Health Informatics Journal

Evaluation of ISO EN 13606 as a result of its implementation in XML

Tony Austin, Shanghua Sun, Taher Hassan and Dipak Kalra

Published by:

$SAGE

http://www.sagepublications.com

Additional services and information for *Health Informatics Journal* can be found at:

Open Access: Immediate free access via SAGE Choice

Email Alerts: http://jhi.sagepub.com/cgi/alerts

Subscriptions: http://jhi.sagepub.com/subscriptions

Reprints: http://www.sagepub.com/journalsReprints.nav

Permissions: http://www.sagepub.com/journalsPermissions.nav

Citations: http://jhi.sagepub.com/content/19/4/264.refs.html

>> Version of Record - Nov 19, 2013

OnlineFirst Version of Record - Aug 30, 2013

What is This?

Evaluation of ISO EN 13606 as a result of its implementation in XML

Tony Austin
University College London, UK

Shanghua Sun
St. George's, University of London, UK

Taher Hassan and Dipak Kalra
University College London, UK

Abstract

The five parts of the ISO EN 13606 standard define a means by which health-care records can be exchanged between computer systems. Starting within the European standardisation process, it has now become internationally ratified in ISO. However, ISO standards do not require that a reference implementation be provided, and in order for ISO EN 13606 to deliver the expected benefits, it must be provided not as a document, but as an operational system that is not vendor specific. This article describes the evolution of an Extensible Markup Language (XML) Schema through three iterations, each of which emphasised one particular approach to delivering an executable equivalent to the printed standard. Developing these operational versions and incorporating feedback from users of these demonstrated where implementation compromises were needed and exposed defects in the standard. These are discussed herein. They may require a future technical revision to ISO EN 13606 to resolve the issues identified.

Keywords

Electronic healthcare record, ISO EN 13606, recommendations, standards

Introduction

This article examines the information models and vocabularies published in the ISO EN 13606 standard for electronic health record communication as the basis for defining an Extensible Markup

Corresponding author:
Tony Austin, CHIME, University College London, 4th Floor, Holborn Union Building, London N19 5LW, UK.
Email: tony.austin@ucl.ac.uk

Language (XML) Schema. Such a schema is an important technical artefact to support the adoption of the standard and its use for interoperability. In this research, the process of developing a valid schema has been used as a method for validating the technical correctness of the standard itself. The Introduction section describes the purpose and overall structure of the ISO EN 13606 standard. The Method section describes the iterative methodology adopted to develop the schema and to refine it by correcting for errors in the published standard. The Comparative Schema Results section presents the errors identified in the published standard. The Discussion section discusses the erroneous features of the standard in relation to their intended requirements, the limitations of the study and also its potential value as a contribution to the forthcoming revision to this standard by the European Standardisation Committee (CEN) and International Standards Organisation (ISO).

Paper records do offer benefits for certain types of recording[1] but they suffer from well-known shortcomings in availability and searchability,[2] and the preference for storage is now usually computer-based. However, in many jurisdictions, the legal owner of the record is not the patient but the clinical organisation whose staff members authored the entries. Implicit is that records will be created and stored in one or more physical systems and cannot be readily decommissioned into a single 'silo' of medical information. Even if it were possible to do so, patient fears about ready accessibility may prevent this from happening.[3] This means that while searchability may be locally improved by computerising records, the availability issue may be less easily solved in practice.

Clinical judgement could clearly be improved if it were possible to draw supporting data from other health-care organisations. However, apart from legal and political issues, there are clinical barriers to this. It is not necessarily the case that two clinicians trained in different places and with different backgrounds and experiences will have the same understanding of a clinical concept even if it could be unambiguously expressed in every conceivable natural language simultaneously.

ISO EN 13606

Before even considering legal issues or those of clinical culture, other technical barriers must be removed. To begin with, there must be a widely accepted standard for the representation of clinical information that is itself clear and unambiguous that can be shared among data producers and consumers. It was this realisation that drove the research and development projects[4–7] from which standards through CEN[8] and identically in ISO were created. Such standards then in turn facilitate the interoperability of different vendor products and enable enterprises to adopt a multi-vendor best of breed solution to local information system requirements whilst remaining consistent with the broader vision of communicable and lifelong health-care records.

When the Technical Board of CEN approved the establishment of a Technical Committee for Medical Informatics (TC 251) in May 1990, the Electronic Health-care Record (EHR) was regarded as one of the most important and most urgent areas for the establishment of European standards. Working Group I defined the scope and terms of reference for two work items (WIs): WI 1.6 'Electronic Health-care Record Architecture' (EHRA) and WI 1.8 'EHR Extended Architecture'; these were intended to be the basis for two consecutive project teams.

The Project Team under WI 1.6, PT1-011, developed a pre-standard (known as an 'ENV') 12265 EHRA.[9] ENV 12265 was a foundation standard defining the basic principles upon which EHRs should be based. The Project Teams under WI 1.8 were convened in 1998 and published a

four-part EHR successor standard ENV 13606 in 1999. The Extended Architecture[10] was built on ENV 12265 and defined additional components for describing the structures and semantics in EHRs conforming to a range of requirements to allow the content of a health-care record to be constructed, used, shared and maintained.

A health-care domain model was developed to represent the requirements of clinical practice including professional, ethical, legal and security requirements that must be satisfied by the Extended Architecture, Domain Termlist and Distribution Rules. The Domain Termlist part of this European pre-standard provided a set of measures to support various degrees of interoperability of the EHRs created on different systems or by different teams on the same system.[11] These measures were aimed at enhancing the likelihood that EHR entries could be accessed or communicated. The Distribution Rules specified a set of data objects that represent the rules for defining access privileges to part or whole EHRs, and the means by which security policies and attributes can be defined and implemented.[12] It also defined the principles that should be employed within an audit trail log. Finally, it defined a set of messages to enable the communication of part or whole EHRs in response to a request message or a need to update a mirror repository of a patient's EHR.[13] These messages were specified in a syntax-independent way (i.e. as message information models), but the publication included an informative XML Document Type Definition (DTD), which was found to be helpful to a number of implementers.

In December 2001 CEN TC 251 confirmed a new Task Force, EHRcom, to review ENV 13606 and to propose a revision that could be adopted by CEN as a formal standard ('EN'). The result is intended as a rigorous and durable information architecture for representing the EHR, in order to support the interoperability of systems and components that need to interact with EHR services:

- As discrete systems or as middleware components;
- To access, transfer, add or modify health record entries;
- Through electronic messages or distributed objects;
- Preserving the original clinical meaning intended by the author and
- Reflecting the confidentiality of those data as intended by the author and patient.

As of 2010, all five parts of EN 13606 have been ratified in CEN and ISO. The five parts cover the following:

- An EHR reference model;
- A clinical model interchange specification;
- Reference term lists;
- Security and audit arrangements; and
- An interface specification.

Part 1 of ISO EN 13606 provides key classes that represent how an extract from a clinical system will be delivered to a recipient. These include the following:

EHR_EXTRACT – the top-level container of part or all of the EHR of a single subject of care, for communication between an EHR Provider system and an EHR Recipient.
FOLDER – the high-level organisation within an EHR, dividing it into compartments relating to care provided for a single condition, by a clinical team or institution, or over a fixed time period such as an episode of care. For example, 'Diabetes care', 'Schizophrenia', 'St Mungo's Hospital', 'GP Folder', 'Episodes 2000–2001' and 'Italy'.

COMPOSITION – the set of information committed to one EHR by one agent, as a result of a single clinical encounter or record documentation session. For example, 'Progress note', 'Laboratory test result form', 'Radiology report', 'Referral letter', 'Clinic visit', 'Discharge summary' and 'Diabetes review'.

SECTION – EHR data within a COMPOSITION that belongs under one clinical heading, usually reflecting the flow of information gathering during a clinical encounter, or structured for the benefit of future human readership. For example, 'Reason for encounter', 'Past history', 'Family history', 'Allergy information', 'Subjective symptoms', 'Objective findings', 'Analysis', 'Plan', 'Treatment', 'Diet', 'Posture' and 'Abdominal examination'.

ENTRY – the information recorded in an EHR as a result of one clinical action, one observation, one clinical interpretation or an intention. This is also known as a clinical statement. For example, 'Symptom', 'Observation', 'Test result', 'Prescribed drug', 'Allergy reaction', 'Differential diagnosis', 'Differential white cell count' and 'Blood pressure measurement'.

CLUSTER – the means of organising nested multi-part data structures such as time series, and to represent the columns of a table. For example, 'Audiogram results', 'Electro-encephalogram interpretation' and 'Weighted differential diagnoses'.

ELEMENT – the leaf node of the EHR hierarchy, containing a single data value. For example, 'Systolic blood pressure', 'Heart rate', 'Drug name', 'Symptom' and 'Body weight'.

These primary record building blocks are then aggregated according to Figure 1.

XML

Any standard for clinical information exchange is likely to be large and complex, and the burden on a medical system supplier compelled to deliver an accreditable version quickly is great. Ideally, there would have to be a widely available vendor-neutral implementation of the standard that suppliers could either use directly or refer to when building systems, which would be available without excessive license encumbrance. However, ISO standards do not require such a reference implementation to be made available.

The most common way of formatting documents in word processed documents is to embed control codes (like 'BOLD') within them, which changes the way the document is formatted from that moment on. This is known as the 'specific coding' of a document and emphasises direct changes to a document's presentation. Towards the end of the 1960s, an alternative option called 'generic coding' was proposed in which markers representing the structure of the text were included (such as that it requires an 'AUTHOR'). A viewer is then free to represent these as desired.[14] The latter was the approach used in the development of Standardised Generalised Markup Language (SGML) that was approved by the Office of Official Publications of the European Community and a year later became standard ISO 8879:1986.[15] HyperText Markup Language (HTML) includes both possibilities, using, for example, <b> tags to embolden text and <p> tags to delineate paragraphs.

While SGML is optimised for documents and on the face of it might have been a worthy representation for health-care records themselves, Pitty[16] notes that SGML does not separate functions for capture and display from those of storage and representation, does not capture temporal and semantic relationships and does not in itself standardise the DTD needed to make records sharable.

Derived from SGML, XML[17] was developed by an XML Working Group (originally known as the SGML Editorial Review Board) formed under the auspices of the World Wide Web Consortium
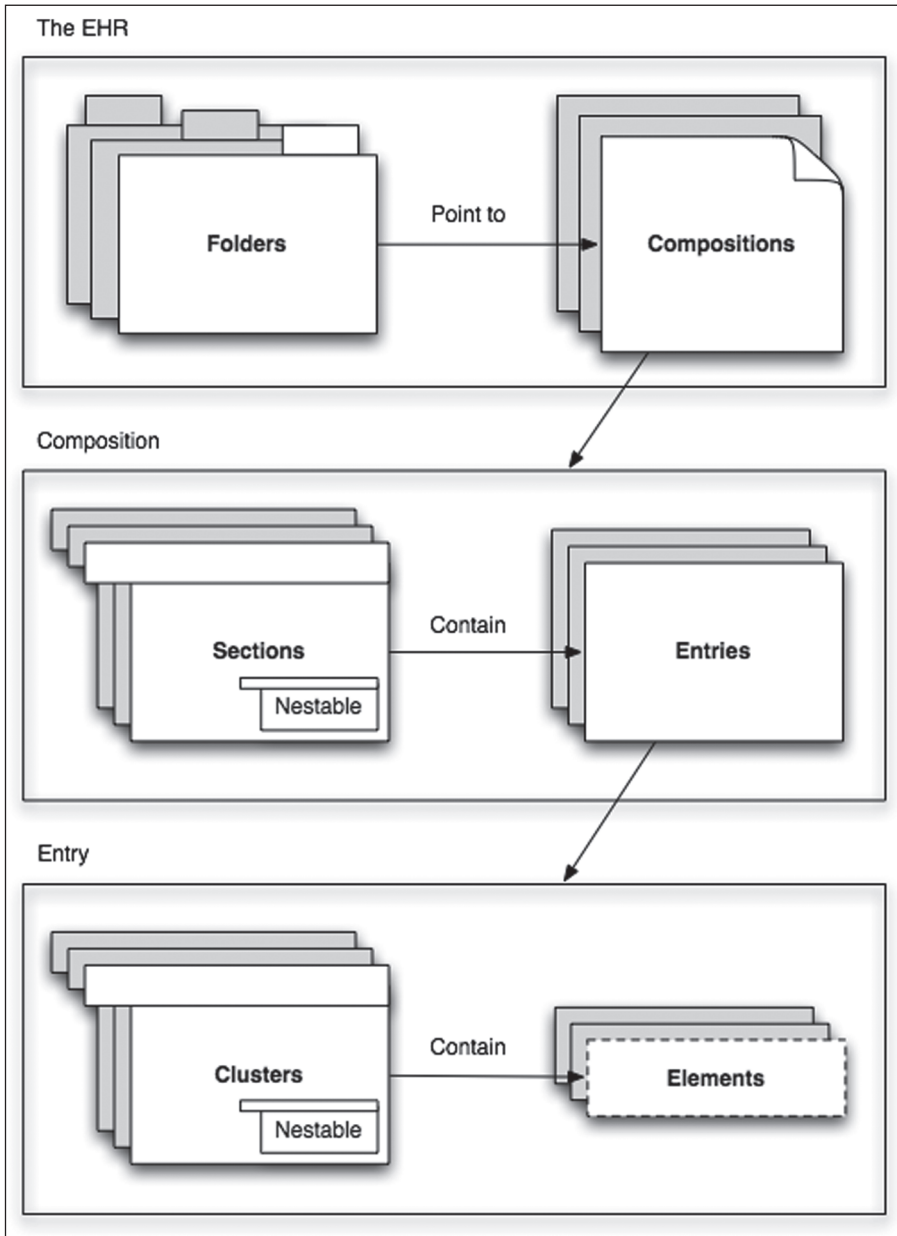
Figure 1. Nesting in the EHR.
EHR: Electronic Health-Care Record.

(W3C) in 1996 and is a simple but flexible text format. It was originally designed for large-scale electronic publishing, and it has an increasingly important role in exchange of a range of data on the Internet. XML is a subset of SGML and emphasises interoperability with SGML and HTML, and ease of implementation.

SGML has nevertheless been studied as a message interchange format in health care.[18] SGML and XML provide a very useful message exchange format by which structured medical records might be sent between systems or institutions. Attempts have already been made to apply XML for specific types of record exchange, for example, Medical Markup Language (MML)[19] and even to encapsulate EN 13606.[20]

Irrespective of the suitability of XML for representing instances of record data, there is an increasing desire to constrain such instances to clinically reasonable statements. Both SGML and XML share a formalism (the DTD) that enables syntactically correct specifications to be created. However, this lacks important features such as data types, and in 1999, the W3C solicited requirements for a new schema language[21] that would provide these using XML itself.

However, although XML Schema can define 'structure, content and semantics of XML documents',[22] there are some specifications that cannot be made by a statically analysing grammar parser alone. These tend to be those where a document inclusion is expected as the result of another in a particular instance document. ISO/IEC 19757 part 3[23] defines a Document Schema Definition Language (DSDL) for rule-based validation known as 'Schematron' that provides another layer of validation within instance documents that ensures semantic relationships between values are maintained.

The lack of a reference implementation of ISO EN 13606 reduces its value for the purpose for which it was designed, namely, the ready unambiguous sharing of clinical data held in multiple computer systems about an individual or population. It also means that while the standard has been widely studied, it is impossible to state that it is computable. Creating a reference implementation, on the other hand, validates the computability of the standard and makes available a technical format in which data may immediately be shared.

The original plan to create an XML implementation came about as a means of receiving data originating from sensors in a UK athletic project.[24] The sensors were worn by athletes and then analysed to optimise their performance, especially in sprinting. Because different collaborating groups generated different types of data but all data related to the athlete, the decision was taken to use a record server built at University College London (UCL) that is compliant with ISO EN 13606.[25] The project lead for the development of the standard, an author of this article, had also received many requests from other countries and vendors for a high-quality XML Schema to support its adoption.

The advantage of using such an XML exchange format is that each collaborator would use the same one, and although potentially complex, they would at least be able to mutually reinforce each other's understanding of it. In addition, at project end, each of the components built by each group would be immediately ready to pair with any other server and not bound only to other components developed within the project.

Instead of the generic XML export, it would have been possible to create a specific one for that particular project. Having a specific document type for a domain significantly lowers the cost of entry for parser/generators and aids comprehensibility. However, specific document descriptions cannot be reused in a wider context – taking the upfront learning cost for the generic approach means that data can be immediately reused everywhere.

Method

Having decided to pursue a generic version of an XML export, but meanwhile understanding the complexity issues that would be faced by implementers from a non-clinical background, it was important not just that the XML Schema be of an acceptable quality and well documented but that

it was easy to create and parse instances of such documents. The brief was therefore to use Java's[26] Java Architecture for XML Binding (JAXB)[27] binding tool to create a set of classes that could read and write documents of the necessary form.

The development clearly needed to begin by building the XML Schema itself and because of the lack of available skills at the outset the authors began with Microsoft's Visual Studio.[28] Unfortunately, this introduced what was presumed were artefacts of the production methodology. For example, the header of the XML included the phrase '{{xmlns:sql="urn:schemas-microsoft-com:mapping-schema"}}', which did not make for a vendor-neutral mapping. While it would have been possible to continue with the approach and eventually remove such references, the authors instead turned to a more elegant (but slower) manual editing procedure using NetBeans. NetBeans[29] is an integrated development environment from Sun Microsystems® (now owned by Oracle®), which can syntax-highlight XML Schema during creation and then subsequently check both that it is well formed and that it is valid. There is also a validity checking tool at the W3C[30] but this is less successful at checking multi-file schemas when they are not available online. In this way, the authors were finally able to create the first XML Schema for ISO EN 13606. However, difficulties with the standard itself described in later sections suggested that the practical utility of such a direct implementation would be limited, and that more than one schemata would be valuable. In the end, the following three were created:

1. The first followed as nearly as possible the published textual version of the standard, changing it only in order that a parser could recognise it as well formed and valid.
2. A second attempt corrected issues with the published form in order that it might better meet its own requirements and included by reference the newly published ISO EN 21090 data types set.
3. Finally, a third schema was produced including a streamlined optimised subset of the provisions in ISO EN 13606 and in ISO EN 21090 that bring with it a much lower burden of implementation.

Schema correctness

From the schema preamble:

> XML Schema equivalent representation of the CEN/ISO 13606 part 1 model for Electronic Healthcare Record Exchange. This version of the schema emphasises (1) fidelity to the 13606-1 standard (even where this leads to sub-optimal XML representation) and (2) maximum use of W3C XML Schema features (even where this increases parser complexity). This Schema definition conforms to the 13606-1 version published February 2007.

There are elementary typographical errors in the standard that make a strict translation impossible to create. Put simply, the standard is not internally consistent and cannot work as written. A list of these simple issues is presented in Table 2. It is in general readily possible to infer the correct meaning of the model presented, and the first version results from making such elementary corrections to derive a consistent schema. However, although consistent and a reasonable equivalent to the printed standard, it is not adequate to represent health-care records because the data types provided in the standard are not themselves adequate. Although a small number of data type classes are provided in ISO EN 13606, the main source of data types intended for representing DATA_VALUEs is published in a CEN Technical Specification – TS 14796.[31]

Table 1. Features and limitations of XML Schema.

Parameterised types	Concrete subclasses of DATA_VALUE [1:6.4.2] used in the high and the low properties of an InterVaL (IVL) [1:6.4.11] should be the same but this cannot be specified in XML Schema and requires additional support
	A InterVaL of TimeStamps (IVL<TS>) [1:6.4.11] in particular should be a restriction on the data types of low and high in IVL to make them TS [1:6.4.7]. This impacts the representation of time_period [1:6.2.3], session_time [1:6.2.6], obs_time [1:6.2.10] and validTime [1:6.3.9]
Possible simplifications	XML Schema obviously copes well with list-type definitions based on primitive or extended types but requires restriction on the definition (rather than an explicit type) to establish further semantics. For example, as a way of establishing Set semantics without an explicit Set type, see http://www.w3.org/TR/xmlschema-0/#specifyingUniqueness. A [0..1] cardinality SET [1:6.5] of anything could therefore be treated as a [0..*] XML list
	The only use of Array in the 13606-1 standard is multimedia-related [1:6.4.6], and it can therefore be replaced with a binary expression
	Coded Simple (CS) value [1:6.4.3] types fully specified in the standard could be represented as simple XML enumerations (with whitespace removed as necessary)
	A general replacement to a consistent <xsd:string> type can be made of miscellaneous strings and TEXT [1:6.4.5] values in the standard, and Encapsulated Data (ED) [1:6.4.6] where the intent is to encode a string
	Mandatory TS [1:6.4.7] timestamps can be an <xsd:dateTime>
	Mandatory BooLean (BL) [1:6.5] markers can be an <xsd:boolean>
Amendments	In the second XML form, a Schematron rule[23] polices that either LINK_NATURE [3:5.7] or LINK_ROLE [3:5.8] is used (defined in EN 13606 part 3) but not both, and that if the latter is used it matches the LINK_NATURE semantically. In the third XML form, both enumerations are collapsed into a single one with values from the previous two

XML: Extensible Markup Language.

Table 2. Correctable oversights in the ISO EN 13606 Standard.

Class structure	Although the EN 13606 primitive package includes many types native to all engineering environments, no such native type will include the full range of possible null_flavours [1:6.4.2]. Indeed, some environments will not even permit primitive types to be NULL. Such examples will require an otherwise redundant redefinition to subclass DATA_VALUE [1:6.4.2] before data can actually be transmitted. The issue could be resolved by adding the null_flavour to the ELEMENT [1:6.2.12] class instead of the DATA_VALUE
	The CS_reason [5:6.4] term list is defined but only one item from the given list is useful because the rest are not related to the clinical exchange. Rather, they describe technical reasons why a request cannot be fulfilled
Attributes	The type of id (identifier) on IDENTIFIED_ENTITY [1:6.3.2] is not given in the printed class description but is included in the class diagram on page 47 of the standard
	At several points in the standard, a graphical nomenclature is adopted where an 'id' in a white box is attached to a class definition. The intent of the standard is that this establishes an association based on the identifier of the associated object (i.e. the II [1:6.4.9]). The textual class descriptions usually make it seem that the actual object type should be used

Table 2. (Continued)

	Set<T> [1:6.5] appears twice in the list of primitives expected to be available in all engineering environments
	Variable naming conventions are very mixed with both underlined_naming and camelCase naming used throughout. Sometimes, this even occurs within a single class
	In general, the obs_time interval [1:6.2.10] should give the beginning and ending of a clinical event or activity. However, the location of this within the ITEM class [1:6.2.10] instead of the ENTRY [1:6.2.9] unnecessarily reduces its utility to time series data only when many clinical events or activities describable in an ENTRY have a start and end
	In part 1, archetype_id is a String in RECORD_COMPONENT [1:6.2.4], but the set of all archetype_ids is SET<II> in EXTRACT_CRITERIA [1:6.2.3]
	Sensitivities can be represented using the CS_SENSITIVITY enumeration of part 4 throughout [4:5.1] (and not Integer)
	The standard mandates the media_type.coding_scheme_name of a multimedia container [1:6.4.6] as 'in ISO/TR 21090:2005, Annex D' and the charset.coding_scheme_name as 'in ISO 21090 Annex C' including the initial 'in' in both cases
	No default values are provided for the category of an ITEM [1:6.2.10], the status of an ACT [3:5.6] or the ROLE or NATURE of a LINK in part 3 [3:5.7, 3:5.8]

Schema corrections and type expansion

From the schema preamble:

> XML Schema equivalent representation of the CEN/ISO 13606 part 1 model for Electronic Healthcare Record Exchange. This Schema definition conforms to the 13606-1 version published in February 2007 […].

> A profile of certain data types is included in 13606 part 1 so that the meaning of types used in the exchange model is clear. However, the included type set would not be enough to model an actual record comprehensively. In this version, the 13606 types are removed, and the script imports the new ISO 21090 type set for general use instead. This also impacts the demographics model which is now significantly reduced from the published documentation.

The second XML Schema version corrected all errors in the printed ISO EN 13606 document presented in Table 3 (although not necessarily in the most efficient way) and overcomes the lack of data type specification by referencing a new comprehensive type set provided by ISO EN 21090. While the first schema is the most 'standards-compliant' version of the three schemas presented, this is the most 'correct' version, adhering most faithfully to the intent if not the letter of the document. However, the second XML Schema version suffers from a high level of verbosity. Although XML itself is a verbose representation formalism, its use here is anything but a human-readable computational form. There are several factors involved, as follows:

- The imported ISO EN 21090 type set is large and in need of reduction.[32]
- ISO EN 13606 chooses ISO-approved types where less verbose ones would be sufficient (e.g. the six-part Instance Identifier (II) class instead of the single-string xs:anyURI (Universal Resource Identifier)).
- ISO EN 13606 envisages not just a client and a server exchange, but also a third-party intermediary whose credentials are unknown.

Schema usability

The third and final version of the XML Schema simplifies the second version to remove much of this unnecessary complexity from the record and data types models. This improves the readability of XML documents resulting from the schema and simplifies the parser necessary for storing information items in the columns of a relational database. The final schema is therefore also possible to offer in a relational script (and this has been done for the PostgreSQL™33 database). PostgreSQL does offer an explicit XML data type that could be used to store a received document (of any detail level). However, without structure simplification, this simply increases query complexity to an unacceptable level instead.

Comparative schema results

Tables 1-3 enumerate those issues noted during the development of the ISO EN 13606 XML versions. These prevented the developers from using the printed specifications as-is. The issues are broken down by type. Note that the sections that follow draw on all three XML versions described in the Method section. Within the tables and following them in the Discussion section are some suggestions for improvement that are aimed at streamlining the ISO EN 13606 classes rather than correcting actual errors. The standards are referenced in the tables in square brackets, with the part followed by the section number, for example, [1:6.4.2] for part 1, section 6.4.2.

Table 3. Errors in the ISO EN 13606 Standard.

Class structure	An ELEMENT [1:6.2.12] can have zero-or-one DATA_VALUE [1:6.4.2] instances. However, the DATA_VALUE is also the container for null_flavour, which makes it structurally possible for an ELEMENT to exist with no value and no null_flavour given. The ELEMENT class does not describe a countervailing invariant
	'null_flavour' [1:6.4.2] is defined as a CS class [1:6.4.3] which means that it can itself be a null_flavour
	The REQUEST in part 4 [4:6.4] defines the properties of a requestor of a policy. Coded Simple FUNCtional ROLE (CS_FUNC_ROLE) [4:6.4] therein refers to the FUNCTIONAL_ROLE [1:6.2.15] class, despite it not being named as such and not described as a CS [1:6.4.3]. CS_SETTING [4:6.4] is not given at all
Attributes	There are disclosure implications of including all policy_ids [1:6.2.4] when providing an EHR_EXTRACT [1:6.2.2]. The recipient then knows the policies governing all other recipients
	In many places in the standard, an invariant is specified on the 'coding_scheme_name' of the CS [1:6.4.3] or Coded Value (CV) [1:6.4.4] data type. This is intended to ensure that the coding scheme used to provide the information is a particular one. However, the actual variable name used in the CS and CV classes is 'codingSchemeName' and so as printed, the restriction cannot be enforced
	On several occasions in the standard, a code is described as having a CS [1:6.4.3] type but only two of the four necessary parts of the CS are provided by the documentation (usually a value and a human-readable portion). The XML third form jettisons the CS type entirely and uses XML enumerations for the standard's term lists

XML: Extensible Markup Language.

Discussion

This research has demonstrated that despite substantial peer review through international ballot cycles, technical errors can remain undetected within a complex standard. The generation of an implementable specification such as an XML Schema could provide a means for technical verification of a specification before its final publication. Second, the discipline of developing such an implementable specification can highlight areas where a chosen modelling construct might not be the optimal way to meet an intended requirement.

Copyright issues prevented the derived XML implementation including explanatory extracts from the standard document itself. Instead, the XML includes a short phrase outlining the purpose of the inclusion and then exhaustively documents the differences between the published and implemented forms. These appear as <documentation> tags.

The original methodology for describing the class structure of the ISO EN 13606 reference model began with Universal Modelling Language (UML)[34] but this was found to lack certain features that were required for the standard (e.g. where a pointer to a class instance was required). The diagramming tool MagicDraw[35] was used to create the diagrams, and it was possible to do so with the missing UML features included. However, this was not related to the engineering environment used to validate the model.[36] The latter draws heavily on so-called invariants as a construct for ensuring correct behaviour of programs. Needless to say, none of these are directly representable in XML Schema, although in the second schema a 'Schematron'[23] rule was used to ensure that the Link Role and Link Nature paired as dictated by the standard.

Although now standardised, there remain questions about some parts of the ISO EN 13606 standard. The final part, for example, describes a logical set of interfaces that a record server would be expected to offer but does not describe a technical solution by which they might be made available. The proposed interface set is in any case vanishingly minimal and would benefit from expansion. The current study did not address part 5 but had it done so, a Simple Object Access Protocol (SOAP)[37] solution might have been appropriate, or even one based on the Resource Description Framework (RDF)[38] and SPARQL Protocol and RDF Query Language (recursively abbreviated as SPARQL).[39]

Part 2 of the standard is also not addressed. This part of the standard specifies a formal representation for clinical archetypes, and so would not be implemented through the same components as those implementing parts 1, 3 and 4. It may be considered for similar examination in the future. The remainder of the discussion section presents changes to the standard the authors feel are justifiable based on the effort to implement it in XML.

Demographics class changes

Some collaborators felt it necessary to split up the early XML versions into multiple files, and the third version is architected to be separable if desired (although this is not recommended for interoperability reasons). Most users preferred to use the ISO data types standard even while still in draft form rather than the ISO EN 13606 data types implied by the first XML Schema form. Others have attempted to co-opt types from the openEHR foundation into a hybrid standard. The ISO types standard defines rather a lot of types, many of which might be awkward to conveniently represent, for example, in a relational database. The third form provides a much more focussed set and relies more heavily on the knowledge model to provide structure. For example, a RATIO class is really two numbers with a defined semantic relationship and is unnecessary to represent uniquely.

Even more awkwardly, some participants tend to remove the demographic components as well. Demographics are clearly an important part of a patient record, and it is important that records can

be interleaved by reliable identifying demographics. The ISO EN 13606 demographics package was developed to solve three scenarios: first, minimum identification to permit demographic matching between two systems; second, a rich enough descriptor set to populate a recipient's demographic system with enough to identify and contact persons or organisations, and third, for the whole thing to be optional if the exchange is occurring inside a shared demographics realm. An exchange standard is particularly concerned with the first and third of these and less so with the second – in fact, a preoccupation with population of a demographics repository with information not really required for matching causes a certain amount of overlap with other more focussed standards. The XML Schema third form takes matching as its one and only reason for existing (which will obviously often imply matching identifiers as well). More specific simplifications are also possible, and these are detailed in the following sections.

Class Subject Of Care Personl dentification. The extractID can be represented as an xsd:ID because it is something that other things in an XML document refer to (although for legacy compatibility reasons, XML Schema specifies that this must be stored as an attribute and not an XML entity). The ISO EN 13606 identifier can simply be a list of anyURI representing the identifiers. Both birthTime and deceasedTime are reasonable xsd:dateTime instances and birthOrderNumber an xsd:positiveInteger. ISO 5218[40] already defines an administrativeGenderCode, and a suitable ordinal can be taken from there (the authors suggest UNKNOWN, MALE, FEMALE and NOT_APPLICABLE).

ISO 22220[41] also offers 'Mother's original family name' and 'Country of birth' as sensible possible matching criteria. However, the first has possible confidentiality implications and the second can subsumed into a 'place of birth' address.

Class Organisation. There is already an ISO standard (6523)[42] for the representation of organisations that uses the following:

- Four-digit International Code Designator (ICD) value that uniquely identifies the authority that issued the code to the organisation;
- Organisation code, up to a maximum of 14 characters (A–Z, 0–9, space or hyphen) and
- Organisation name, up to a maximum of 250 characters.

The character '/' is used to separate values during transmission, and this makes it look conveniently like an xsd:anyURI. One advantage of using anyURI is that if a non-standard String is presented, it might still be parsable by a recipient system.

Class Identified Healthcare Professional. There are apparently no international standards that facilitate the identification of health-care professionals. ISO EN 13606 provides for the identification of roles held by a health-care professional but it is not clear why this needs another set of identifiers as well as those inherited from IdentifiedEntity. First, the extract needs only to reference a health-care professional repository populated from an official source (that could include valid specialities and so forth). But in any case, the extract should not state whether someone was 'junior' or whether they are a 'diabetologist' because these are non-durable facts and will change as the source updates its information. It is arguably important to know the role a professional played in producing the present record extract but this could be collapsed to a String identifying the role (such as the logged-in 'Role Name'). That perhaps makes the HealthcareProfessionalRole contain one role String and the (xsd:IDREF) link to Organisation. This would be durable even if a globally recognised repository of Role Names ever became available.

Table 4. Exmaple Software Or Device information.

deviceManufacturerModelName	Apple 1.67 GHz PowerBook5, 8
deviceSerialNumber	WXXXXMXSXX
owningOrganisation	UCL
description	Fred Blogg's Laptop
softwareManufacturerModelName	Apple Pages '08 3.0.3

UCL: University College London.

Class Software Or Device. There is already an ISO standard (19770-2)[43] governing Software Identification Tags but it appears that 19770-2 is a heavyweight solution requiring registration, and in contrast, ISO EN 13606 seems satisfied with a lightweight approach that describes classes of device, classes of software, specific devices or specific software. The authors would remove 'version' because the 'manufacturerModelName' description also includes the possibility of a version. The attributes would then become as in Table 4 (example included).

A class of device only needs deviceManufacturerModelName, and a class of software needs only softwareManufacturerModelName. A specific device would also require deviceSerialNumber (with owningOrganisation and description present to narrow the physical search space), and finally, specific software is installed on a particular device so it needs deviceManufacturerModelName, deviceSerialNumber and softwareManufacturerModelName together.

Record class changes

Much of the methodology and attribute provision in ISO EN 13606 is based on an assumption that an interaction will not be based on a client–server exchange. In particular, it provides for the possibility that a third party with separate credentials will have become involved who then in turn is responsible for ensuring adequate credentials of any party that it hands the data onto. For that reason, it provides for a lot of attributes in the EHRExtract and ExtractCriteria classes related to the situation of the original data request that would be redundant for a client–server exchange because the requestor would already know what it requested. Counter-proposing that the XML Schema methodology be entirely based around a client–server methodology allows for most of the attributes of EHRExtract and the entire ExtractCriteria class to be removed. Furthermore, assuming that identifying data (such as the subject of care identifier) appears in the Composition class returned means that the EHRExtract class can also be removed, and that the response to a request is simply a list comprising the Compositions, Folders and Demographic extracts relevant to the response.

The ISO EN 13606 standard does not take a strongly Composition-centric approach to modelling. However, in practice, this is necessary. The medico-legal purpose of the Composition is to act as the container for a contribution to the record (which also seems to make 'contributionID' redundant). In reverse, this means that in order for an extraction to be medico-legally complete, it must include the enclosing Composition. In particular, although the Record Component is declared as having a sensitivity attribute, it is the Composition where the declared sensitivity of the contribution as a whole must be stored. This is because it is not clinically safe for parts of a contribution to be omitted because they have higher sensitivity than other parts. Although it would be clear to future investigators which parts of a record were not available to a user (assuming versioning included versions of the sensitivity attribute), any routine clinical decision based upon data that were not reliably complete would always be questionable.

There is a suspicious asymmetry about the modelling of AUDIT_INFO. The intent is that newly developed systems concentrate their audit information in the Composition but that older systems may store audit information lower in the hierarchy and must be catered for. In other words, good newly developed systems impose Compositions and Entries on their data and store their commitment information in the Composition, while bad old systems still impose Compositions and Entries and so on but spew their committal data throughout. It is inefficient to have, for example, a table join for each AUDIT_INFO × RECORD_COMPONENT when in fact the feature will only be of use in a limited number of circumstances outside the Composition itself. The authors would definitively fold AUDIT_INFO into the Composition and let feeder system vendors decide how to describe authors of sub-components (perhaps via the other_participants attribute since imposing an attribute methodology is no different to imposing one for aggregation). To facilitate this move, it might be desirable to differentiate between the creator (the logged-in user of the present system who performed an import) and the committer (the logged-in user of an original system who created the data in the first place).

The authors would merge the FunctionalRole and RelatedParty classes. The merged class would consist of optional identifiers for HealthcareProfessional, Organisation and Device, along with the mode (from FunctionalRole) and a relationship. In general, information might need to be recorded where the 'participant' disavows his or her role for confidentiality reasons. For example, a biological mother who gave her daughter up for adoption may have started to exhibit signs of a hereditary disease but if challenged would deny having had a child. Recording of the possibility of that disease in the child's record could not formally identify the biological mother (via the performer attribute) and would be better listed simply as a relationship. Conversely, the subject of information might well be about 'the dialysis machine that didn't work', and therefore, the subjectOfInformation attribute should permit actual identifying data too.

The authors would remove sub-folders. Folders now seem to work like social networking tags on well-known websites such as Flickr[44] and only link to identifiers rather than representing some sort of genuine containment as they did in previous iterations of the standard. Having sub-folders only serves to make merging two records more complex because it is not clear how a target record should aggregate migrated folders if they started with a different aggregation to the source.

Record attribute changes. The 'meaning' attribute of the Record Component is a redundant attribute if the name can be a coded value. 'parent_ref', whose intent is to imply that a component has been copied from another place in the record, might be better as a Link with a new nature attribute of 'copied to' or 'copied from'. 'policy_id' is also redundant. It caters to a situation where a recipient server is expected to abide by the same policy considerations as a source server, and the standard assumes that the identifier will not be passed along to any user of the recipient server. Instead, the recipient server de-references the policy identifier separately from the originating server to determine what the operational access control should be. Obviously, the recipient server could supply the record component identifier instead and simply ask what policies govern it.

Attestations should also be removed. Attesting data is very difficult, and ISO EN 13606 misses some key information that would be needed to accomplish it. To do so requires a canonicalised form of the structure being attested but because ISO EN 13606 provides no reference serialisation format itself (in turn because the data types were not fixed by the standard), it is not possible to create an example of this canonicalised form. A digital signature could be applied to a part of a component such as the identifier or even to the structure using the XML Schema described herein as the canonicalised form. But it would be better for the feature to be reserved for the attestation of a whole contribution only or removed altogether since a contribution itself already implies the attestation of its truth by the contributor.

The authors would use a simple Boolean value to indicate emphasis. Laboratory experts indicated that flagging a result as abnormally high or abnormally low is common in laboratory systems, and the ISO EN 13606 standard represents those kinds of values faithfully. However, the abnormal nature of a laboratory result is a domain-specific piece of knowledge and should really be represented as a construct in the knowledge layer.

Conclusion

The authors have successfully implemented much of the ISO EN 13606 standard as XML Schema with different underlying purposes. XML document extracts representing parts of a record have been successfully transmitted and received using the last of these. Now several groups worldwide have begun experimenting with the scripts as a means to actually exchange health-care records between systems. These include at least two national endeavours in Sweden and the United Kingdom.

The research reported here has identified specific technical errors and some suggestions for improvement to the information model in part 1 of ISO EN 13606. These errors were not identified despite multiple ballot cycles and widespread international peer review of the published information. This suggests that a more technical, tools based, validation stage is needed in order to ensure that a published standard is technically correct and faithfully implementable. The authors would recommend that such a stage is formally introduced into the standards development life cycle prior to a final version being balloted. The ISO EN 13606 standard is now due for its periodic technical review by the CEN and ISO, and the results of this article will feed into that review process.

The study is a strictly technical one based on extensive implementation experience among the contributing teams. Further implementation experience may yet bring further simplifications to light, and these will be folded into further iterations of the schema. However, not all influences on an international standard are technical, despite the outcome being a technical specification. The study does not address existing organisational or national political requirements that act against the general desire to simplify.

The final XML Schema produced as a result of this study has been made publicly available at http://www.ehr.chime.ucl.ac.uk/code/schema_3.0.3/

Funding

References

1. Tange HJ. Consultation of medical narratives in the electronic medical record. *Methods Inf Med* 1999; 38(4–5): 289–293.
2. Kalra D. *Clinical foundations and information architecture for the implementation of a federated health record service*. PhD Thesis, University of London, UK, 2002, http://eprints.ucl.ac.uk/1584/ (accessed March 2012).
3. TheBigOptOut.org. About the campaign, http://www.nhsconfidentiality.org/?page id=3 (accessed January 2012).
4. Lloyd D, Kalra D, Beale T, et al. (eds). *The GEHR final architecture description*. The Good European Health Record Project: Deliverable 19, 1995. Brussels: European Commission, 250 pp., https://www.ucl.ac.uk/chime/research/gehr/deliverable-19.pdf (accessed July 2010).
5. Grimson J, Grimson W, Berry D, et al. A CORBA-based integration of distributed electronic healthcare records using the synapses approach. *IEEE Trans Inf Technol Biomed* 1998; 2(3): 124–138.

6. Sottile PA, Ferrara FM, Grimson W, et al. The holistic healthcare information system. In: *Toward an electronic health record Europe '99*, London, UK, 14–17 November 1999, Boston, USA: Medical Records Institute, pp. 259–266.

7. Dixon R, Grubb PA, Lloyd D, et al. *Consolidated list of requirements*. EHCR Support Action Deliverable 1.4, 2001. Brussels: European Commission DGXIII, 59 pp., http://www.chime.ucl.ac.uk/HealthI/EHCR-SupA/del1-4v13.PDF (accessed July 2010).

8. CEN Technical Committee 251. http://www.cen.eu/CEN/Sectors/TechnicalCommitteesWorkshops/CENTechnicalCommittees/Pages/Standards.aspx?param=6232&;title=CEN%2FTC+251 (accessed March 2012).

9. Hurlen P (ed.). ENV 12265:1995. Electronic Healthcare Record Architecture, 1995. Brussels CEN Technical Committee/251.

10. Kay S and Marley T (eds). ENV 13606:1999. EHCR Communications: Part 1 Electronic Healthcare Record Architecture, 1999. Brussels CEN Technical Committee/251.

11. Rossi Mori A, Kalra D, Rodrigues JM, et al. (eds). *Project team 1-027. ENV 13606: 1999 EHCR communications: part 2 domain termlist*, 1999. CEN Technical Committee/251.

12. Hopkins R Grøtan T, Nystadnes T, et al. (eds). *ENV 13606:1999. EHCR communications: part 3 distribution rules*, 1999. Brussels CEN Technical Committee/251.

13. Markwell D (ed). *Project team 1-029. ENV 13606: EHCR communications: part 4 messages for exchange of information*, 1999. Brussels CEN Technical Committee/251.

14. Turner L. A brief history of the development of SGML. *SGML User's Group*, http://xml.coverpages.org/sgmlhist0.html (1990, accessed March 2012).

15. ISO 8879:1986. Information processing – text and office systems – Standard Generalized Markup Language (SGML).

16. Pitty D, Gordon C, Reeves P, et al. The place of SGML and HTML in building electronic patient records. In: *Medical informatics Europe '97 Thessaloniki, Greece, 25th - 29th May, 1997*, pp. 329–333. Amsterdam, Netherlands: IOS Press.

17. Bray T, Paoli J, Sperberg-McQueen CM, et al. *Extensible markup language (XML) 1.0. World Wide Web Consortium (W3C)*. 2000, 60 pp., http://www.w3.org/TR/2000/REC-xml-20001006 (accessed March 2012).

18. Dolin RH, Alschuler L, Bray T, et al. SGML as a message interchange format in healthcare. In: *Proceedings/AMIA annual fall symposium*, October 25–29, 1997, AMIA Nashville Tennessee, pp. 635–639.

19. Takeda H, Matsumura Y, Kuwata S, et al. Architecture for networked electronic patient record systems. *Int J Med Inform* 2000; 60: 161–167.

20. The LinkEHR Normalization Platform. http://www.linkehr.com/ (accessed September 2010).

21. W3C. XML schema requirements, February 1999, http://www.w3.org/TR/1999/NOTE-xml-schema-req-19990215 (accessed May 2012).

22. W3C. XML schema, http://www.w3.org/XML/Schema (accessed May 2012).

23. ISO/IEC 19757-3:2006. Information technology – Document Schema Definition Languages (DSDL) – part 3: rule-based validation – Schematron.

24. SESAME. SESAME: SEnsing for Sport And Managed Exercise. http://www.sesame.ucl.ac.uk/ (accessed September 2010).

25. Austin T, Lim YS, Nguyen D, et al. Design of an electronic healthcare record server based on part 1 of ISO EN 13606. *J Healthc Eng* 2011; 2(2): 143–160.

26. Sun Microsystems. Java 2 Platform Standard Edition (J2SE), http://www.oracle.com/technetwork/java/javase/overview/index.html (accessed March 2012).

27. Sun Microsystems. Project JAXB Reference Implementation, http://jaxb.java.net/ (accessed March 2010).

28. Microsoft. Microsoft Visual Studio, http://www.microsoft.com/visualstudio (accessed March 2012).

29. Oracle. NetBeans, http://netbeans.org (accessed March 2012).

30. The World Wide Web Consortium. XML schema checking service, http://www.w3.org/2001/03/webdata/xsv (2007, accessed March 2012).

31. DD CEN/TS 14796:2004. Health informatics. Data types.
32. Sun S, Austin T and Kalra D. A data types profile suitable for use with ISO EN 13606. *J Med Syst*, 36(6): 3621–3635. doi: 10.1007/s10916-012-9837-z.
33. PostgreSQL Global Development Group. About PostgreSQL, http://www.postgresql.org/ (accessed March 2012).
34. Object Management Group, Inc. Unified Modeling Language, http://www.uml.org/ (accessed March 2012).
35. No Magic, Inc. Introducing MagicDraw, https://www.magicdraw.com/what (accessed March 2012).
36. Meyer B. *Eiffel: the language (Prentice Hall object-oriented series)*. USA: Prentice Hall, 1991, ISBN: 0132479257.
37. W3C. SOAP version 1.2 part 1: messaging framework (second edition). W3C Recommendation, 2007, http://www.w3.org/TR/soap12-part1/ (accessed March 2012).
38. W3C. RDF working group. Resource Description Framework (RDF), http://www.w3.org/RDF/ (2004, accessed March 2012).
39. W3C. SPARQL Query Language for RDF. W3C Recommendation, 2008, http://www.w3.org/TR/rdf-sparql-query (accessed March 2012).
40. ISO/IEC 5218:2004. Information technology – codes for the representation of human sexes.
41. ISO/TS 22220:2009. Health informatics – identification of subjects of health care.
42. ISO/IEC 6523-1:1998. Information technology – structure for the identification of organizations and organization parts – part 1: identification of organization identification schemes.
43. ISO/IEC 19770-2:2009. Information technology – software asset management – part 2: software identification tag.
44. Yahoo! Inc. Flickr from YAHOO! http://www.flickr.com/ (accessed March 2012).