Dublin Institute of Technology
ARROW@DIT

Articles

Directorate of Academic Affairs

2001-01-01

Sharing Health-care Records over the Internet

Jane Grimson
Trinity College Dublin

Gaye Stephens
Trinity College Dublin

Benjamin Jung
Trinity College Dublin

William Grimson
Dublin Institute of Technology, william.grimson@dit.ie

Damon Berry
Dublin Institute of Technology

See next page for additional authors

Follow this and additional works at: http://arrow.dit.ie/diraaart

Part of the Electrical and Computer Engineering Commons

Recommended Citation

Grimson, J., Stephens, G., Jung, B., Grimson, W., Berry, D., Pardon, S.: Sharing health-care records over the internet. IEEE Internet Computing, Vol. 5, no.3, pp. 49-58. 2001.

This Article is brought to you for free and open access by the Directorate of Academic Affairs at ARROW@DIT. It has been accepted for inclusion in Articles by an authorized administrator of ARROW@DIT. For more information, please contact yvonne.desmond@dit.ie, arrow.admin@dit.ie.

This work is licensed under a Creative Commons Attribution-Noncommercial-Share Alike 3.0 License

Authors
Jane Grimson, Gaye Stephens, Benjamin Jung, William Grimson, Damon Berry, and Sebastien Pardon

This article is available at ARROW@DIT: http://arrow.dit.ie/diraaart/7

Sharing Health-Care Records over the Internet

Synapses and SynEx illustrate a generic approach in applying Internet technologies for viewing shared records, integrated with existing health computing environments.

Jane Grimson, Gaye Stephens, and Benjamin Jung
Trinity College Dublin

William Grimson, Damon Berry, and Sebastien Pardon
Dublin Institute of Technology

The health-care sector is highly heterogeneous, widely distributed and fragmented, and has strong local autonomy.[1] Individual patient information is scattered throughout the sector, residing anywhere from primary care physicians' offices to clinical laboratories and specialist centers. From the clinical perspective, delivering appropriate patient care requires access to relevant patient information. Frequently, such information is not available when and where it is needed. This lack results in multiple requests to patients for personal data, unnecessary duplication of tests and other investigations, and ultimately delays in the patients receiving appropriate care.[2]

Fundamentally, the information that must be available at the point of care is the patient's record. Yet, currently, an individual's cradle-to-grave, longitudinal health-care record is fragmented, distributed among all the care providers with whom the patient has ever had contact. Increasingly, this information is stored in electronic form. What is required, therefore, is a means to integrate this information and make it available to those health professionals who need it. The Internet's ubiquity and the Web's ease of access offer a potential solution to such integration, provided the approach is scalable and sufficiently generic to apply in many different clinical domains. The solution must also be easily integrated into existing health computing environments and must guarantee security and quality of service.

This article presents a novel approach to sharing electronic health-care records that leverages the Internet and the Web, developed as part of two European Commission-funded projects, Synapses and SynEx. The approach provides an integrated view of patient data from heterogeneous, distributed information systems and presents it to users electronically. The solution is generic, and prototypes have been validated in a variety of clinical domains and health-care settings.

Synapses and Synex

The key enabling technologies in the devel-

 1089-7801/01/$10.00©2001 IEEE

Authorized licensed use limited to: DUBLIN INSTITUTE OF TECHNOLOGY. Downloaded on August 12, 2009 at 09:55 from IEEE Xplore. Restrictions apply.

opment of shared records are federated databases[3] (collections of autonomous, heterogeneous databases to which integrated access is required), distributed object technology,[4] and the Internet (including intra- and extranets). Both the Synapses (http://www.cs.tcd.ie/synapses/public) and SynEx (http://www.gesi.it/synex) projects applied these technologies to sharing records. Synapses concentrated on the specification of a Federated Healthcare Record (FHCR) server, which provides integrated access to a record's distributed components.[5,6] An Electronic Healthcare Record (EHCR) is a structured multimedia collection of health-care data about an individual patient. Extracts of individual patient records, obtained from a number of EHCR systems and brought together in a structured manner, results in a Federated Healthcare Record (FHCR).

The SynEx project concerned integrating a number of components — the Synapses record server being just one — to form an information system from which client applications could access a wide range of data in support of the health-care business. In SynEx, the record server was further developed as a component within an open, distributed-computing environment[7] in an enterprise-wide manner and, using XML technologies, developed support for Web-based clients.[8] The SynEx name came from the phrase, "synergy on the extranet".

In this article, we address the technical issues associated with providing integrated access, via the Internet, to electronic health-care records, regardless of where they reside or in what format they are stored.

Federating the Record

The Synapses server is a middleware component that gives client applications an integrated view of records, tailored to suit users' needs. For example, intensive-care unit personnel require different views of patient data than, say, primary care physicians. Both, however, would at least share patient identification and demographic information. The health care delivered to an individual patient is the responsibility of a team of collaborating health-care professionals. To provide such integrated care, the team must be able to share patient information easily and securely. Synapses is directed at supporting that care.

The Synapses FHCR is essentially a template of record components populated with patient data from various sources. In Synapses terms, these data sources are *feeder systems,* which are heterogeneous and generally autonomous. Feeder systems can be a simple relational database or another Synapses server, for example. The FHCR captures the concept of independent systems collaborating according to a set of rules to achieve a common goal: a shareable and mutually accessible record. In the Synapses paradigm, health-care organizations and associated departments can select whichever systems meet their needs and best suit their particular environment. The Synapses approach also preserves investment in legacy health-information systems.

Figure 1 (next page) shows an overview of Synapses. The client issues a query for a record component — for example, a patient's demographic details or most recent blood glucose level. The server receives this request and decomposes it into individual queries to the feeder systems that store the requested data; the data requested by one client query can be distributed across numerous feeder systems. The individual feeder systems then process these requests, according to their own protocols, and pass the responses back to the server. The server integrates the responses and returns the result in the form of a record component to the client.

Essentially, the Synapses server provides the functionality of a federated database system.[3] Additionally, individual patient data could be distributed across numerous hospital and community health sites, which might necessitate forwarding a query to a server located elsewhere. A Synapses server at an external site would then act as a federating device, hiding the details of its local feeder systems from the remote client, as Figure 1 shows. This scalability is an important feature of the Synapses approach.

Synapses Goals

Two goals governed the Synapses server's design and implementation:

- The server only presents views of federated records to support shared care; it does not directly support writing into patients' records or processing records. However, as part of the later SynEx project, the Synapses record server was integrated within an existing hospital information infrastructure that does support these features.
- The design optimizes the federation granularity, that is, the server can federate patient data from many types of feeder systems.

The Synapses approach to achieving these goals is unique in that it does not prescribe a rigid structure for the federated record. Instead, it uses a model based on EHCR architectures that lets health-care professionals design the record according to the standard that best suits their needs. Health-care

Authorized licensed use limited to: DUBLIN INSTITUTE OF TECHNOLOGY. Downloaded on August 12, 2009 at 09:55 from IEEE Xplore. Restrictions apply.

professionals need to share data, not models. Accordingly, this approach lets health-care institutions migrate from non-record-based structures to a record-based structure without restrictions on the underlying data model.

Synapses Concepts

Three main concepts underpin the Synapses paradigm: the Synapses object model (SynOM), the Synapses object dictionary (SynOD), and the Synapses record.

The SynOM. All federated database management systems require a canonical model, whether implicit or explicit. Similarly, the creation and communication of an FHCR requires a common model, by which users can view the record. Synapses uses the flexible SynOM, which provides a set of building blocks and aggregation rules for creating an FHCR. These building blocks — referred to as record components — have been described elsewhere, as have the aggregation rules.[5]

Broadly speaking, record components are of two types: those that describe the FHCR structure and those that contain the data to be included in the FHCR. Record components that describe the structure are referred to as *record item complexes* (RICs); those that contain data, *record items* (RIs). These concepts are based on an early, European pre-standard, ENV12265 Electronic Healthcare Record Architecture (http://www.centc251.org/), which has since been superseded by ENV13606. The SynOM is not a fixed record architecture; it is a model for building an FHCR.

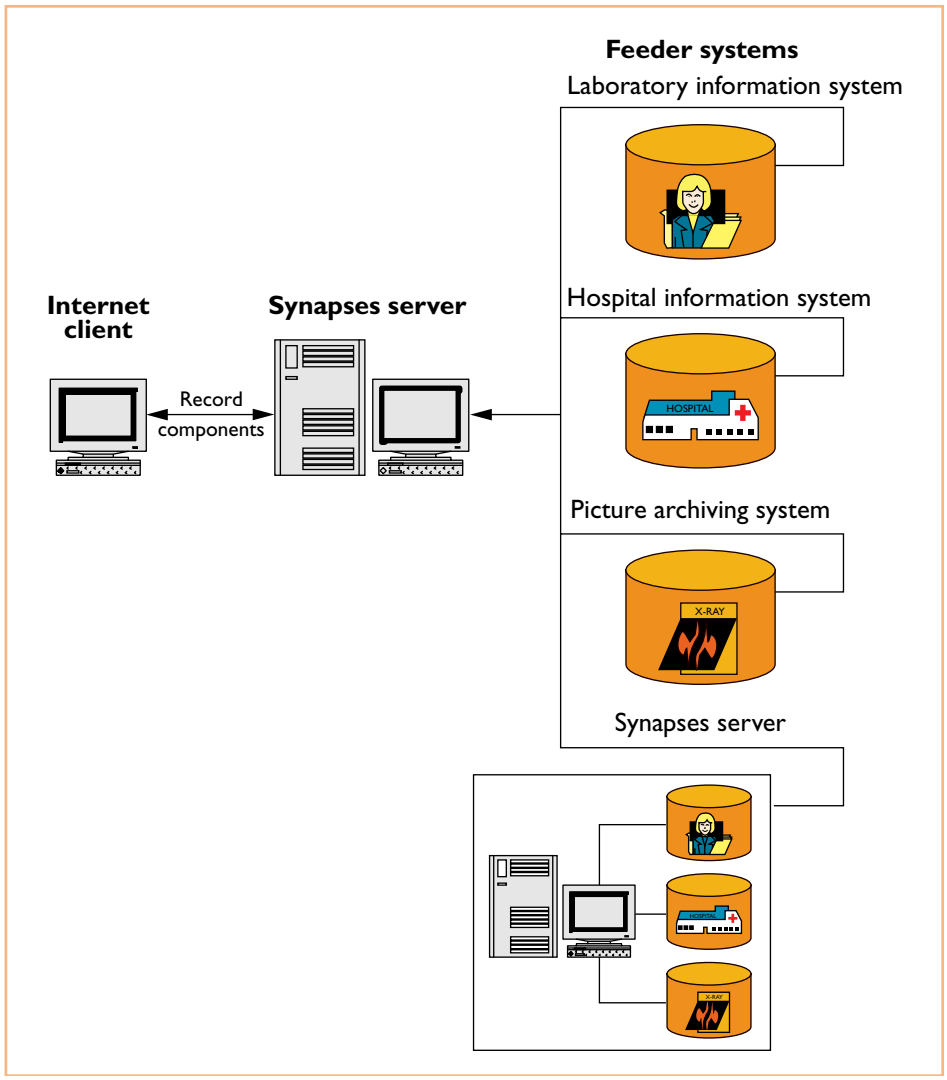

Figure 1. Overview of Synapses. The client issues a request for a record component, and the server decomposes the request into individual queries to the feeder systems that store the requested data. The feeder systems process these requests and pass the responses back to the server. The server integrates the responses and returns the result in the form of a record component to the client.

The SynOD. Health-care professionals describe the "shape" of their FHCR by using the SynOM. These FHCR shapes do not contain any patient data; they're templates that define the record structure. The record templates are stored in the SynOD, which also contains information about the location of the data to populate the FHCR — that is, the details about the feeder systems. (This functionality resembles that of the Healthcare Informational Locator Service component in CORBAmed, http://www.omg.org/corbamed/). Each SynOD can contain many record templates for many different FHCRs defined across the data in the feeder systems connected to an individual server. These templates could be specific to a hospital, a hospital department, a particular health-care professional, a specific class of patients, or an individual patient.

The record. A health-care professional can request a patient's record based on a particular template in

Authorized licensed use limited to: DUBLIN INSTITUTE OF TECHNOLOGY. Downloaded on August 12, 2009 at 09:55 from IEEE Xplore. Restrictions apply.

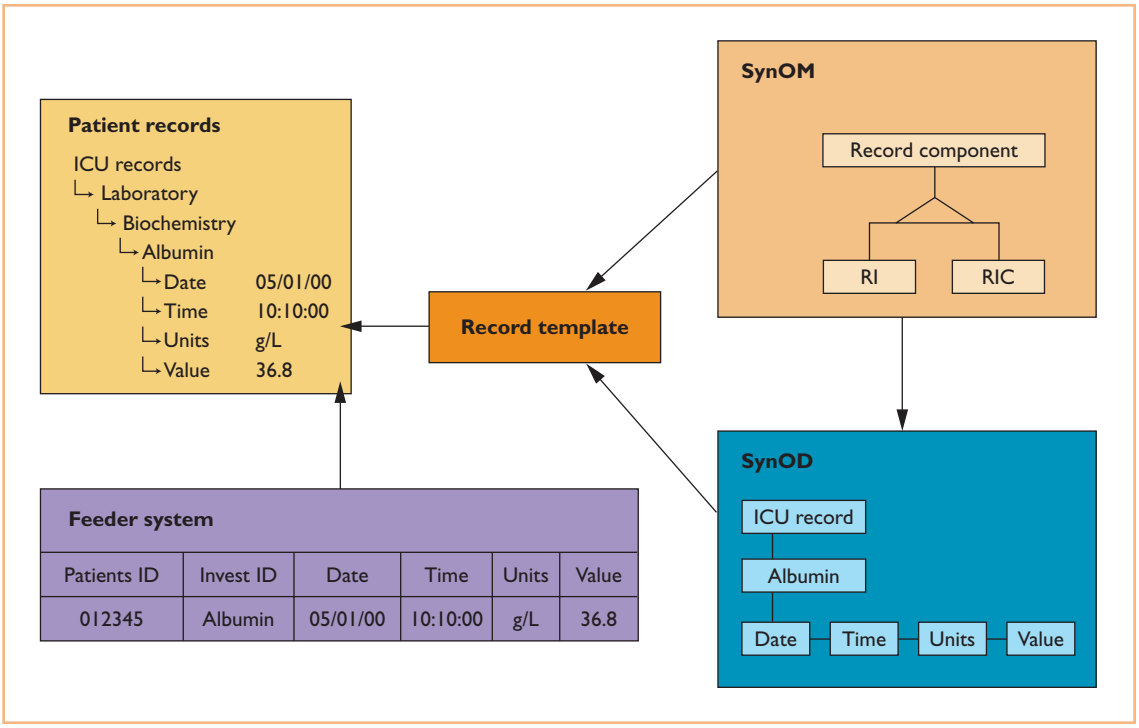

Figure 2. Instantiating the FHCR. A specific SynOD record template is loaded into memory and the dictionary metadata is used to locate and retrieve the relevant patient data from the feeder systems, which is then inserted into the template thus creating the record. Because this record is assembled, potentially from a number of feeder systems, the record is said to be a Federated Health-care Record.

the SynOD. After the request has been issued, the template is instantiated and populated with metadata about the feeder systems containing the patient data. The FHCR record is created from patient data retrieved from the feeder systems and inserted into the template.

As part of a configuration phase, the SynOD is constructed using a Record Structure Builder (RSB), described later. The SynOD entries contain metadata that indicates where in the various connected feeder systems the record components are located and describes how they can be accessed, for example, via SQL scripts. Figure 2 shows the process involved in instantiating an FHCR. The SynOD is used to load a record template into memory (constrained according to the abstract model provided by the SynOM), and the dictionary metadata is used to retrieve the patient data, specified in the record template, from the connected feeder systems. The FHCR is created when all the requested data has been retrieved and inserted into the template.

A record template, once defined, can be populated with data from the feeder systems at runtime. The rules governing the structure ensure that the data retrieved from the feeder systems is inserted in the correct place in the record, and, most importantly, that the context in which the data was originally generated is preserved.

The SynOM allows for great flexibility in building SynODs, but this flexibility could result in a proliferation of templates that makes communication of the FHCR between institutions (servers) difficult. It is the responsibility of health-care institutions to decide what types of FHCRs they want to support; for example, Health Level Seven (HL7) might be the format of choice (http://www.hl7.org/). The Synapses approach facilitates and promotes adoption of such standards by allowing organizations the freedom to structure their records in the SynOD according to their own needs. This flexibility means that even if the connected feeder systems are highly nonstandard, as is frequently the case, the FHCR can be designed to adhere to an emerging standard.

Synapses System Architecture

The Synapses Internet client connects with the Synapses server to request and receive Synapses record components. A browser displays these record components in a structure determined by the SynOD. Effectively, the Synapses client functions as a (federated) record viewer. The server-provided data is available for viewing by users via a browser, or for integration with a separate, independently developed

Authorized licensed use limited to: DUBLIN INSTITUTE OF TECHNOLOGY. Downloaded on August 12, 2009 at 09:55 from IEEE Xplore. Restrictions apply.

application such as a clinical workstation.

The Synapses server is divided into three layers: the communication mechanism, the kernel, and the generic adapter.

Communication mechanism. The communication mechanism layer provides insulation between the client-server interface and the interface's implementation. This allows system developers to update or change the communication mechanism without having to recode the server. In the initial Synapses server implementation, the system developers chose the CORBA standard as the communication mechanism[6] from alternatives that included, for example, DCOM.

Kernel. The kernel deals with the Synapses record components and is responsible for

- creating record components,
- arranging these components as an FHCR, and
- populating the record components with patient data.

Each of these tasks is carried out according to the templates defined in the SynOD. The kernel is unaware of where the data originates; it simply receives the data and transmits it to the Internet client.

The kernel layer also

- validates any SynODs used by the kernel; and
- interfaces with the communication mechanism layer and the generic adapter layer.

Generic adapter. This layer provides the interface between the server kernel on one side and the various feeder systems on the other. Patient data from the feeders is represented in a generic data structure, which the generic adapter subsequently formats as Synapses record components on the server side. The generic data structure can contain data from many feeder systems simultaneously. The generic adapter provides a more scalable solution than interfacing each feeder directly to the server. By connecting the generic adapter of a local Synapses server to the communications mechanism of its remote counterpart, the Synapses architecture scales to a highly distributed system.

SynOD Design

Beyond the runtime components just described, Synapses provides administrative tools for configuring and managing the server, including the Record Structure Builder (RSB), which is used in defining SynOD entries.[9] The RSB is a specialized editor that allows the clinician (with the support of a systems analyst) to construct or define record templates conforming to the general rules of the Synapses object model. These templates would normally be based on existing paper charts. In addition, system developers use the RSB to assemble the metadata about the feeder systems.

Thus the RSB is used in both stages of SynOD definition. In the first stage, the clinician defines the FHCR structure and its components; in the second stage, the system developer identifies how these components are created or derived from the various feeder systems.

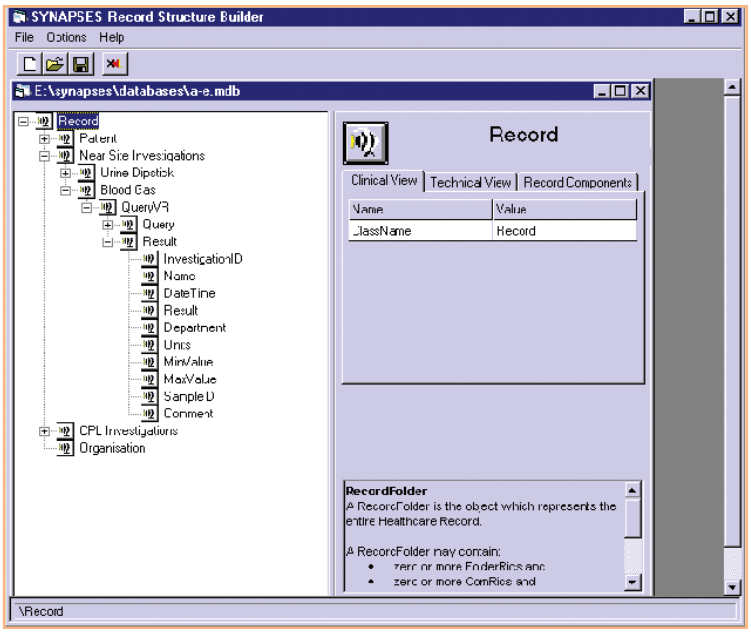

Figure 3. This screen shot from the Record Structure Builder shows the Accident and Emergency SynOD partly expanded. The left window shows the Synapses representation of a clinician's SynOD entry (record template) for a patient.

Record Structure Builder

Figure 3 shows a screen from the RSB during the process of defining an FHCR for an Accident and Emergency Department. The screen's left window shows the Synapses representation of a clinician's SynOD entry (record template) for a patient, which was created by the RSB.

A SynOD consists of a hierarchical collection of folders, which describe the FHCR structure. To reach the patient data, the clinician "drills down" through this metadata structure. The SynOD in Figure 3 contains four folders: Patient, Near-Site Investigation, CPL Investigations, and Organisation. The Near-Site Investigation folder has two subfolders: Urine Dip-

Authorized licensed use limited to: DUBLIN INSTITUTE OF TECHNOLOGY. Downloaded on August 12, 2009 at 09:55 from IEEE Xplore. Restrictions apply.

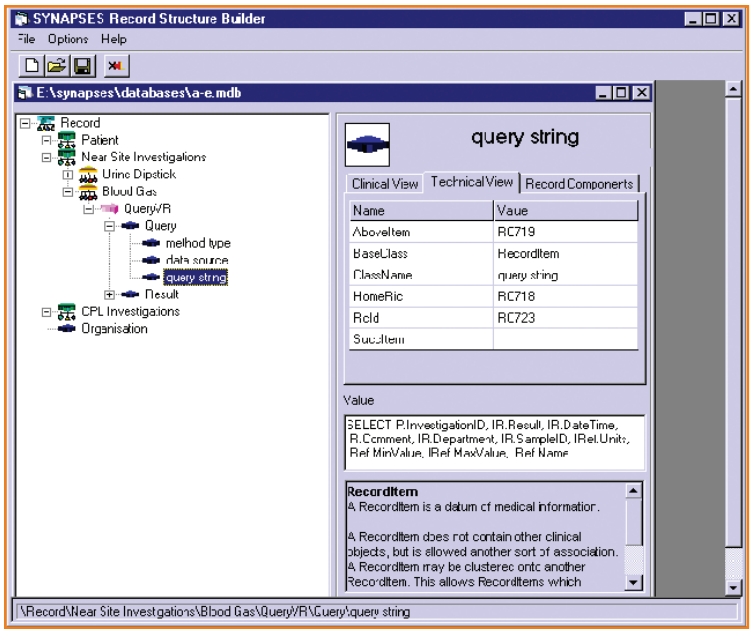

Figure 4. Screen shot from the Record Structure Builder showing the Accident and Emergency SynOD and, at right, the pending query that will retrieve the specified data. The query — obtaining the Blood Gas result — is presented as an SQL string.

stick and Blood Gas. Expanding the Blood Gas subfolder reveals the data elements of interest to the clinician. So, by the end of the first stage, the health-care professionals' data requirements have been identified and their structure specified.

Figure 4 shows the same SynOD as in Figure 3, indicating the query to extract the Blood Gas result (Value box on the right-hand side of the screen) for this portion of the FHCR. In this case, the query is represented as an SQL string. Writing the query script to retrieve the data from the feeder systems would generally require the assistance of a programmer or software engineer who is familiar with the query language of the particular feeder system.

The completed SynOD contains

- a specification of the data requirements to be included in the SynOD and their organization, and
- the means of retrieving these data from a (legacy) feeder system.

An XML-Based RSB

The developers of Synapses based the first version of the RSB on a Visual Basic client, with the resulting SynOD stored on an Access back-end database. At runtime, the server loaded a SynOD into memory and instantiated a record template.

The next RSB version will be Web based and exploit XML for representing the SynOM. Representing the SynOD as a set of XML documents instead of a relational database affords organizations greater flexibility and increased interoperability. A document-template-based mechanism allows the Web-based RSB to populate the XML document that defines the FHCR on-the-fly.[10] XML schemas and vocabularies, as well as XML document templates, provide a new mechanism for creating records. Requests for an FHCR (or fragments thereof) can be interpreted by a document template processor, then executed and merged into the final document, which is sent to the client.

Accessing Synapses via the Web

The first version of the Synapses server was based on CORBA technology, as we've noted. In the SynEx project, the objective was to integrate the Synapses record server as a generic middleware service of the Distributed Health Environment. DHE is a commercial implementation (http://www.gesi.it/dhe/overview.htm) of the CEN TC251 (the Comitié Européen de Normalisation Technical Committee 251, which represents the Health-care Informatics Technical Committee) pre-standard Health Information System Architecture (HISA, http://www.centc251.org). System developers realized, early in the process of defining the SynEx data exchange model, that the user requirements closely matched with the W3C's published XML design goals.[11]

This realization led the system developers to specify an XML document type definition (DTD) for the SynOM known as SynExML.[8] XML features several advantages over existing data formats in supporting patient data exchange between sites. Mainly, XML provides a hardware- and software-independent specification. Both the Synapses-based FHCR model and XML are hierarchical, object-oriented data structures with the option to link elements. Transforming the SynOM into an XML DTD and later into an XML schema is thus relatively straightforward.

Extending Existing Systems

The second reason for introducing XML as the exchange format in the SynEx project was to facilitate extensions to existing systems. Additional interfaces are relatively easy and quick to integrate because the SynOM is already hierarchically organized and internally represented in object-oriented data structures. Transmitting structure and data within a single document, even a human-readable one, is an advantage. Furthermore, the Internet was the obvious choice for the basic transport mechanism because infrastructure and transport protocols

Authorized licensed use limited to: DUBLIN INSTITUTE OF TECHNOLOGY. Downloaded on August 12, 2009 at 09:55 from IEEE Xplore. Restrictions apply.

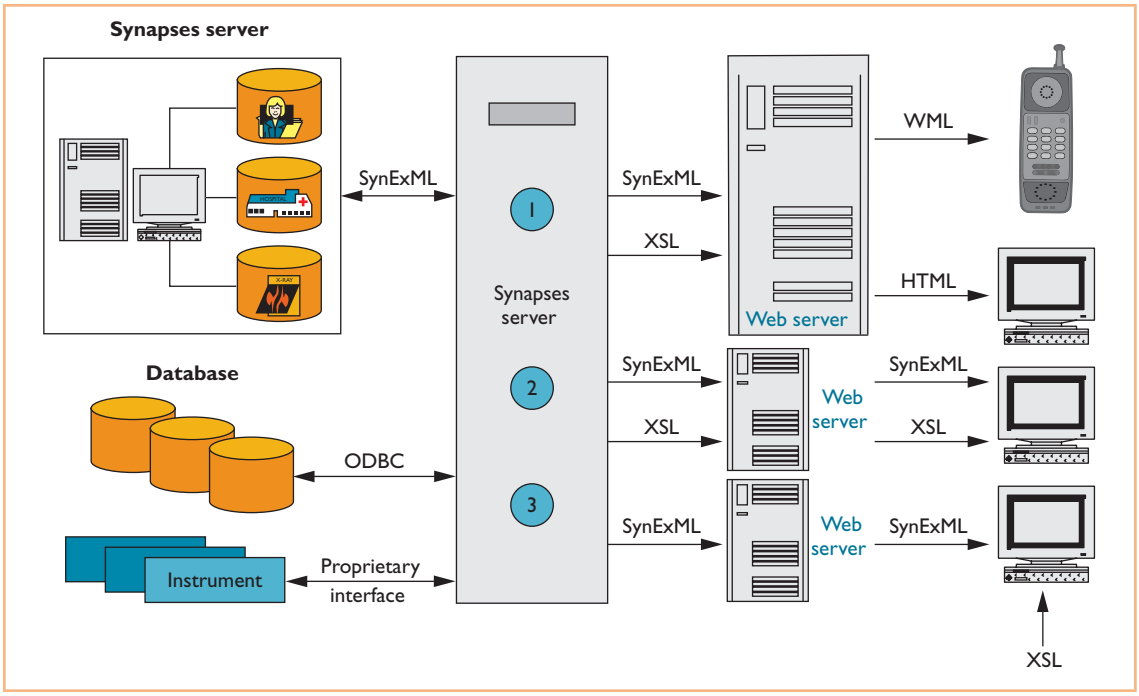

Figure 5. Processing SynExML documents with XSL style sheets. The SynExML document can be retrieved from a variety of sources such as another (remote) Synapses server, local databases, or instruments. XSL style sheets reside on either the server side (1, 2) or the client side (3). Scenario 1 shows XSL processing on the server side into the client's native presentation format (such as HTML or WAP), whereas in scenario 2, the client is responsible for applying the style sheet rules. This allows unlimited transformations without reconnecting to the server in order to retrieve the original data (SynExML) and can lead to scenario 3, where the source is processed with locally available style sheets.

are in place. The Internet connection is straightforward via Web-server and scripting technologies — for example, common gateway interface (CGI), active server pages (ASP), and Java server pages (JSP).

The communication mechanism of the Synapses server's first version was exclusively based on an early version of a CORBA implementation. However, what seemed to be a perfect match increasingly became an obstacle to fast service. Due to CORBA's architecture, a new connection had to be established for every FHCR object. This caused an average FHCR's transmission and retrieval time to be as long as 20 seconds rather than the expected and acceptable fraction of a second. Through XML, a complete FHCR fragment (consisting of more than 600 objects) was wrapped into a SynExML document, which preserved the object-oriented structure, and sent using CORBA within a single connection to the client. This reduced retrieval time by an average factor of 100.

Presentation

A major advantage of XML is flexibility in presenting information on the client side. Synapses is primarily a record viewer. Using Extensible Stylesheet Language (XSL), data presentation can be readily tailored to suit a particular device or environment, from a desktop computer to a WAP phone.

Figure 5 shows three scenarios relevant to the exchange of health-care records, including the processing and display of SynExML with XSL and cascading style sheets (CSS). Synapses feeder systems, including databases, medical instruments, and remote Synapses server, are shown on the left.

In a server-server connection, the remote machine acts like a client to the local server. The connection follows an ODBC-driven (database) method, SynExML (text over HTTP), or instrument-proprietary mechanism, as appropriate, interfacing to the generic adapter on the server side. The scenarios follow.

- *Hardware/software-specific data presentation.* The first scenario consists of a fat server and thin clients. Characteristically, the server processes the XML and XSL and serves hardware-specific (for example, WAP for mobile devices) or software-specific documents (for example, Internet Explorer 5.0+). This would be

Authorized licensed use limited to: DUBLIN INSTITUTE OF TECHNOLOGY. Downloaded on August 12, 2009 at 09:55 from IEEE Xplore. Restrictions apply.

Related Work in Accessing Patient Records

Health-care institutions and vendors have shown increasing interest, and related activity, in providing Web-based access to their systems.[1,2] Moreover, because it isn't difficult to present information, derived from numerous and different underlying systems, in a single browser window, it's theoretically possible to display the "integrated record" to end users. However, without an underlying canonical model of the shared model, such solutions are not scalable.

As far as we know, Synapses is the only approach based on a flexible model that lets health-care professionals construct the record best suited to their needs. Kohane and colleagues,[1] for example, in the World Wide Web–Electronic Medical Record System (W3-EMRS) base their shared record on a common medical record (CMR) standard, which defines the kernel of the record that can be shared. Such a solution is appropriate where reaching agreement on the CMR is feasible and where various systems are tightly coupled. Both the Synapses approach and W3-EMRS identify the need for a layered open architecture to solve the problem of sharing medical records.

Combining the two approaches to create a record system that is independent of a particular record model and that supports updating records across various sources would be feasible. In fact, the Synapses object model could be used to define a CMR-compliant record.

Similar developments are ongoing in, for example, the Khospad [Knocking at the hospital for patient data][3] and OpenEMed (formerly TeleMed) projects, and in the Intermed collaboratory.[4] Khospad builds the record from diverse sources but, unlike Synapses, physically stores it in an object-oriented database. OpenEMed (http://www.acl.lanl.gov/TeleMed/) uses a component-based architecture aligned with the emerging CORBAmed standards (http://www.omg.org/corbamed/).

The OpenEMed, Khospad, and Synapses architectures are similar, integrating data from heterogeneous collections of underlying health information systems. All aim to support shared care, but unlike Synapses, both Khospad and OpenEMed are based, initially, on domain-specific data models for cardiology and radiology, respectively.

Finally, the Intermed collaboratory uses a similar open layered architecture, but rather than specifically focusing on electronic records, it promotes the use of shared concepts for the general development of health information systems. Specific applications to support records electronically are therefore further down the line.

Organizations working on standardization of record architectures include the American Society for Testing and Materials (ASTM), CEN/TC251 (the Comitié Européen de Normalisation Technical Committee 251, which represents the Healthcare Informatics Technical Committee), Health Level Seven (HL7), ISO TC215 (the International Organization for Standardization's Technical Committee 215, on Health Informatics concerned with Health Records and Information Modelling Coordination), and the Object Management Group's CORBAmed. ASTM, CEN/TC251, and HL7 are defining data dictionaries for various medical areas. So far, data dictionaries have been finalized for discharge summaries, clinical notes, admission notes, operative reports, procedure notes, diagnostic imaging, and prescriptions.[5]

CEN/TC251 further developed the very flexible ENV12265 approach, which was the SynOM's basis, and is working on migrating its architecture models into XML schemas. HL7, on the other hand, proposes a multilevel XML architecture for its patient record architecture. The levels concept refers to varying degrees of required markup granularity and specificity and does not refer to the degree of granularity or depth of clinical information contained within the document. This approach generically combines the ideas of CEN/TC251 and ASTM. The medical institution itself defines the level of automated data exchange, depending on the support of the HL7 markup granularity.

References

1. I.S. Kohane et al., "Building National Electronic Medical Record Systems via the World Wide Web," *J. American Medical Informatics Assoc.*, vol. 3, 1996, pp. 191-207.
2. D. Forslund et al., "TeleMed: Development of a Java/CORBA-Based Virtual Electronic Medical Record," *Proc. RSNA 98*, Radiological Soc. of North America Inc., Oak Brook, Ill., 1998.
3. F. Pinciroli et al., "WWW-Based Access to Object-Oriented Clinical Databases: The KHOSPAD Project," *Computers in Biology and Medicine*, vol. 24, 1998, pp. 531-552.
4. D.C. Kilman and D.W. Forslund, "An International Collaboratory Based on Virtual Patient Records," *Comm. ACM*, vol. 40, no. 8, ACM Press, New York, 1997, pp. 111-117.
5. *Minutes and Results*, ASTM Committee E31 and the Massachusetts Medical Society Workshop on XML Document Type Definitions in Health Care, Oct. 1999; http://www.astm.org/COMMIT/E31Oct99rprt.zip.

the ideal solution in environments where clients have no further processing components available. The server processes the SynExML document with the client-specific style sheet, which has been selected on the basis of the requester's ID (extracted from the HTTP request).

- *FHCR source-specific data presentation*. In the second scenario, the local client processes the XML with XSL and CSS. Both the XML document and the style sheet files are provided by the source (server). The receiver can view the data in the originator's preferred presentation format or re-present the information using another style sheet.
- *Localized data presentation*. The third example (shown in Figure 5) is similar to scenario 2, except that the SynExML document is processed with local style sheets. This enables "shallow exchange" in the sense that SynExML from remote sources can be displayed within any SynExML supporting local application.

Authorized licensed use limited to: DUBLIN INSTITUTE OF TECHNOLOGY. Downloaded on August 12, 2009 at 09:55 from IEEE Xplore. Restrictions apply.

Clearly, from these scenarios, XML technologies offer significant flexibility in presenting the record or fragments thereof on the client side, while facilitating data communication across highly heterogeneous computing environments. The technologies provide good support for the health-care professional to experiment with a variety of different FHCR structures and approaches.

Conclusions

Synapses is still in the prototype phase, and although it has been validated using real patient data in real clinical settings, it is not yet routinely used. (For information on other approaches to sharing patient data, see the "Related Work in Accessing Patient Records" sidebar, previous page.) Plans are currently under way to install the server to support the management of diabetic patients in a shared-care environment involving specialist clinics in two Dublin hospitals, a group of primary care physicians, and patients themselves.

Further work is required before Synapses can be deployed to support actual patient care, especially if, as is intended, the EHCR replaces the paper record. In particular, security and quality-of-service issues must be addressed, along with the inevitable business process re-engineering.

Security and quality of service — key issues in the context of Internet-based application development in health care — are being actively promoted as part of the Internet-2 and Next-Generation Internet initiatives.[1] At a technical level, it is possible to ensure the privacy of data transmitted across the network. However, security also encompasses confidentiality, integrity, availability, and accountability.[12] The requirement to address these issues comprehensively is being increasingly driven by new legislation, including in Europe the EU Data Protection Directive and the Council of Europe Recommendation R(97)5,[12] and, in the U.S., the Health Insurance Portability and Accountability Act (HIPAA, http://aspe.hhs.gov/admnsimp/).

Synapses has not yet comprehensively addressed security but has provided "hooks." For example, Synapses, as part of SynEx, supports secure HTTP (SHHTP) over secure sockets layer (SSL), maintains a comprehensive audit trail of all access via the server, and provides authentication and identification through the security component. The plan is to undertake a comprehensive risk analysis based on the SEISMED guidelines[13] so that the risks and benefits from sharing FHCR via the Synapses server are clearly identified and assessed. A particular challenge to be addressed is the fact that Synapses brings together feeder systems, some of which may be less secure than others (for example, out in the community as opposed to the controlled environment of a hospital intranet). Moreover, because individual patient data can potentially be combined with data at unrelated facilities, inferences might be drawn that could compromise patient confidentiality (for example, deducing a patient's medical condition from a listing of drugs prescribed).

The Synapses approach seeks to support an evolutionary approach toward EHCRs by allowing health-care professionals (and potentially patients) to define how they want to view records, rather than requiring them to conform to uniform record structure and content through, for example, specific data sets. However, it is hoped that Synapses will promote consensus through the sharing, for example, of domain-specific SynODs. The evidence from the domains in which Synapses has been validated supports this view. The "standardization" of the record cannot be divorced from the increasing movement toward the standardization of care through evidence-based clinical guidelines and protocols.[14] This in turn leads to the need for comprehensive business process re-engineering.[15] Current research is directed toward integrating clinical guidelines and protocols with the record to produce an active record that can manage the delivery of health care to the individual patient (http://www.cs.tcd.ie/medilink/).

The Synapses FHCR is constructed on-the-fly by the server in response to specific queries from the client and thus represents an integrated view of the data sources stored in the underlying feeder systems. The current system therefore supports only read access, but it is clear that write access to the record will be required, both to annotate the record directly and to update the data in the feeder systems through the record. While it's not possible to support generic update transactions in a federated database environment[16,17] in the presence of full local autonomy, a number of approaches can limit the problem's complexity and provide practical solutions. Such solutions will be simplified by the fact that such "global" update transactions will almost certainly involve only a single local system at a time.

Acknowledgments

We acknowledge the contribution to the research described in this article from members of the Dublin Health Informatics Group at Trinity College Dublin, the Dublin Institute of Technology and St. James's Hospital, and colleagues in the Synapses and SynEx consortia. The work has been partially supported by the EU 4th Framework Programme in Telematic Applications for Health under the Synapses (HC1046) and SynEx (HC4020) projects.

Authorized licensed use limited to: DUBLIN INSTITUTE OF TECHNOLOGY. Downloaded on August 12, 2009 at 09:55 from IEEE Xplore. Restrictions apply.

References

1. *Networking Health; Prescriptions for the Internet*. Tech report TCD-CS-2001-01, Dept. of Computer Science, Trinity College Dublin; available online at http://www.cs.tcd.ie/publications/tech-reports/.
2. R.S. Dick, E.B. Steen, and D.E. Detmer, eds., *The Computer-Based Patient Record: An Essential Technology for Healthcare,* National Academy Press, Washington, D.C., 1997.
3. A.P. Sheth and J.A. Larson, "Federated Database Systems for Managing Distributed, Heterogeneous and Autonomous Database Systems," *ACM Computing Surveys*, vol. 22, no. 3, 1990, pp. 183-236.
4. R. Orfali et al., *The Essential Distributed Objects Survival Guide,* John Wiley & Sons, New York, 1996.
5. W. Grimson et al., "Federated Healthcare Record Server — The Synapses Paradigm," *Int'l J. Medical Informatics*, vol. 52, 1998, pp. 3-27.
6. J. Grimson et al., "A CORBA-Based Integration of Distributed Electronic Healthcare Records Using the Synapses Approach," *IEEE Trans. Information Technology in Biomedicine*, vol. 2, no. 3, Sept 1998, pp. 124-138.
7. J. Grimson, W. Grimson, and W. Hasselbring, "The System Integration Challenge in Healthcare," *Comm. ACM,* vol. 43, no. 6, June 2000, pp. 49-55.
8. B. Jung and J. Grimson, "Synapses/SynEx Goes XML," *Proc. Medical Informatics (Europe 99)*, IOS Press, Amsterdam, the Netherlands, 1999, pp. 906-911.
9. B. Jung et al., "A Record Structure Builder for Synapses," *Proc. 3rd Ann. Conf. and Scientific Symp. Health Informatics Soc. of Ireland*, Health Informatics Soc. of Ireland, Dublin, 1998.
10. J. McKeown and J. Grimson, "SVG: Putting XML in the Picture," *Proc. XML Europe 2000,* Graphics Comm. Assoc., Alexandria, Va., 2000, pp. 903-912.
11. T. Bray, J. Paoli, and C.M. Sperberg-Queen, "Extensible Markup Language (XML) 1.0," W3C Recommendation, 10 Feb. 1998, available online at http://www.w3.org/TR/REC-xml/.
12. B. Barber, K. Louwerse, and J. Davey, "White Paper on Health Care Information Security," available online from http://ted.see.plym.ac.uk/ishtar/deliverables/white%20paper.html/, 1998.
13. SEISMED [A Secure Environment for Information Systems in Medicine] Consortium, ed., "Data Security for Health Care," *Studies in Health Technology and Informatics,* IOS Press, Amsterdam, the Netherlands, vols. I-III, 1996.
14. D. Sackett et al., "Evidence-Based Medicine: What It Is and What It Isn't," *British Medical J.,* vol. 312; 1996, pp. 71-72.
15. J. Grimson, "Delivering the Electronic Healthcare Record for the 21st Century," *Int'l J. Medical Informatics*, to be published, 2001.
16. P.A. Bernstein, V. Hadzilacas, and N. Goodman, *Concurrency Control and Recovery in Database Systems,* Addison-Wesley, Reading, Mass., 1987.
17. T. Öszu and P. Valduriez, *Principles of Distributed Database Systems,* 2nd ed., Prentice Hall, Upper Saddle River, N.J., 1999.

Jane Grimson is dean of research and chairperson of the Centre for Health Informatics, Trinity College, Dublin. Her research interests include distributed database systems and health informatics. Grimson has a master's and doctorate in computer science from the University of Toronto and the University of Edinburgh, respectively. She is a Chartered Engineer, a member of the IEEE and the ACM, and a fellow and past president of the Institution of Engineers of Ireland.

Benjamin Jung is a full-time lecturer and PhD student in the Knowledge and Data Engineering Group, Trinity College, and CEO of deepX Ltd. His research interests include databases, document architectures, and XML technologies in medical informatics. Jung holds a degree in computer science and theoretical medicine from Technische Universität München, Germany.

Gaye Stephens is a lecturer and MSc project coordinator for the Centre of Health Informatics, Trinity College. Her research interests include the current state of technology in the Irish health-care system and end users' views on, and client aspects of, distributed systems. Stephens received a diploma in computer science from the Dublin Institute of Technology, a degree in computer science from the British Computer Society and an MSc from Trinity College Dublin.

Damon Berry has recently joined the full-time academic staff in the School of Control Systems and Electrical Engineering in the Dublin Institute of Technology (DIT). His research interests include test request protocol systems, medical instrument interfacing, and health-care record architectures. Berry received a BSc in control systems and electrical engineering from DIT and an MSc in computer science from Dublin City University.

Sébastien Pardon is a researcher in the Department of Electrical Engineering, Dublin Institute of Technology. His research interests include electronic patient records, protocol systems, and Java programming. Pardon has an honours diploma in electrical engineering from DIT, and a degree in computers from ESSTIN (Ecole Superieure des Sciences et Technologies de l'Ingenieur de Nancy), France.

William Grimson heads the Department of Control Engineering at the Dublin Institute of Technology. His research interests include the specification and implementation of information systems, systems modeling, system integration, and the development of standards in health-care informatics. Grimson has a BA in mathematics and economics and a BAI in electronic engineering from Trinity College, Dublin, and an MASc from the University of Toronto. He is a Chartered Engineer and is a member of the Institution of Engineers of Ireland.

Readers can contact Jane Grimson at jane.grimson@tcd.ie.

Authorized licensed use limited to: DUBLIN INSTITUTE OF TECHNOLOGY. Downloaded on August 12, 2009 at 09:55 from IEEE Xplore. Restrictions apply.