SynExML as a vehicle for Electronic Patient Records

A status report from the SynEx project

Benjamin Jung
benjamin.jung@cs.tcd.ie
www.cs.tcd.ie/CHI/
Trinity College Dublin, , Dublin , Ireland

Egil P. Andersen
Egil.Andersen@nr.no
www.nr.no/
Norwegian Computing Center, , Oslo , Norway

Jane Grimson
jane.grimson@cs.tcd.ie
www.cs.tcd.ie/CHI/
Trinity College Dublin, , Dublin , Ireland

Abstract:

The advent of shared care coupled with the proliferation of heterogeneous distributed systems containing patient data have resulted in an urgent need to provide seamless integration of distributed electronic patient records. This paper presents the approach to this problem, which is being developed as part of a major EU (European Union) Health Telematics project, SynEx. The approach is based on a core XML (eXtensible Markup Language) DTD corresponding to a flexible common model of the shared - or federated - Electronic Healthcare Record. Using this DTD, SynEx has successfully demonstrated the exchange of records between two heterogeneous and geographically separated sites.

Introduction and motivation

The management of electronic patient data was relatively easy as long as the data was collected, stored and viewed in a closed environment like a single hospital/department or doctor's practice. A single vendor provided compatible systems and a centralised system administrator was responsible for the smooth running of all installed components. No 'digital' connection to the 'outside world' was provided which avoided many of the problems associated with "open" systems, in particular safeguarding the security and confidentiality of patient data. With the use of hospital-specific information system, that are well known by the staff, maintenance and use became very easy and people felt familiar with it. On the other side, complexity increased with each new version and/or change to a different product. Data exchange with external organisations was only possible by paper, mail, fax or telephone, which is clearly both time-consuming and error-prone. Any feedback and results had to be manually entered and often even cross-checked. This approach is no longer sufficient. Increased computerisation throughout the health sector has given rise to a proliferation of independent systems storing and maintaining patient data. However, the growing trend towards shared care requires that these systems are able to share their data. This has led to the development of projects such as Synapses and its successor, SynEx that aim to provide healthcare professionals with integrated access to patient records and related data. One of the goals of SynEx is to bring together the work on Federated Healthcare Records from the Good European Health Record project , emerging standards from CEN/TC251 (Comité Européen de Normalisation Technical Committee 251) and the Synapses project with the Information Systems perspective of middleware based pre-standard HISA (Health Information Systems Architecture) . The Synapses project exploited ideas from federated database technology that provide client applications with an integrated view of data stored in heterogeneous, distributed database systems. At the heart of Synapses is the FHCR (Federated Health Care Record) server that accepts requests for data (in the form of clinical objects) from clients, decomposes them into queries against the connected "feeder" systems,

where the data is actually stored, and integrates the responses dynamically. Synapses was concerned with the specification of an open standard for the server and its interfaces and for pragmatic reasons used an ad hoc mechanism for exchanging clinical data between feeders, server and client. Obviously, a candidate for such an exchange format would be based on XML and this is being actively pursued in the SynEx project. This paper assesses how XML could be usefully incorporated into both Synapses and SynEx.

XML has the power to become the independent data exchange format of the future. The use of XML to exchange data between heterogeneous hard- as well as software-systems provides support for hierarchical structured patient data, user defined tags and machine-understandable assertions for searching, reasoning and analysing FHCR objects. This paper presents the Synapses approach to FHCRs and explains how the common model of the FHCR developed in Synapses has been used as the basis for the core XML DTD in SynEx (SynExML (SynEx Markup Language)). A demonstration of the integration of records and exchange of information between two sites is described and the paper concludes with a brief review of the future direction of this research. It furthermore outlines some ideas on how the use of XML as the basic data-structure can help integrating hardware-devices, such as handheld-devices and mobile phones, into the internal and external patient care environment.

The Synapses approach

The Synapses Record Architecture specifies in a generic and simplified way the SynOM (Synapses Object Model) and the SynOD (Synapses Object Dictionary). More detailed and technical descriptions can be found in .

In contrast to the very detailed and highly specific record architectures of for example HL7 (Health Level 7) , the SynOM is a generic and flexible common object model. It extends the model of the EPR (Electronic Patient Record) described in ENV12265 from CEN/TC251, . Experiences from ongoing work and implementations were fed back into the development of ENV12265's successors ENV13606/1-4 .

The Synapses Record Architecture consists of a single class hierarchy, and every Synapses patient record consists of a set of objects where each object is instantiated from a class in this hierarchy. Each class in the hierarchy belongs to one of two main groups of classes. The structure of a record is made up of objects instantiated from the **structural classes**, called RIC (Record Item Complexe) , while the data (information) within a record consists of objects instantiated from **data value classes**, called RI (Record Item) .

The structure of a Synapses record corresponds to a tree structure of RIC objects, and each record can have unidirectional links to other such tree structures, i.e. to other Synapses records. The tree structure of each record is rooted in a single **RecordFolder** object, instantiated from the class **RecordFolder**, which represents the overall record. Below this object there will be a structure of folders (**FolderRIC** objects) and documents (**ComRIC** objects). Each document will itself consist of a tree structure of objects, which can be **DataRIC** objects and/or **ViewRIC1** objects. The former contains information that is explicitly stored in the record, while the latter is used to represent computed or derived information. In addition there are objects that represent links to other Synapses records, called **ViewRIC2** objects. These are the keys to the integration of Synapses records. They contain the unique identification of another RIC object, and they are used as follows. The root object of a record, or a folder within a record, may contain a single **ViewRIC2** object that references another record or folder object respectively. In addition, a document may contain one or more **ViewRIC2** objects that each reference some subset of other documents. The RIC object referenced by a **ViewRIC2** object is either local, intra- or inter-record, or remote. If remote then the **ViewRIC2** object contains an URI (Uniform Resource Identifiers) that identifies the server where the target RIC object resides.

Each RIC object instantiated from one of the "structural classes" will have a small set of static, predefined attributes, e.g. as required for their unique identification, or the target address of a **ViewRIC2** object. However, most of the information content of a record, and all the medical information, exists in RI objects instantiated from the "data value classes". That is, a set of RI objects can be attached to a structural RIC object and thus function as its dynamic attributes with actual data values such as a blood pressure measurement. The RI objects that belong to a particular RIC object can also be organized into a tree structure. This allows the information content of a record to be dynamically extended, including new information types that may not have been foreseen at the time the record itself was created.

The distinction between RIC classes and RI classes comprises of "vertical" grouping of the overall Synapses class hierarchy. In addition the class hierarchy is split "horizontally" into a predefined set of base classes common to every Synapses server, called the SynOM, and an extendable set of classes that are derived from these SynOM classes, called the SynOD. The above RIC classes **RecordFolder**, **FolderRIC**, **ComRIC**, etc, all belong to the SynOM, and they define the core part of Synapses' generic record model. The SynOD classes on the other hand are site specific and thus may differ for each Synapses server, are the classes from which the actual patient record objects are instantiated. Thus while every record object has the above SynOM characteristics and properties, they can also be customised to the needs of each individual site.

Supporting software tools will guide and help the user in the process of creating SynODs and linking the Synapses Server with the different feeder systems. One candidate for such a tool is the RSB (Record Structure Builder) , that was developed by the Dublin group. It's functionality includes a GUI (Graphical User Interface) to the SynOD, context sensitive management of RICs and RIs, and localisation of Synapses concepts into site-specific clinical concepts for even easier maintenance.

SynExML - the story so far

There are several advantages over existing formats to the use of XML to support the exchange of structured patient data between diverse sites. XML provides a hardware- and a software- independent specification. Furthermore, both the Synapses based FHCR model and XML are hierarchical/oject-oriented data structures with the option to link their elements. It was assumed that a straightforward transformation from SynOM into an XML DTD (Document Type Definition) would not cause any problems. This proved to be somewhat optimistic as XML is based on a very flexible specification and allows 'colourful' variations.
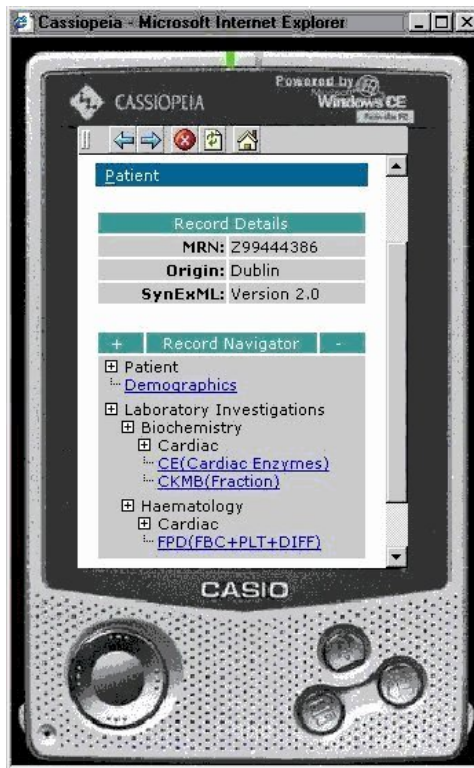
Figure 1. HHD SynEx client

The advent of XML presented a new solution to achieving an agreed common model for the exchange of Synapses FHCRs. Not only did it provide a structure to define the model (DTD), but it also showed a first possible use and implementation (instantiation of an XML document). It was decided to investigate the use of the XML DTD as another modelling tool. Hopes in the new technology were high and local variations were accommodated by the optional introduction of site-specific extensions to the DTD. Unfortunately, this rapidly led to a situation in which these local extensions exceeded the commonly agreed core part of the DTD by a factor of four. It quickly became clear that XML had not eased the problems associated with data exchange as had been hoped. Following a review, it was agreed to eliminate virtually all these locally proposed extensions and to concentrate on the common core, on which consensus was achieved. Unfortunately, there is no common methodology or standard that could be used as a guideline for this conversion. Due to the flexible nature of XML, many different (legal and acceptable) conversions are possible. The project decided to use basic DTD modelling as the only method, as this is (at the moment) less limiting and a clean procedure.

The second main goal for the introduction of XML as the exchange format was to facilitate extensions to existing systems. The integration of additional interfaces is relatively easy and quick to realise as the SynOM is already hierarchically organised and internally represented in object-oriented data structures. The transmission of 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 as infrastructure and transport protocols are already in place. The connection to the Internet via web-server and scripting technologies (e.g. CGI (Common Gateway Interface) , ASP (Active Server Pages) , JSP (Java Servlet Pages)) can be quickly achieved.

Last but not least, the separation of content (XML) and associated rules how to present the content on the client side (using e.g. XSL (eXtensible Stylesheet Language)) made the effort of developing SynExML in the SynEx project a success. Each of the SynEx data providers delivers a stylesheet, which describes how to present their data on the client side with the option of locally selecting another (preferred and/or personalised) view. Recent work has been undertaken not only to create stylesheets for localisation purposes but also to disseminate content more flexible to additional hardware, such as HHD (Hand Held Devices) , PD (Palmtop Devices) and mobile phones (1). A whole range of new the medical personnel supporting applications is envisaged. 3 shows the three axis of the XML triangle and their relation to each other. XSL (with it's subsections of XSLT, FO and Xpath) is the perfect candidate to provide intra-axis mapping (e.g. between HL7 and SynExML) and inter-axis mapping (e.g. between proprietary/limited sets of HTML implementations). This can be achieved on the server as well as on the client side (see 2).
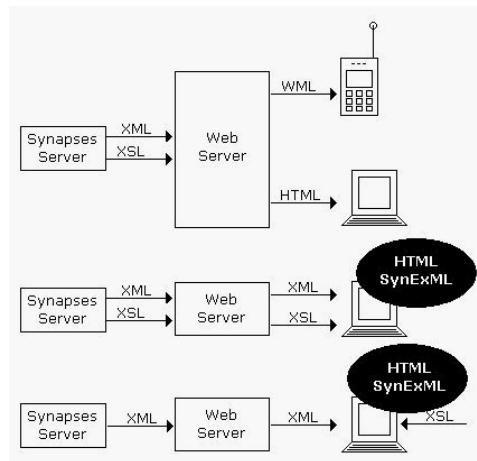

Figure 2. Server-/client side formatting of medical data

This diagram shows the three different ways in which XML documents are processed with the corresponding XSL stylesheets. This can occur on the server as well as on the client side :

1. The data is delivered in a ready to publish format (native format) to the client application.

2. The data as well as the processing rules are delivered from the same source system to the client and processed after arrival.

3. The client retrieves data and processing rules from different systems and processes them into the local application format.

The SynExML DTD (version 2.2) is publicly available and already implemented and used in prototype environments to exchange and link FHCRs between Synapses Servers in Dublin and Oslo. Demonstrations can be viewed at the following sites:

Dublin: http://wilde.cs.tcd.ie:9090/SynEx/Demonstrator/ and http://wilde.cs.tcd.ie:9090/SynEx/Demonstrator/Palmto

Oslo: http://citroen.nr.no/SynExDemo/

Record integration and information exchange

Synapses servers do not rely on a central catalogue service for retrieving information on where the various parts of a particular patient record reside. Instead, as explained above, this information is distributed such that every record on a particular server has the information required to access other parts of it that reside in other servers. Thus after having agreed on the SynExML DTD as the common XML format for the information exchange, the hyperlink capabilities of Synapses in combination with state-of-the-art web technology makes it fairly straightforward to achieve seamless integration of distributed patient records.

For example, consider a user that connects to a SynEx compliant site such as RH (Rikshospitalet) in Oslo, the national hospital of Norway. The user's access rights are collected and stored from an initial login operation. After the successful login, the user chooses what he wants to browse from a set of available records and record fragments. Parts of a particular selected record may reside at other sites such as SJH (St. James's Hospital) in Dublin. When presenting the user with information from this record, the fact that the information content is distributed should be transparent to the user. If a document from RH is requested it will be retrieved from the current server, while if a document from SJH is requested then the client will forward the request to SJH transparently to the user and retrieve the requested document from there. The information to forward the request is entered at the first request of data from SJH and will be stored at RH whereas the data itself resides at SJH. Of course, using this technique might lead to multiple levels of redirection, but it ensures a consistent storage of data at exactly one place.
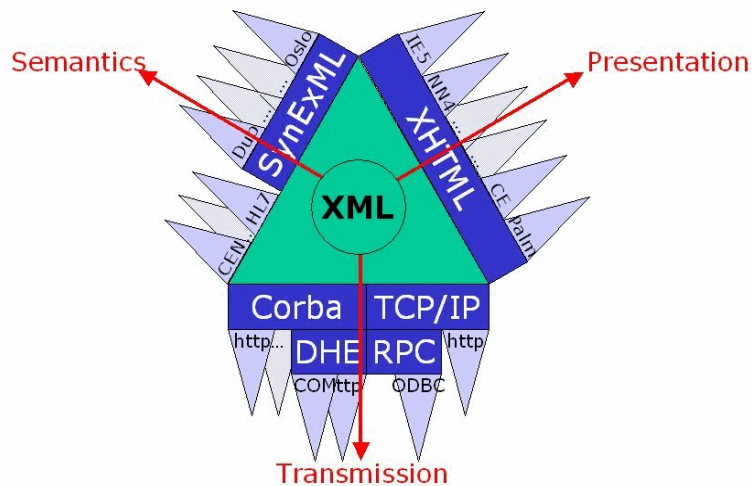

Figure 3. XML triangle

An important characteristic of this integration scenario is it's client-side processing, where the requests were made. Naturally, this implies that the client application supports functionalities like XML parsing and XSL processing. Without this support, server-side processing is required, which limits the flexibility on the client but broadens the support of components in relation to hard- and software. The server's only responsibility is to maintain valid links to where remote parts of its records reside. Furthermore, the implementation of a particular Synapses server, and its legacy feeder systems, is irrelevant to the integration. For example, the Dublin Synapses server is based on a C++/CORBA environment connecting to various data sources via a generic database interface (Generic Adapter), while the Oslo Synapses server uses MTS (Microsoft Transaction Server) with COM (Component Object Model) components as the application layer, and SQL (Structured Query Language) Server for the data store. Thus a first attempt has been made to standardise the web server interfaces. This will be an important issue for the upcoming developing phase to complete

the automatic data exchange. **Request** as well as **Exception** ELEMENTs will be added to the next version of the SynExML. The Dublin web server interface is based on C++ CGI scripts, while the Oslo web server uses ASP; this is just a matter of implementation, both are equally suitable and could easily be exchanged or even replaced by a third one. Even the insertion of a native http-interface to the existing server to speed up the data retrieval process will be part of the future research activities. ASPs in Oslo implemented in a manner that corresponds technically with the new Microsoft SOAP (Simple Object Access Protocol) , where requests to the server are themselves formatted in XML; i.e. XML sent to the server specifies a sequence of functions and their parameters. After parsing the received XML the server executes the specified functions and returns the resulting XML to the client.
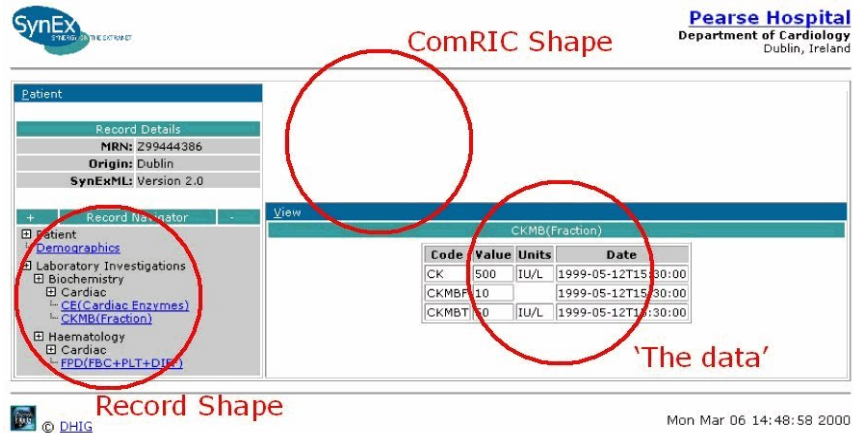

Figure 4. SynEx generic client

A great benefit of basing the information exchange on XML is that it makes the technology offering this information transparent to the task of achieving seamless integration. It would have been much more cumbersome and time-consuming to achieve record integration based on a common protocol, e.g. (D)COM or CORBA (Common Object Request Broker Architecture) components.

As a first step in the process of transparently exchanging FHCRs, all parties in the SynEx consortium agreed to provide the functionality to process and display patient records that are marked up according to the SynExML DTD. As far as the local institution doesn't already use SynExML, it requires a single mapping to and from their local schema into SynExML . For further demonstration of the Synapses concept, the Dublin group developed a prototype of a Generic Client (4), based on MS Internet Explorer 5 and support for XML/XSL, that displays the medical data in various (but predefined) structures, depending on their Synapses concept. This ensures the presentation of remote data in (at least) one way and provides methods to override the predefined stylesheet with local enhancements.

Summary and conclusions

In summary, the development of the SynExML DTD and its possible future use have brought the objective of seamless integration of patient information closer than it was originally envisaged. It even provoked and pushed the idea in the Dublin group to transmit Patient Records to a number of different Hardware platforms (e.g. Handheld/Palmtop-devices, mobile phones) using various transport mechanisms and standards. Agreement on a common syntax for the FHCR and exchange mechanisms and the foreseeable progress in the development of web-based APIs as well as the flexible conversion of XML documents into various formats using XSL made it work. However, the use of XML documents following the SynExML specification, together with HTTP (Hypertext Transfer Protocol) support and a hardware- and software-independent

data format do not by themselves provide a flexible exchange of patient records. More research has to be undertaken in the following three major areas before real, automated integration of patient information can take place:

1. XML by itself is only a transport format for data: easy and powerful, flexible and independent. Without applications, which interpret and present the XML document at the receiving site, the interpretation of the data is possible but highly complex. XSLT (XSL Transformations) and FO (Formatting Objects) provide a generic way to specify publishing rules for display in e.g. web-browsers. More specific applications are imaginable and will follow.

2. The syntactical and semantic mapping between different models (i.e. SynODs in various medical institutions, but also for exchange with other EPR standards standard such as HL7 and CEN) still is and will be the main issue for automated data exchange between heterogeneous systems in the longer term. Because of its flexibility, XML can help to speed up the process of developing an integrated data exchange. Recent implementations within the SynEx project guarantee the 'dumb' display of patient data conforming to the SynExML DTD in their local applications. This was envisaged as the first step in the integration process. Future work will include mapping large data-structures, followed by the automated data-entry into local data-storage systems.

3. Emphasis has to be put on the automated conversion of pure statistical data into visually more appealing, easier to understand and interpretative diagrams. Clearly, this shouldn't decrease usability through the use of large images and therefore excessive (mis-)use of bandwidth. Instead, new specifications in the XML-based vector-graphics area like SVG (Scalable Vector Graphics) and VML (Vector Markup Language) will lead the way to more suitable image transmission over low-bandwidth connections (e.g. wireless).

Acknowledgments

Bibliography

[C13606]   CEN/TC251, **Health Informatics - Electronic Healthcare Record Communication, Part 1: extended Architecture, Part 2: Domain Term List, Part 3: Distribution rules, Part 4: Messages for the exchange of information**, **http://www.centc251.org/WItems/listwis.htm#WGI** , 1999.

[Env97]   CEN/TC251 WG I, **Healthcare Information System Architecture Part 1 (HISA) Healthcare Middleware Layer**, Final Draft prENV 12967-1, March 1997.

[FG99]   Ferrara, F.M., Grimson, B., **The holistic architectural approach to integrating the healthcare record in the overall system**, Proceedings MIE99, IOS Press, pp. 847-852, 1999.

[GBG98]   Grimson, W., Berry, D., Grimson, J., Stephens, G., Felton, E., Given, P. and O'Moore, R., **Federated healthcare record server - the Synapses paradigm**, International Journal of Medical Informatics, 1998.

[GGB98]   Grimson, J., Grimson W., Berry D., Stephens G., Felton E., Kalra, D., Toussaint, P. and Weier, O. **A CORBA-based integration of distributed electronic healthcare records using the Synapses approach**, IEEE Transactions on Information technology in Biomedicine, vol.2, no. 3, Sept, 1998, 124-138.

[HL7]   **HL7 Homepage**, <u>http://www.hl7.org/</u>

[In95]   Ingram D, **The Good European Health Record, In Health in the new communication age**, MF Laires, MF Ladeira and JP Christensen (Eds), IOS, 1995, 66-74.

[JGG99]   Jung, B., Grimson, W., Grimson, J., **The EHCR - if not now, when?**, Proceedings of "Towards An Electronic Health Record Europe '99" (TEHRE99), pages 51-55, England, London, C. Peter Waegemann (Eds.), Medical Records Institute, November, 1999.

[JuGr99]   Jung, B., Grimson, J., **Synapses/SynEx goes XML**, Proceedings of the Medical Informatics Europe '99 Conference (MIE99), pages 906-911, Slovenia, Ljubljana, Peter Kokol, Blaz Zupan, Janez Stare, Marjan Premik and Rolf Engelbrecht (Eds.), IOS Press, August, 1999.

[ShLa90]   Sheth, A.P., Larson, J.A., **Federated database systems for managing distributed, heterogeneous and autonomous databases**, ACM Computing Surveys, 22, 183-235, 1990.

[SVG00]   Ferraiolo, J., **Scalable Vector Graphics (SVG) 1.0 Specification**, <u>http://www.w3.org/TR/SVG/</u> , W3C Working Draft 03-March-2000.

[Syn99]   **Synapses Homepage**, <u>http://www.cs.tcd.ie/synapses/public/</u> .

[Synex99]   **SynEx Homepage**, <u>http://www.gesi.it/synex/</u> .

[TC251]   **CEN/TC251 Homepage**, <u>http://www.centc251.org/</u> .

[VML98]   Mathews, B., Lee, D., Dister, B., Bowler, J., Cooperstein, H., Jindal, A., Nguyen, T., Wu, P., Sandal, T., **Vector Markup Language (VML)**, <u>http://www.w3.org/TR/NOTE-VML/</u> , W3C Note 13-May-1998.

[XMI99]   **XMI Toolkit**, <u>http://www.alphaworks.ibm.com/tech/xmitoolkit/</u> .

[XML98]   Bray, T., Paoli, J., Sperberg-McQueen, C. M., **Extensible Markup Language (XML) 1.0**, <u>http://www.w3.org/TR/1998/REC-xml-19980210</u> , W3C Recommendation 10-February-1998.

[XSL00]   Adler, S., Berglund, A., Caruso, J., Deach, S., Milowski, A., Parnell, S., Richman, J., Zilles, S., **Extensible Stylesheet Language (XSL) Version 1.0**, <u>http://www.w3.org/TR/xsl/</u> , W3C Working Draft 12-January-2000.

Authors

Benjamin Jung

Research Assistant
Trinity College Dublin
Postal Address:
    Centre for Health Informatics
    Dept. of Computer Science
    2 Dublin
    Ireland
Telephon: +353 (1) 608 1335
Fax: +353 (1) 677 2204
E-mail: benjamin.jung@cs.tcd.ie
Web: www.cs.tcd.ie/CHI/

Benjamin Jung - Benjamin Jung is a PhD student and Research Assistant in the Knowledge and Data Engineering Group at Trinity College Dublin. Before he came to Ireland he graduated from Technische Universitaet Muenchen in 1996 with a masters degree in computer science and theoretical medicine. He is involved in two major European Medical Informatics projects (Synapses, SynEx), where he works in the Electronic Healthcare Record Architecture group. Within the EU-project SynEx, he is the Technical Team Leader for Trinity College Dublin and additionally chairs the SynExML task-force which deals with the development of a common DTD for the exchange of Electronic Patient Records (EPR). He is also a member of the CEN/TC251 XML-Taskforce and co-founder of the local XML working group in the computer science department of Trinity College Dublin.

Egil P. Andersen

Senior Research Scientist
Norwegian Computing Center
Postal Address:
    P.O.Box 114
    Blindern
    0314 Oslo
    Norway
Telephon: +47 (22) 85 25 94
Fax: +47 (22) 69 76 60
E-mail: Egil.Andersen@nr.no
Web: www.nr.no/

Egil P. Andersen - Egil P.Andersen is a senior research scientist at the Norwegian Computing Center in Oslo, Norway, and he represents Siemens Health Services in the SynEx project. He has 10 years of research and development experience with object-oriented and relational systems. His current research interests are object-oriented analysis/design and software architectures, particularly relating to the development of distributed, component-based information systems. He has a BSc, MSc and PhD degree in computer science.

Jane Grimson

Associate Professor
Trinity College Dublin
Postal Address:
    Centre for Health Informatics
    Dept. of Computer Science
    2 Dublin
    Ireland
Telephon: +353 (1) 608 1780
Fax: +353 (1) 677 2204
E-mail: jane.grimson@cs.tcd.ie
Web: www.cs.tcd.ie/CHI/

Jane Grimson - Professor Jane Grimson obtained her batchelors degree in Engineering from Trinity College Dublin and her Masters and Doctorate in Computer Science form the Universities of Toronto and Edinburgh. She joined the Department of Computer Science in Trinity College Dublin as a lecturer in 1980 where she is now an Associate Professor. She was Dean of the Faculty of Engineering and Systems Sciences from 1996 to 1999. She has published

over 80 papers in journals and the proceedings of international conferences, and has co-authored a textbook on Distributed Database Systems published by Addison-Wesley. Her main research interests are in distributed information systems and health informatics. She has acted as principal investigator on several EU and national research projects. She established the Knowledge and Data Engineering Research Group in the Department of Computer Science and also, together with colleagues from the Faculty of Health Sciences, the Centre for Health Informatics in Trinity College. She is a Chartered Engineer and Fellow of the British Computer Society, Irish Computer Society, Institution of Engineers of Ireland, Irish Academy of Engineering, and the Royal Academy of Medicine in Ireland and a member of ACM and IEEE. She is a member of the Irish Council for Science, Technology and Innovation and chaired the Materials and Manufacturing Processes Panel of the Government's Technology Foresight Initiative. She is President of the Institution of Engineers of Ireland for 1999-2000.