The left hand side of the functions specifies a path in the archetype definition. Italics are used to denote name of archetypes and non-italics characters are used to denote archetypes attributes. In the first mapping function the value stored in the field "height" of table "measurements" belonging to database "outpatient" multiplied by 100 is assigned to the attribute "Result_numerical_value" from the composed attribute "Numerical_value" of archetype "Height". The second function simply assigns the string "cm" to attribute "Unit_of_Quantity".

Height.Numerical_value_Result.numerical_value
← Outpatient.measurements.height*100
Height.Numerical_value_Result.Unit_of_Quantity ← "cm"

Figure 1. An example of archetype-data repositories mapping

From the set of functions and the implicit structure defined by the archetypes the system is capable of finding a candidate query that will allow the population of the EHCR extract defined by the archetype and structuring and tagging the resulting XML document according to ENV13606 rules. This approach alleviates the work of defining how to populate archetypes since it is easier for the designer to indicate which table field or expression involving several table fields is relevant to a certain archetype attribute, rather than to specify the possible complex query required to extract all the relevant information, which may involve many relations possibly from several databases.

When generating a candidate query for an archetype some issues have been considered:

- We should keep the existing relationships among the data, typically expressed by foreign keys.
- We should not lose information.

The problem is related to that of computing the natural outerjoin of many relations in a way that preserves all possible connections among facts and how to simplify the result according to a set of conditions and filters imposed by both archetypes and users' queries. In relational database theory the full disjunction of a set of relations R, denoted by FD(R), is defined as the maximum information without redundancy that can be obtained from the relations in R [8]. As demonstrated in [9] the full disjunction is unique for a set of relations and generally can be computed as a single stream of full outerjoins. It is possible to simplify the full disjunction if we take into account the condition imposed by user queries, properties of foreign keys and archetypes and that not all the information held by the underlying data repositories is relevant, we are only interested in data directly or indirectly related to a patient. Generally, the resulting query is a single stream of left and inner joins that can by computed by most relational database engines. Note that a basic mapping query is assigned to each archetype in design time. In run time and depending on the query imposed by the user the mapping query is simplified creating different query plans.

The main requirement for publishing relational data as XML documents is the need for a specification that dictates how to perform the conversion from relational data to XML documents. The specification should describe how to structure and tag data from one or more table as a hierarchical XML document. The attribute mapping functions are powerful enough to help in defining this specification by automatically deriving, when possible, the set of relation attributes from the underlying data sources which univocally identifies a single instance of a data aggregation defined by archetype. When this set can not be computed, i.e. lack of primary keys, the archetype designer is responsible for specifying such set. In PANGEA a conversion specification is compiled for each archetype in design time, when the archetype is instanciated for a particular patient the specification is run and the XML document compliant with ENV13606 is generated.

Results

PANGEA is a middleware between the application and databases making the former independent of the data sources, helping them to communicate in a more meaningful and efficient way. It has been developed in Java and Web Services are used for client-System interaction. PANGEA allows the publishing of health data distributed among several departmental information systems as EHCR extracts compliant with ENV13606. The functionality needed to accomplish this is transformation and sub-setting of databases using view definitions, methods to access and merge data from multiple databases and support for abstraction and generalization of underlying data. The basic architecture is illustrated in Figure 1. The main components of the system are the metadata server and the EHR (Electronic Healthcare Record) server.

The metadata server is the module that is in charge of managing the system's data dictionary, for this purpose it makes use of an object-oriented database. The data dictionary contains the archetypes definition, the underlying databases schemas, the archetypes-databases schemas mappings, relational-XML conversion specification, information about the location of patients' social-demographic data, technical metadata about the data sources such as network addresses and query capabilities and user-related metadata, i.e. user profiles. Two visual tools have been developed to assist in the creation and management of metadata: the archetype editor and the schemata manager. The former facilitates the edition of archetypes, it allows the creation of new ones from scratch or the reuse of the existing ones, it validates their correctness, define the mappings with the component databases schemas and it also controls the versioning. The latter allows the retrieval, caching, management and enrichment of the underlying database schemas and define where demographic information is located.

The Electronic Healthcare Record Server (EHR server) is the core of the whole system. It is layered between the client applications and the data repositories. This server retrieves, by request, all the relevant patient information wherever it is located and presents back the information as a XML document, therefore the whole patient's EHR in the organization can be seen as a XML document compliant with ENV13606. Client applications ask for health information about a particular patient o set o patients who agree on a certain condition as one or more instances of any archetype. The EHR server obtains the definition and mapping specifications of the requested archetype from the metadata server. Afterwards, it builds and populates, by interpreting the mapping specifications, the XML documents that contain the healthcare record extract. The EHR server offers a set of web services that can be used by client applications. Basically, a web service is an interface that describes a collection of operations that are network accessible through standardized XML messaging. The protocol designed to manage the XML messaging is called SOAP (Simple Object Access Protocol). This protocol defines a standard structure, encoding rules and associations to transport XML documents through other protocols such as HTTP, SMTP, etc. SOAP allows a high level of interoperation between heterogeneous applications. Thus, it suits perfectly to accomplish the desirable requirement that between organization communications needs to be achieved, ideally using the same technological solution as for intraorganizational communication.

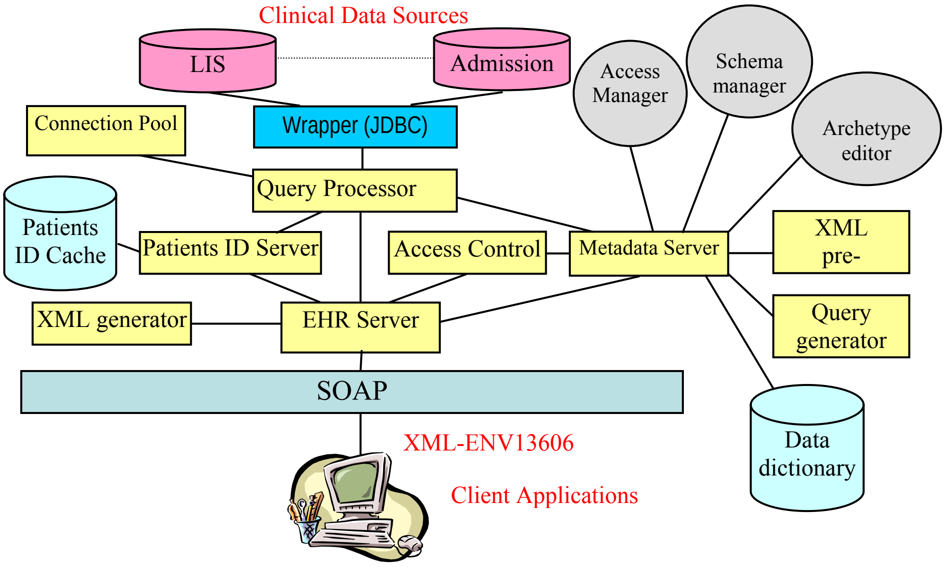

Figure 2. PANGEA architecture

Conclusions

Healthcare is fast becoming more distributed in nature, thus the ability to share health data about patients effectively, meaningfully and securely is the key issue in providing good and cost-effective healthcare. The above outlined system model and architecture define a clinical data access system that provides a single point of entry for providing an integrated virtual view of distributed patient's healthcare records across an institution. The EHCR architecture used is ENV 13606 from CEN/TC 251. Our solution is based on defining a set of formalized aggregates of data (archetypes) with specific semantics and associating them with the heterogeneous structures found in the autonomous information systems. The system is in the line of the virtual approach and read-only view systems, i.e. systems that support read-only views of data held by multiple databases.

Acknowledgements

This work is partially supported by the Ministerio de Sanidad y Consumo of Spain, supporting grants INBIOMED (G03/160) and new model of healthcare services using telemedicine (G03/117) research networks.

References

[1] Wiederhold, G. (1992): Mediators in the Architecture of Future Information Systems. IEEE Computer, 25(3), pp. 38-49.

[2] CEN/TC251 WG I. (1999): Health Informatics-Electronic Healthcare Record Communication-Parts 1, 2, 3 y 4. Final Draft prENV13606.

[3] CEN/TC251. (2003): Health Informatics-Electronic health record communication. Part 1: Reference model. PrEN13606-1.

[4] Beale, T. (2002). Archetypes: Constraint-based domain models for future-proof information systems. Proceedings of the 2002 OOPSLA workshop on behavioural semantics.

[5] Grimson J, William W, Berry D, Stephens, G, Felton E, Kalra D, Toussaint P, and Weier, W.A. (1998): CORBA-based integration of distributed electronic healthcare records using the Synapses approach. Transactions on Information Technology in Biomedicine: 2 (3), pp. 124-138..

[6] http://www.openehr.org/

[7] Shammugasundaram J, Shekita E, Barr R, Carey M, Lindsay B, Pirahesh H, and Reinwald, B. (2001): Efficiently publishing relational data as XML documents. The VLDB Journal, 10(2-3), pp. 133-154.

[8] Galindo-Legaria C.A. (1994): Outerjoins as disjunction. Proceedings of the 1994 ACM-SIGMOD International Conference on the Management of Data, Minneapolis, USA, pp. 248-258.

[9] Rajaraman, A., and Ullman J.D. (1996): Integrating information by outerjoins and full disjunction. ACM PODS, pp. 238-248.