

Gou Masuda, Norihiro Sakamoto, et. al.	An HL7 Verision3 Based Regional Diabetes Patient Record Project Developed in Japan	Journal of Korean Society of Medical Informatics	9 (Suppl. 2)	s284-s288	2003年
坂本憲広, 公文 敦	電子カルテにおける医 療情報の証拠能力	医療情報学	24 (Suppl.)	54-56	2004年
星本弘之, 増田 剛, 坂本憲広	HL7バージョン3の病院 情報システムへの適用	医療情報学	24 (Suppl.)	390-391	2004年
増田剛, 坂本憲広	HL7バージョン3処方メ ッセージの開発	医療情報学	24 (Suppl.)	392-393	2004年
森山和好, 東重 行, 星本弘之, 坂 本憲広	HL7Ver3による臨床検査 システム接続	医療情報学	24 (Suppl.)	394-395	2004年
広井嘉栄, 増田 剛, 星本弘之, 坂 本憲広	薬品マスタのHL7バージ ョン3による記述	医療情報学	24 (Suppl.)	396-397	2004年
山本さつき, 前田 英一, 星本弘之, 坂本憲広	電子化に向けた糖尿病 カルテの分析	医療情報学	24 (Suppl.)	902-903	2004年
西脇清行, 増田 剛, 坂本憲広	PKIを用いた広域対応の 臨床試験情報システム の構築	医療情報学	24 (Suppl.)	978-979	2004年
増田剛, 広井嘉 栄, 坂本憲広	保健医療情報標準化規 格に基づく疾患関連遺 伝子解析研究のための データ収集基盤	医療情報学	24 (Suppl.)	1206-1207	2004年
Norihiro Sakamoto	Towards the Construction of the Information Infrastructure for Genome Medicine	Lecture Notes in Computer Science Series	in press	in press	2005年

Hiroyuki Hoshimoto, Norihiro Sakamoto	An Implementation of A Communication Interface between a Hospital Information System and a Laboratory Information System Based on the HL7 Version 3	International Journal of Medical Informatics	投稿中	投稿中	2005年
Gou Masuda, Norihiro Sakamoto	Design and Implementation of a Software Library to Support Healthcare Information Exchange Standards	Journal of the American Medical Informatics Association	投稿中	投稿中	2005年

HL7 Version 3メッセージを用いた保健医療情報システムの開発支援

増田 剛¹⁾ 坂本 憲広²⁾ 山本 隆一¹⁾

大阪医科大学病院医療情報部¹⁾ 神戸大学医学部附属病院医療情報部²⁾

Development Support for Healthcare Information Systems Using HL7 Version 3

Gou Masuda¹⁾ Norihiro Sakamoto²⁾ Ryuichi Yamamoto¹⁾

Division of Medical Informatics, Osaka Medical College¹⁾

Department of Medical Informatics, Kobe University Hospital²⁾

Abstract: HL7 Version 3(HL7 V3) is a standard for exchanging messages among information systems in healthcare domain. It is based on a consistent message development methodology and a shared information model that is the source for the data content of all HL7 V3 messages. When using HL7 V3 messages, developers are required to find appropriate messages that meet their requirements in the HL7 V3 standard. However, it is a hard task for the developers because HL7 V3 defines over 1,300 message interactions and about 300 messages. An inappropriate application of messages leads to less interoperability. In this paper, we propose two methods to navigate developers to appropriate HL7 V3 messages. One is navigation from the point of requirements of healthcare information systems. The other is one from the point of data contents. We develop a prototype tool and evaluate our proposed methods.

Keywords: Standardization, HL7 Version 3, Development Support

1. はじめに

HL7 Version 3(以下HL7 V3)は、一貫したメッセージ開発方法論と共通の参照情報モデルに基づいて開発された、保健医療情報システムにおける情報交換のためのメッセージ規約である。HL7 V3メッセージを用いた保健医療情報システム開発においては、開発者はシステムに対する要求からそこに発生する情報とその流れを分析し、適切なHL7メッセージを探し出し適用する必要がある。しかし、HL7 V3は現在開発途中でありながら、1300を越えるメッセージ交換の場面(インタラクション)に関して約300のメッセージが定義されている。この膨大なインタラクションの中から自分たちの要求に適合するインタラクションを特定し、適用すべきメッセージを見つけることは容易ではない。もし適切でないメッセージを適用すれば、たとえ標準規約を用いていたとしても、相互運用性を損なうことになり兼ねない。そこで本研究では、保健医療情報システムの開発において、適用すべきHL7メッセージを導く、HL7メッセージの適用支援の方法を提案する。

2. 方法

保健医療情報システムの開発においてHL7メッセージを適用する場合、少なくとも次の二つの状況が考えられる。一つ目の状況は、システムに対する要求が既に明確になっている場合である。この場合、その要求に対応するアプリケーションの役割を決定することで、適用すべきHL7メッセージを導くことができる(システムの要求に基づく提示)。二つ目の状況はシステム間で交換すべきデータが明確に

なっている場合である。この状況では、それらのデータを扱うことのできるHL7メッセージを、そのメッセージが含む要素を手掛かりとすることで導くことができる(データ項目に基づく提示)。そこで、この二つの観点からメッセージの提示を検討する。

2.1 HL7 V3 メッセージ

HL7 V3メッセージは、医療分野において、ある役割を持ったアプリケーション間のメッセージ交換の文脈(インタラクション)において定義される。一つのインタラクションは、(1)送信アプリケーションロール、(2)受信アプリケーションロール、(3)トリガイベント、(4)メッセージタイプ、(5)受信者責任、から構成される。送信・受信アプリケーションロールは、メッセージ交換を行なうアプリケーションの種類を特定する。トリガイベントは、そのメッセージ交換を発動させるきっかけであり、メッセージタイプは、交換されるメッセージの内容そのものに対応する。

2.2 システムの要求に基づく提示

システムの要求からメッセージを導出するためには、その要求とそれを実現することのできるアプリケーションロールとを対応付けることが必要である。アプリケーションロールが決まることで、そのアプリケーションロールを送信者または受信者にもつインタラクションが決定される。HL7 V3では、同じメッセージタイプが異なるインタラクションで使用される場合があるため、システムに対する要求とメッセージを直接対応付けるよりは、要求からインタラクションに対応付けることが望ましいと考える。インタラクションが決まればそのインタラクションで交換されるメッ

セージタイプは一意に定まる。

HL7 V3において、アプリケーションロール名は「能力」や「ステレオタイプ」といったあらかじめ定義された複数の構成要素の組み合わせとして表現される。例えば、検査オーダー依頼者は、"Observation Order Global Placer"というアプリケーションロール名で表される。また、処方オーダー中止の実施者は"P harmacy Administration Order Suspension Fulfiller"として表される。そこで本研究では、ユーザーがこれらの構成要素を段階的に選択していくことで、要求とアプリケーションロールとの対応付けを与える。

2.3 データ項目に基づく提示

もう一つのアプローチとして、実システムの情報交換において送るべき内容が既に決定している場合に、その内容から適用すべきメッセージを提示する方法を検討する。本研究では、メッセージを構成する共通メッセージ要素型(Common Message Element Type:CMET)に着目する。CMETは、複数のメッセージに出現する再利用可能な共通部品であり、例えば患者や医師、検体といった、一連の関連するメッセージ要素の集まりを定義する。そのためCMETはメッセージにおいて特徴的な要素とみなすことができる。そこで、実際に交換されるデータ内容をCMETと対応付けることによって、そのデータ内容に関係するメッセージの導出を試みる。データ内容の表現には、「電子保存された診療録情報の交換のためのデータ項目セット」(以下データ項目セット)²⁾を利用し、CMETとデータ項目セットの各項目とを対応付けた。例えば、患者を表す"CMET: R Patient"には、データ項目セット第1章の「患者.ID」や「患者.氏名」といった項目が対応する。これにより、データ項目セットの項目名から、それらの項目を含むCMETを導くことができ、さらに、それらのCMETを含むメッセージを導出することが可能となる。

3. 結果

以上述べた方法に基づき、HL7 V3メッセージ提示ツールを試作した。メッセージデータベースは、今後の版改訂にも容易に対応できるように、HL7から提供されるモデルリポジトリをそのまま使用することが望ましい。しかし、現在のところV3の全メッセージを含むモデルリポジトリは提供されていないため、今回の試作には、HL7のオーダー・検査技術委員会から提供される検査メッセージのみを含むモデルリポジトリを使用した。そのため、評価はHL7 V3委員会用投票パッケージ第3版の検査メッセージを対象に行なった。実行結果例を図1に示す。

4. 考察

まず要求に基づくメッセージ提示については、今回対象とした検査メッセージに関しては、要求とアプリケーションロールとの対応付けが行ない易い領域であったため、適切なメッセージを提示することが可能であった。今後、検査以外の領域のメッセージに対する評価が必要である。一方、データ項目からのメッセージ提示に関しては、本来CMETに対応付けることのできないデータ項目が多く存在するため、それらのデータ項目が指定された場合、関連する

メッセージを導くことができなかった。逆に、「患者.ID」といった、様々なメッセージの中で利用されているCMETに対応するデータ項目を指定すると、多くのメッセージが提示される結果となった。これは、今回の対応付けでは、CMETがどのような文脈の中で用いられるかまでは考慮せずに、単純にCMETを含むメッセージを提示していることが原因である。今後は、より適切なメッセージを提示できるように、CMETだけではなく、メッセージの中心要素となるRIMクラスの属性や、メッセージを含む用語(ボキャブラリ)との対応も考慮することが必要である。

5. おわりに

本稿では、HL7 V3メッセージを用いた開発支援のための、メッセージ提示方法について述べた。今後は、メッセージの提示だけではなく、ユースケースやメッセージの実例といった情報も含めた統合的な開発支援システムへの発展を検討する。

参考文献

- [1] HL7 Version 3 Standard, Health Level Seven Inc., Available at <http://www.hl7.org/>, 2002.
- [2] 電子保存された診療録情報の交換のためのデータ項目セットの作成、(財)医療情報システム開発センター, 2002.

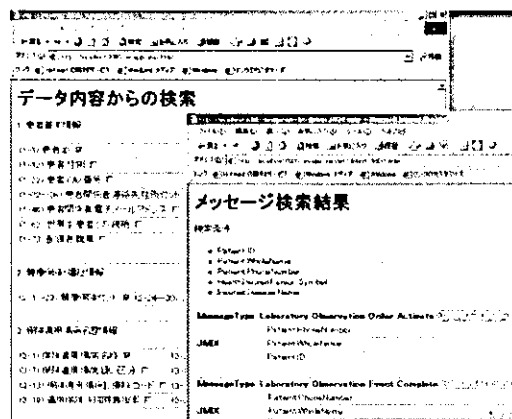


図1 メッセージ提示ツールの実行例

HL7を用いたシステム構築の現状と今後の展開

坂本 憲広

神戸大学 医学部 附属病院 医療情報部

The current state and the future of healthcare information system development by using HL7

Norihiro SAKAMOTO

Department of Medical Informatics, Kobe University Hospita

Abstract: HL7 (Health Level Seven) is internationally regarded as the standard for communication and messaging of healthcare information. In Japan, developments of electronic healthcare records based on HL7 are becoming more popular. However, there are still such cases that system developers make their local rules and formats different from HL7. One of the reasons why the HL7 wasn't employed can be that existing HL7 messages are not systematically archived, which results in the difficulty for system developers to know what HL7 messages are available for their developing system.

In this session, we collect cases of system development based on HL7 and analyzed them. The analysis results contain system development methodologies, utility tools, HL7 messages developed, and actual program modules, and are organized as the report on the current state of healthcare information system development by using HL7. This report which should be reference on starting a system development based on HL7.

Following the analysis of the practical cases, we also describe an ongoing implementation using HL7 Version 3 of electronic patient record. We make the detailed description of the basic library module that enable HL7 Version 3 messaging very simply.

We expect that the report of the current state and our newly developed HL7 library are of great help to those who are developing health information system.

Keywords: HL7, Healthcare Information System Development

1. はじめに

標準化されたデータフォーマットあるいはプロトコルによるシステム開発は、他システムとの通信を容易にするだけでなく、それによって開発されたシステムの信頼性を向上させ、さらには、開発コスト(期間および費用)の軽減につながる。保健医療福祉分野においては、HL7(Health Level Seven)が国際的にも広くこの標準規格として周知されている。こうした現実的なメリットおよび国際的認知などを背景に、HL7を用いた医療情報システムあるいは電子カルテの開発が進みつつある。しかしながら、一方で、HL7を用いずに、独自の仕様を取り決め、システム間の接続を行おうとする事例も未だに少なくない。われわれはその一因を、既存のHL7メッセージが体系化されていないために利用しづらく、また、せっかく開発されたHL7メッセージがあるにも拘らず、それを知らずに新たなメッセージを開発したり、あるいは独自仕様を策定していることにあると考えている。

2. HL7を用いたシステム開発の現状

われわれは、これまでのHL7を用いたシステム開発例を把握するために、医療情報連合大会に投稿されたHL7に関連する論文を過去3年間に渡りサーベイした。その結果、第20回大会 16件、第21回大会 13件、第22回大会 12件であり、その適

用範囲も、オーダリングから、看護関連、画像関連、さらには地域連携とさまざまであった。第20回大会で、HL7関連の論文が増えているのは「データベースと標準化/パネルディスカッション: 医療情報基盤としての各コード標準化」が行われたからである。これらの数字を見ると、HL7を用いたシステム開発は着実に行われていることが分かる。本セッションでは、われわれのサーベイ結果と今大会へのHL7を用いたシステム開発に関する論文とを併せて、詳細な検討を加え、レポートとする。

3. HL7を用いたシステム開発のためのツール

現在、HL7を用いたシステム開発を行う場合、その多くはスクラッチからHL7メッセージ生成、パース、送受信のソフトを作り、全体のシステム開発の一部としていると思われる。しかしながら、すでに、商用ベースで多くのHL7支援ツールが流通しているので、ここでその一部を紹介したい。

3.1 ネオツール

米国のNeoTool Development, LLC (<http://www.neotool.com>)は古くからHL7の支援ツールを提供している。現在、NeoBrowse TCP for HL7(ネットワークベースのTCP/IPメッセージのエディタ、シミュレータ)、NeoEnable for HL7(HL7パーズング、エンコーディングのためのActiveXコンポーネント)、

NeoConnect TCP (TCP/IPメッセージ交換のためのActiveXコンポーネント)、NeoQueue (非同期メッセージキューのためのコンポーネント)、NeoSend for HL7 (メッセージ送信のためのActiveXコンポーネント)、NeoReceive for HL7 (メッセージ受信およびacknowledgment送信のためのActiveXコンポーネント) などがあり、これらを用いると、HL7のシステム開発がかなり容易になることが予想される。

3.2 オラクル

関係データベースシステムのOracleは、様々な業界のスタンダードを統合した、E-Business Suiteを有している。その中で、保健医療分野のスタンダードを実現するものとして、Healthcare Transaction Baseを提供している。このHealthcare Transaction Baseではシステム間通信をHL7で行い、データベースのデータ構造はHL7Version3RIMに準拠している。また、HL7Version2.xメッセージとHL7Version3メッセージとの変換もサポートしている。システム開発者はこれまでのデータベースアクセスと同様の感覚でHL7メッセージを送受信することができるため、移行がよいであると予想される。

3.3 マイクロソフト

2003年6月にMicrosoftがBizTalk Accelerator for Healthcareを発表した。このBizTalk Accelerator for HealthcareはHL7による保健医療情報の交換をサポートするものであり、さらにはHIPAA (Healthcare Insurance Portability and Accountability Act) にも準拠しており、患者プライバシー、情報の機密性、データセキュリティを高いレベルで実現している。

4. HL7を用いたシステム開発の将来展望

現在、HL7にはVersion 2.xとVersion 3の2つのバージョンが存在する。そのため、これからHL7を用いたシステム開発を始めようとする際には、どちらのバージョンを使用すべきか、判断に迷う場合がある。先のオラクルの節でも触れたが、HL7 Version 2.4メッセージをHL7 Version 3メッセージに自動的に変換することは容易である。また、HL7 Version 3では、まだBallot中のものが多く残っており、詳細が変更になる可能性もある。したがって、現在HL7を用いたシステム開発を行う場合には、既存のシステムでHL7 Version 2.xでメッセージ交換が可能なものについては、Version 2.xで行うのがもっとも確実である。

一方、HL7 Version 2.xはその適応範囲が主としてオーダリングに限られており、カルテ情報など複雑な情報を記述するためには、情報項目が不足する

場合がある。従って、電子カルテなどを開発する際には、必然的にHL7 Version 3を使用しなければならない。しかしながら、現在HL7 Version 3については、Version 2.xに比べて開発環境が整っているとは言えない。そこで、筆者らは、下記のようなHL7 Version 3開発環境を提供している。

4.1 HL7 バージョン3日本語版仕様書

HL7 Version 3の資料は英文で2,000ページを超える非常に大量の文書である。¹⁾これらをすべてのシステム開発者が熟読して理解し、システム開発に取り組むのは大変な労力である。そこで、われわれはこれらの資料の日本語化に取り組んでおり、Webページより参照できるようにしている。

4.2 HL7 バージョン3メッセージングライブラリ

今後保健医療情報システム開発への要求がますます複雑化し、システム開発者はそのドメインの専門家としてシステムへその知識を反映することが求められるであろう。そのような状況下において、それらのシステム開発者がHL7の仕様を理解することは必須ではないとわれわれは考えている。むしろ、HL7の仕様は隠蔽され、システム開発者はドメインの要求解決に専念できる環境が望ましいと考えている。システム開発をRDBMSを用いて行う際には、システム開発者は不要なデータベースの知識は必要とせず、自分の関わっているドメインの項目名さえ知っていればSQLを利用可能である。われわれは、それと同等レベルでHL7 Version 3メッセージを送受信可能なHL7 Version 3 Messaging Libraryを開発し、提供している。

5. おわりに

HL7を用いたシステム開発にも様々なメリット、デメリットがある。特に、国際標準を日本で使用するための努力やこれまでの環境を変更することは、見過ごすことのできないデメリットである。これは、おそらく独自のネットワークからTCP/IPネットワークに移行する際の苦勞に通じる点がある。しかしながら、HL7を用いたシステム開発を支援するための開発環境も徐々に整いつつある。今後のシステム開発に際しては、HL7を用いるメリットだけではなく、“HL7を用いないデメリット”についても勘案する必要がある。

参考文献

- [1] <http://www.hl7.org>

入院患者向け検体検査結果閲覧サービスの構築と評価

星本 弘之¹⁾ 渡辺 宏樹²⁾ 篠原 信夫²⁾ 美代 賢吾²⁾ 土榮 尚紀³⁾ 甘粕 敏昭³⁾ 武井 和浩³⁾ 本間 伸一³⁾
大江 和彦¹⁾

東京大学大学院 医学系研究科 社会医学専攻 医療情報経済学教室¹⁾
東京大学医学部附属病院 企画情報運営部²⁾ 鹿島建設株式会社 ITソリューション部³⁾

Design and Implementation of a Bedside Clinical Information System for Inpatients

Hiroyuki Hoshimoto¹⁾ Hiroki Watanabe²⁾ Nobuo Shinohara²⁾ Kengo Miyo²⁾ Takanori Doei³⁾
Toshiaki Amakasu³⁾ Kazuhiro Takei³⁾ Shinichi Homma³⁾ Kazuhiko Ohe¹⁾

Dept. Medical Informatics and Economics, Graduate School of Medicine, the University of Tokyo¹⁾
Department of Planning, Information and Management, The University of Tokyo Hospital²⁾
I.T. Solutions Department, Kajima Corporation³⁾

Abstract: Recently, more patients' participation in decision making in health-care is becoming very important, and their demands for sharing and/or disclosure of their own clinical information with health-care providers is getting very common. To satisfy these patients' demands, we developed a bedside clinical information system which provides the access to the clinical information for inpatients. With this system, patients can check their own clinical data and related information on the Web browser using their bedside information terminal. The system was developed as a Web-based application in Java Language, and uses various open, distributed technologies and standard message formats such as HL7. At present, the system is under test phase. In this paper, we describe the design and the implementation of the bedside clinical information system for inpatients in the University of Tokyo Hospital.

Keywords: Patient-centered clinical information system, Bedside Information Terminal, Computerized Patient Record, HL7, CORBA

1. はじめに

近年、医療は患者と医師による共同行為であるという認識が高まり、医療者は患者に対して必要な情報を提供し、患者は自己が受ける医療に関して選択・決定を行い、さらに治療の過程においても、患者と医師の共同作業が求められている。このような患者の自己決定、および自己管理をサポートするためには、患者自身が自己の状態や医療サービスについて、適切な内容の情報を入手できることが重要となる^{1,2)}。

患者に対する情報提供システムは、海外においては、インターネットを経由して患者が自己の検査結果や利用できる医療サービスの内容について自由に閲覧できるシステムなどの先行事例がみられ、それらのシステムの導入によって、患者と医師の間の情報量の非対称性が解消され、患者がより積極的に治療過程に参加するようになったとの報告がある³⁾。また、日本国内においても、成育医療センターなど一部の医療機関において、患者への情報提供システムの運用が行われている。

東大病院においては、新入院病棟の開設時より、入院患者が自分のベッドサイドに設置された情報端末を用いて、病院内の施設・設備などに関する各種情報や、検査や服薬などにおける一般的な情報や注意事項を閲覧することができるサービスが提

供されてきた⁴⁾。しかし、このシステム単独では、患者は自己の臨床情報を閲覧することはできず、主治医などの医療スタッフに依頼して、検査結果などの臨床情報を紙などに出力した形で入手していた。私たちは今回、東大病院の入院棟において、入院患者が自己の受けた各種検査の結果を、希望するときに自由に参照することができるシステムを構築した。このシステムは、先述した入院患者向けベッドサイド情報端末のサービスとして実現されており、患者は自分の入院ベッド脇に設置されている情報端末から、自分の希望する検査結果や関連する情報などにアクセスすることができる。本論文において、今回開発した患者向けの臨床情報提供システムの設計と実装について紹介する。

2. システム仕様

2.1 現行の患者向け情報システム

東大病院では、各入院患者のベッドサイドに、テレビと兼用の情報端末が設置されている。この情報端末は液晶テレビと、それに接続したiBox(セットトップボックス型コンピュータ)からなり、iBoxに組み込まれたWebブラウザを用いて、病院関連情報、検査に関する一般的な注意や情報、服薬の際の一般的な注意、健康管理上の情報などを閲覧することができる。また、有料ではあるが外部インターネット接続サービス経由で、インターネット上に存在する

各種情報源にアクセスすることが可能である。

2.2 システムの概要

今回開発したベッドサイド臨床情報システムは、現行のベッドサイド情報システムに対して、入院患者の検査結果や認証に必要な患者の入院状況を管理するサーバーを追加する形で構築した。表示用のクライアントは、新規開発とはせず、iBox内蔵のWebブラウザを用いて情報を閲覧するものとした。システムの開発に当たって定義した仕様は以下のものである。

- 1) 検体検査結果を HIS の CORBA-GW より取得
- 2) 内視鏡画像を内視鏡画像 Web-IF より取得
- 3) 患者の入退院、外泊、ベッド移動などの情報を反映した利用者認証
- 4) 医療従事者向けの任意患者の情報閲覧機能を提供
- 5) ログイン。患者が情報を閲覧した日時を記録し、利用状況統計を閲覧可能とする

以上の仕様に基づき、システムはベッドサイド端末向けの画面情報を生成する患者向け Web サーバー (PWS) と PWS が必要とする情報を病院情報システム (HIS) から取得して PWS に転送するゲートウェイによって構成されている (図1参照)。それぞれシステムの構成部品の設計を以下に述べる。

2.3 患者向け Web サーバー (PWS)

患者向け Web サーバー (PWS) は入院患者の臨床検査の結果と、病棟の入院患者に関する情報を管理し、患者からの閲覧要求に応じて、表示用の Web ページを生成して患者端末に表示する。そのため、PWS は Web ベースアプリケーションとして設計した。

2.3.1 検体検査

検体検査は、検査オーダーと検査種別をキーとして RDB で管理され、各検査項目がそれぞれ、標準値とともにレコードとして記録されている。表示に際しては、各検査項目ごとに、検査値とその標準値の範囲、および標準値範囲における検査結果値の概略位置についてのグラフが表示される。また、各検査項目についての簡単な説明文書へのリンクも表示される (図2参照)。

2.3.2 内視鏡画像

内視鏡画像は、画像本体とその管理情報をそれぞれ処理する必要がある。PWS では、画像の管理は 1 検査オーダーあたり 1 ディレクトリを作成し、その中に内視鏡画像ファイルとサムネイルファイルを格納している。それぞれの検査画像がどの患者の検査かという情報は RDB で管理され、各画像ごとに独立したレコードとして記録されている。表示の際には、サムネイル画像の一覧が表示され、閲覧したい画像を選択すると、それに対応する検査結果画像が表示される。(図3参照)

2.3.3 患者認証

PWS は患者が自己の臨床情報を閲覧するに当たり、他の患者の情報を過って閲覧することがないよ

う、アクセス元の端末の識別番号と入院患者の在床情報をつきあわせた認証管理情報と患者個人ごとに設定されたパスワードにより患者の認証処理を行う。

2.4 臨床情報変換ゲートウェイ (GW)

臨床情報変換ゲートウェイ (GW) は、HIS から入院患者の検査結果の情報を取得し、PWS に転送する。HIS から取得する情報は、検体検査結果と内視鏡検査結果の画像情報であり、これを HL7 バージョン 3 に準拠した XML 文書の形式に変換した上で、PWS に転送する。検体検査結果は、HIS が持つ CORBA-GW³⁾ を経由して定期的に取得する。内視鏡検査結果は、内視鏡画像システムの Web インターフェイスを利用し、http で画像と患者情報を取得する。また GW は、患者認証に必要な患者の在床情報を常に最新の状態に維持するため、HIS から定期的に入退院患者の情報を取得し PWS に転送する。入院患者については、各患者の現在入院しているベッド番号、診療科などのほか、外泊に関する情報も取得し、PWS に配信する。退院患者については、その日に退院した患者の一覧を PWS に定期的に配信する。入院患者の情報を取得する際にも、HIS の CORBA-GW を使用するが PWS に転送する際のフォーマットは CSV である。

2.5 セキュリティ

本システムが取り扱う情報は患者自身のプライバシーに深く関わる情報であるため、誤って他の患者の情報を閲覧してしまうことがないよう、患者が検査結果を閲覧することができる端末を限定した。具体的には、アクセス可能な端末をその患者が入院しているベッドサイドに設置されている端末に限定し、その端末からのみ、パスワード認証を経て各患者の検査結果の閲覧画面にアクセスすることができるようにした。また、患者が外泊している期間中は、帰院オーダーが実施され、その情報が PWS に配信されるまで、その患者の情報は患者、医療従事者ともに閲覧不可とした。さらに、患者が退院した際には、GW から定期的に PWS に配信される退院患者リストに基づき、PWS に保存されている退院患者に関する臨床情報はすべて削除するものとした。

3. 実装

3.1 技術要素

本システムの実装は Java 言語で行った。すべてのシステムは、Java2-SDK (バージョン 1.3.1 以上、Sun Microsystems, Inc.) で記述した。ベース OS は、PWS には RedHat Linux (バージョン 7.2、RedHat, Inc.)、GW には RedHat Linux Advanced Server (バージョン 2.1、同) を使用した。PWS は、Web アプリケーション動作環境として Jakarta-Tomcat (バージョン 4.1 以上、The Apache Software Foundation) と、データベースエンジンとして PostgreSQL (バージョン 7.2) を用いた。GW は HIS との通信に必要なミドルウェアとして、Intersgate Application Server

(Web-Jエディションバージョン5.1, 富士通株式会社)を用いた。

3.2 HL7v3 メッセージフォーマット

GWとPWSとの間の情報交換フォーマットとして、現在規格制定作業中のHL7バージョン3に準拠したメッセージ形式を採用した。メッセージスキーマは、昨年8月に公開された投票パッケージ第3版に収録された検査結果通知メッセージ(ObservationEvent Completed)を用いた⁷⁾。

4. 結果

4.1 テスト環境

テストを行った環境は以下の通りである。

- GW : Pentium4-2.4GHz, 1GB-RAM, 50GB-HDD.
- PWS : Dual Pentium3-1.13GHz, 1GB-RAM, 50GB-HDD.

テストは、東大病院入院棟の全入院患者の在床・外泊・退院情報と、あるフロアに入院している全患者(n=43)についての検査結果を30分ごとにHISより取得してPWSに配信するという設定とし、9月2日14:45より24時間連続して行った。

4.2 性能

全入院患者の一覧データをHISより取得するのに必要な時間は平均して1分30秒、PWSへの転送は平均11秒で完了した。

テスト期間中にHISより取得した検査結果は310件。そのうち処理対象メッセージは161件、対象外(尿検査など)は149件であった。処理時間は、HISからデータを取得するのに一件平均約0.67秒、GWからPWSにFTP転送するには、認証に約10秒を要するが、メッセージ転送は一件あたり約1.6秒程度であった。

PWSでのデータ処理時間は、検体検査結果のRDBへの登録が1メッセージあたり平均0.12秒、患者認証テーブルの更新には平均11分30秒(患者数:約940~950)を要した。

検査結果の閲覧は、表示項目を選択してからおおむね1秒以内に画面表示が行われた。

5. 討論

今回我々が構築したシステムは、入院患者に対してリアルタイムに検査結果の閲覧サービスを提供するものである。本システムが提供する臨床情報は血液検査と内視鏡検査のみに限られているが、その結果を患者が任意の時間に閲覧することが可能であり、また、患者自身が希望すれば、病院外の情報源へのアクセスも可能であることから、患者の自己の疾患に対する理解が深まり、患者がより主体的に治療に関与することも期待できる。また患者と医療提供者との間の情報共有が促進されるため、先行研究でも示されているように、患者と医師との関係にも何らかの影響を与える可能性がある。現段階では試験運用段階であり、システムの使用

感についてのデータはまだ得られていないが、データの更新や画面の生成・表示に必要な時間が短いことと、過度のアクセスが集中する事態は考えにくいことから、実運用に当たっても大きな問題は生じないと思われる。

本システムは入院患者のみを対象としたサービスだが、外来治療を受ける慢性疾患の患者などに対するデータ参照サービスは、患者が今までの検査履歴を参照して自己管理に利用可能となるため、非常に有用と考えられるため、是非実現したい拡張である。さらに、本システムで採用しているHL7形式のメッセージは、国際的な標準規格であるため、外来患者や退院する患者に対して、検査結果をHL7形式のデータファイルとして提供するサービスも考えられる。これにより、HL7形式のデータインターフェイスを持つシステムを使用している医療機関に転院した際には、過去の検査結果をそのまま新たな医療機関に渡すことができるため、医療の継続性という点からも有用なサービスとなりうると思われる。

6. まとめ

我々は今回、入院患者に対して検体検査結果と内視鏡検査画像を閲覧するサービスを提供するシステムを開発した。このシステムには、HL7V3などの新規技術を採用しており、また今後の拡張の可能性も十分なものがあると考えられる。本システムの有効性は現在検証中であるが、患者中心の医療を実現していく上で、この種のシステムは非常に重要な役割を果たすものである。

参考文献

- [1] Kaihara S. Administrators, doctors or patients. Who are the users of information system. In: Cesnik B et al. eds., Proceedings of MEDINFO98 Supplement, Amsterdam: IOS press, 1998; Part 1: pp. 6-7.
- [2] Ball MJ, Lillis J. E-Health: transforming the physician/patient relationship. International Journal of Medical Informatics, 2001; 61: pp. 1-10.
- [3] Cimino JJ, Patel VL, Kushniruk AW. The patient clinical information system (PatCIS): technical solutions for and experience with giving patients access to their electronic medical records. International Journal of Medical Informatics, 2002; 68 (1-3): pp. 113-127.
- [4] Takamatsu S, Miyo K, Hujisaku S, Amakasu K, Takei K, Igarashi T, Koide D, Onogi Y, Ohe K. Development and Implementation of Bed-Side Network Computer for Providing Various Information to Inpatients. AMIA 2002 Annual Symposium Proceedings, 2002; pp. 1176.
- [5] Ohe K. A Hospital Information System based on Common Object Request Broker Architecture (CORBA) for Exchanging Distributed Medical Objects - an approach to future environment of sharing healthcare information. In: Cesnik B. et al. eds., Proceedings of MEDINFO98, Amsterdam: IOS press, 1998; pp. 962-964.

[6] Ohe K, Miyo K, Onogi Y, Ueda K, Takada M, Chihara T. Implications of a General data model for implementing OODB/CORBA-based computerized patient record system. In: Patel V. et al. eds., Proceedings of MEDINFO2001, Amsterdam: IOS press, 2001; pp. 789.

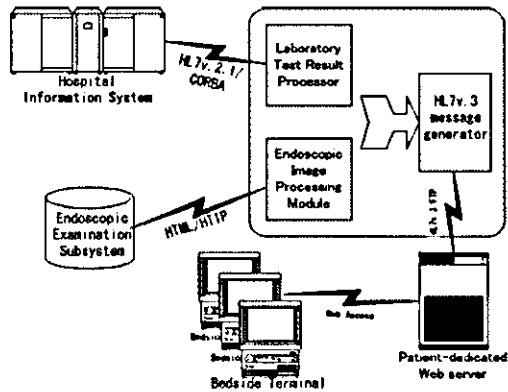


図1 検査結果提供システム構成図

[7] Health Level Seven, Inc. URL: <http://www.hl7.org/>. Downloaded on August 15, 2002

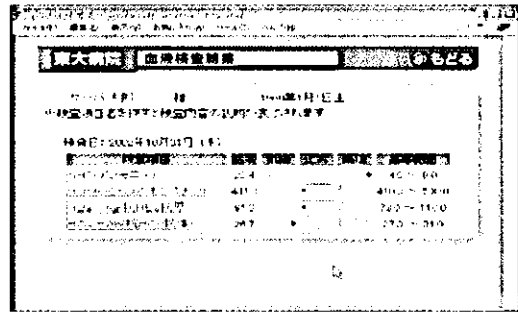


図2 検体検査結果閲覧画面

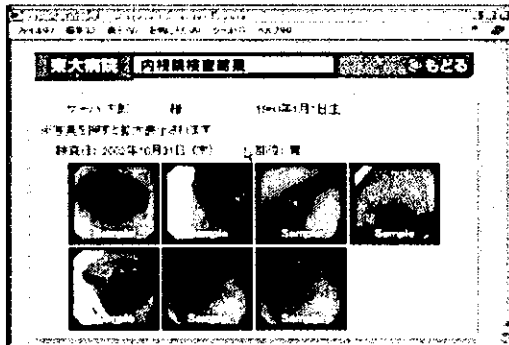


図3 内視鏡検査閲覧画面

HL7メッセージによる大規模病院情報システムの開発効率の向上

森山 和好¹⁾ 中村 伏孝²⁾ 青木 栄治³⁾ 河野 順⁴⁾ 美代 賢吾⁵⁾ 坂本 憲広⁶⁾
株式会社麻生情報システム 医療システム事業部 先端医療情報システム研究所¹⁾
神戸大学医学部附属病院 医療情報部²⁾

A Method for Improving System Development Efficiency of A Large-scale Hospital Information System By Using HL7 Messages

KAZUYOSHI MORIYAMA¹⁾ FUMITAKA NAKAMURA¹⁾ EIJI AOKI¹⁾ JUN KOUNO¹⁾
KENGO MIYO²⁾ NORIHIRO SAKAMOTO²⁾

Laboratory for Advanced Medical Information Systems, Aso Information System, Inc.¹⁾
Division of Medical Informatics, Kobe University Hospital²⁾

Abstract: Aso Information System Co. Ltd. has made the overall updates of the Large-scale Hospital Information System in Kobe University Hospital in March, 2003. We have implemented the connection between four external device vendors and order entry system with the use of HL7 messaging. We only had 5 months for this whole system development and 1.5 month for connecting the external devices. Given this extremely limited amount of time, effective system development and implementation was the essential factor. In order to implement the high-quality system under this severe condition, we have applied HL7 for connecting the external devices. By utilizing HL7 in our project, we were able to commence the operation of system as originally planned and also, we have not encountered any system troubles so far.

Keywords: HL7 Version2.4, Interface

1. はじめに

麻生情報システムは、平成15年3月に神戸大学病院の総合医療情報システムの全面リプレースに際し、外部機器ベンダー4社とオーダーエントリーシステムの接続をHL7メッセージ交換方式により行った。今回の全システム開発期間は5ヶ月という非常に短い期間であり、外部機器接続においては、実質作業期間が1.5ヶ月しか確保できない状況であり、より効率の良い方法を採用して開発を行うことが必須条件であった。このような厳しい条件をクリアし、品質の高いシステムを導入することを目的として、HL7メッセージ交換方式を利用した外部機器接続を行った。

HL7メッセージ交換方式を採用したことによる開発効率の向上について紹介する。

2. HL7メッセージ交換方式を採用した理由

2.1 各ベンダーへの送信イベントが共通

A社: 患者属性更新、入院通知、入院患者の移動、退院通知、検査結果

B社: 患者属性更新、入院通知、入院患者の移動、退院通知、看護情報

C社: 一般臨床オーダー

D社: 患者属性更新

ベンダー単位での設計・製造ではなく、イベント単位(部品化)で設計・製造を行うことによる作業効率の向上が可能であった。

2.2 標準規格

3月リリース後もトリガーイベントの追加や接続機器の増加が予定されていたことから、機能変更・拡張にも容易に対応できるよう、確立された標準規格である必要があった。

HL7メッセージは海外22カ国で使用されており、企業会員はヘルスケアシステムベンダーの90%を占める400社以上が加盟している。

2.3 信頼性

より短い期間で安定したシステムを構築するために、信頼性の高いメッセージを採用する必要性があった。

2.4 通信プロトコルに依存しない

HL7メッセージはOSI手順の第7層(アプリケーション層)に位置するため、通信プロトコルの制約を受けないシステム構築が可能であった。

平成15年3月のリプレース前より、ソケット通信によってHIS接続が行われていたため、HL7メッセージを送信する方法として、各ベンダーにプログラム作成のノウハウがある既存のソケット通信を用いて開発を行った。

3. 開発効率

設計フェーズにおいては、イベント毎にメッセージ形式が確立しているHL7を用いたことによって、メッセージレイアウト作成に係る工数を、大幅に削減することができた。しかし、製造フェーズにおいては、弊社で初めてHL7を実装したことにより、イベント毎のメッセージ編集プログラムを作成し、それらをコンポーネント化する作業が必要であった為、今回は、このフェーズでの工数削減につなげることはできな

かったが、今後、同様の開発を行う際には、今回作成したコンポーネントを流用することが可能なため、製造フェーズにおいても大幅な工数削減が期待できる。

4. 今回の開発を通して

設計フェーズの工数が削減できたことによって、接続テスト等、システムの品質向上につながるフェーズに工数削減分を回すことが可能となり、実質開発1.5ヶ月という非常に短い期間であったが、リリース後に大きなトラブルを起こすことなく正常

に稼動している。

5. 今後の取り組み

今回の開発ではHL7 Ver2.4メッセージを用い、通信方法として既に稼動状態であったソケットを選択したが、今後はシステムに更なる汎用性を持たせるために、SOAP、HTTP等の通信方法に対応していく予定である。

また、電子カルテの開発に際して、全面的にHL7 Ver3へバージョンアップする予定である。

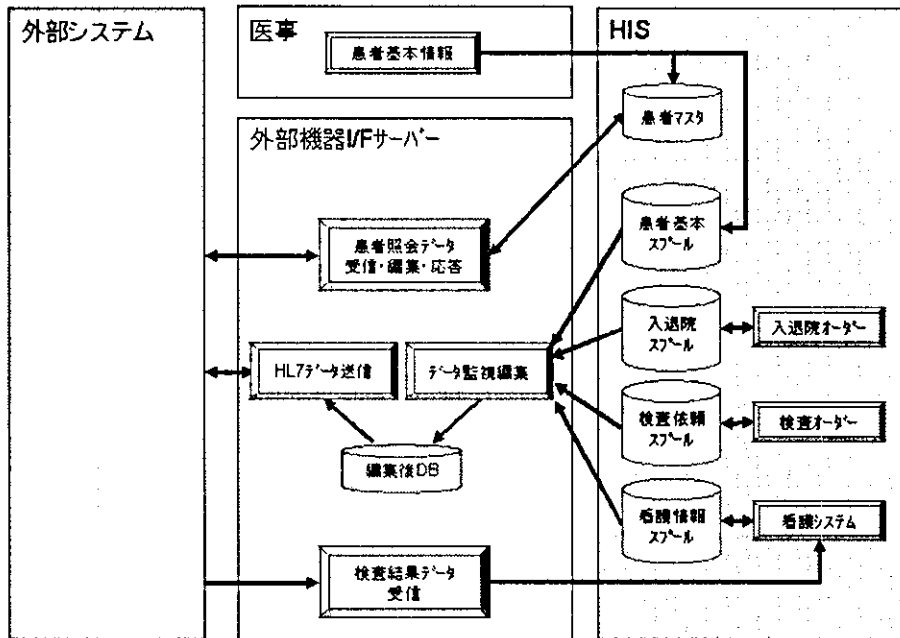


図1 システム構成図

HL7バージョン3メッセージングライブラリの開発

増田剛¹⁾ 坂本憲広²⁾

財団法人先端医療振興財団臨床研究情報センター遺伝子データベース研究部¹⁾
神戸大学医学部附属病院医療情報部²⁾

A Development of an HL7 Version 3 Messaging Library

Gou Masuda¹⁾ Norihiro Sakamoto²⁾

Translational Research Informatics Center, Foundation for Biomedical Research and Innovation¹⁾
Department of Medical Informatics, Kobe University Hospital²⁾

Abstract: Since the Japanese Ministry of Health, Welfare and Labor recommends the use of HL7 Version 3 (HL7 V3) as one of the standard protocols for exchanging clinical information, it is expected that HL7 V3 is widely used in the development of Electronic Patient Record (EPR) system in the near future. However, there are few development tools that can be easily used for implementing HL7 V3 based EPR systems in present. Therefore, every developer of HL7 V3 based EPR systems is required to deeply understand the specifications of HL7 V3 such as Reference Information Model (RIM), V3 Data Types, and message interactions in order to achieve conformance to the standards. Moreover, because the difference of the interpretation of the standard among the developers can be caused, interoperability might be ruined even if they are based on the same standards. To cope with these problems, we developed an HL7 V3 messaging library for the development of EPR systems. In this library, individual RIM class and data type is implemented as a C# class by using the Microsoft .NET framework. Implementation details of HL7 V3 XML messaging are encapsulated into the individual class. It improves maintenance of library. Even if the standards will be modified in the future, the change part in the library can be suppressed to the minimum. Moreover, we provide more generic interfaces such as Patient and Doctor as an upper layer of these classes. These interfaces enable developers to handle HL7 V3 messages without knowing details of RIM and the message interaction. The improvement of the development efficiency can be expected. We develop a prototype of prescription ordering system in order to evaluate our library. A result of performance evaluation shows that the library constructs RIM objects from clinical data from users and generates an XML message in reasonable time.

Keywords: HL7 Version 3, Microsoft .NET Framework, Prescription Order System, XML Digital Signature

1. はじめに

保健医療情報の交換規約であるHL7バージョン3¹⁾は、厚生労働省「保健医療分野の情報化にむけてのグランドデザイン」で電子的報交換のための標準規約として利用が推奨されており、今後の電子カルテシステム開発で広く利用されることが予想される。しかし、現在利用可能な開発ツールやライブラリがほとんどないため、電子カルテシステムの開発者は規約に準拠するシステムを実装するために、規約で定義された情報モデルや、メッセージインタラクションの詳細を理解する必要があり、開発には多大な時間と労力を要する。また、アプリケーション毎の実装により、開発者間での規約の解釈の違いが生じうるため、標準規約を使用しているにもかかわらず他システムとの互換性が損なわれかねない。そこで本研究ではこれらの問題を解決するために、電子カルテシステム開発のためのHL7バージョン3メッセージングライブラリを開発し評価を行なった。

2. 方法

現在使用されている電子カルテシステムの多くは、

Visual Basic言語を初めとするMicrosoft社の開発環境を使用し、Windows OS上に実装されている。このような現状を踏まえ、これらの既存電子カルテシステムから、HL7バージョン3ベースのシステムへ容易に移行できるように、本ライブラリはMicrosoft .NETフレームワーク上でオブジェクト指向言語であるC#を用いて実装する。

本ライブラリの構成概念図を図1に示す。HL7バージョン3では、メッセージの構成要素を規定するための情報モデルとして次の3つのモデルが提供される。

- 1) 参照情報モデル (RIM: Reference Information Model): 人や物を表すEntityクラス、役割を表すRoleクラス、行為を表すActクラス、行為への参加を表すParticipationクラスの4つのクラスを中心として、HL7が対象とする保健医療分野の診療情報の構造や関係を定義するための計68個のRIMクラスが定義されている。RIMは、バージョン3で定義される全てのメッセージの構成要素として共通に参照される。
- 2) データタイプ: RIMクラスの構成要素を定義する。例えば、コード化値を表す

ConceptDescriptor型や物理量を表すPhysicalQuantity型、住所を表すPostalAddress型が定義される。

- 3) ボキャブラリドメイン: RIMクラスの各コード化属性に対応付けられ、その属性において使用可能なコード値を規定する。例えば、ボキャブラリドメインAdministrativeGenderは、人を含めた生物を表すRIMクラスである、LivingSubjectクラスのadministrativeGenderCode属性に使用可能なコード化値を規定する。

本ライブラリでは、これらの情報モデルを、HL7バージョン3の規約に忠実に、個々のC#クラスとして実装する(図1第1層)。HL7バージョン3で定義されるメッセージはメッセージタイプと呼ばれ、第1層で定義された情報モデルを用いて構成される。本ライブラリでは、一つのメッセージタイプは、一つのC#クラスとして実装する(図1第2層)。さらに、これらのC#クラスをアプリケーションから使用する場合に、HL7の知識を可能な限り隠蔽するために、患者や医師、処方箋といった粒度のデータに対する汎用的なアクセス方法を規定したインターフェースを提供する(図1第3層)。これらは、C#のインターフェース型として実装する。例えば電子カルテシステムや遺伝子情報管理システムといった、これらのライブラリを利用する上位層アプリケーションは、第3層に定義されたインターフェースを通してデータをバージョン3メッセージに設定し、また、メッセージに含まれるデータを取得する(図1第4層)。

3. 結果

3.1 RIM

現在利用可能な仕様の最新バージョンであるRIM 2.01に忠実に、全てのRIMクラスについて、各RIMクラスに1対1に対応するC#クラスを実装した。RIM属性は、その属性のデータタイプの型を持つ、C#のプロパティメンバとして実装し、各RIM属性に対してその属性値を取得するgetアクセサ、および属性値を設定するsetアクセサを実装した。RIM関連は、多重度が0または1である関連については、その関連名で宣言される、関連先クラスの型を持つプロパティメンバとして実装した。また、多重度が2以上のRIM関連については、関連先クラスを要素とするコレクション型のプロパティメンバとして実装した。クラスの継承関係についても、RIMの仕様に従い、InfrastructureRootクラスをスーパークラスとして、それぞれのRIMクラスを実装した。

各RIMクラスに対応するC#クラスは、そのRIMクラスのXML表現に関する知識を隠蔽する。RIMクラスは、ToXmlElement()メソッドを実装し、その中で、RIM属性のXMLへの変換は、その属性を構成するデータタイプクラスに処理を委譲し、また、RIM関連については、関連先のRIMクラスに対応するC#クラスに処理を委譲しながら、そのRIMクラスのXML表現を表すXML要素を作成する。一方、XMLメッセージからRIMオブジェクトへの復元

は、各RIMクラスがクラスメソッドLoadXML()を実装し、XML化とは逆の手続きでXML要素からRIMオブジェクトを復元する。

3.2 データタイプ

HL7バージョン3投票パッケージ4に含まれる、"Data Types - Abstract(1st Informative Ballot)"及び、"Data Types Implementation Technology Specification for XML (5th Committee Level Ballot)"に準拠し、RIMクラスと同様に、各データタイプも個々のC#クラスとして実装した。これらのクラスは、上記の仕様で規定される各データタイプのXML実装に関する知識を隠蔽している。データタイプ仕様は、実装技術に依存しない概念的な仕様であるため、データタイプのセマンティックな性質を規定する項目が数多く含まれている。しかし、今回は、これらの項目についての実装は必要最小限に留め、主にXML表現に関する部分を中心に実装を行った。

3.3 ボキャブラリドメイン

HL7が提供するボキャブラリドメインについて、個々のボキャブラリドメインを一つのC#クラスとして実装した。これらのクラスには、そのボキャブラリドメインに含まれるコード化値を取得するためのプロパティメンバや、与えられたコード化値が、そのボキャブラリドメインに含まれる妥当な値かどうかを検査するメソッドを実装した。

3.4 メッセージタイプ

メッセージタイプについても、これまで述べた各モデルと同様に、ひとつのメッセージタイプに対して、一つのC#クラスを実装した。このクラスは、一つのメッセージタイプを構成するRIMオブジェクトのツリーを持つ、RIM Messageクラスのラッパークラスとして表現される。ライブラリを利用するアプリケーションに公開されるインターフェースは、メッセージを構成するRIMオブジェクトではなくメッセージタイプである。各メッセージタイプは、後述するインターフェース型を実装することにより、アプリケーションに対して、HL7の知識を隠蔽した、データの汎用的なアクセス方法を提供する。

3.5 汎用インターフェース

C#のインターフェース型は、そのインターフェース型を実装するクラスに対して課せられる契約を示す。例えば、患者というインターフェース型は、患者氏名の取得および設定をサポートするという契約を含み、このインターフェース型を実装するクラスは、その契約に基づき、患者氏名の取得および設定についての実装を提供しなければならない。本ライブラリでは、アプリケーションの開発者に対して可能な限りHL7の知識を隠蔽するために、患者や処方箋といった粒度のデータに対する操作を契約として定義した汎用インターフェースを、このインターフェース型として実装した。ライブラリを使用するアプリケーションは、特定のユースケースにおい

て必要となるインターフェース型を実装したC#クラスを定義することで、HL7 V3メッセージに含まれるデータを設定または取得することが可能となる。このとき、クラスではなくインターフェース型として定義されるため、アプリケーション側では、そのインターフェース型で規定される契約さえ満足するならば、実装がどのようなものであってもよい。一方、メッセージタイプクラスもまた、そのメッセージタイプに含まれる要素に対する患者や医者といったデータに対して、汎用的なアクセス方法を提供するために、必要なインターフェース型を実装する。

これらのインターフェースでは、RIMやデータタイプについての知識は隠蔽され、C#で提供されているStringやInt32、DateTimeといった基本型を用いてメッセージを構成する診療情報を設定、取得することが可能である。また、我々が過去の研究で示したように、「電子保存された診療録情報の交換のためのデータ項目セット」(J-MIX)³⁾の項目は、HL7 RIMで表現可能であるため⁴⁾、インターフェースで使用される引数名やメソッド名は、可能な限り、(J-MIX)の英語標準ラベルに基づき設計した。これにより、アプリケーションの開発者にとってより一般的であるJ-MIXの用語を使って、HL7のメッセージを扱うことが可能となる。

3.6 アプリケーション

開発したライブラリに基づき処方オーダーアプリケーションを試作し、処方オーダーアプリケーションのための汎用インターフェースとして、表1に示すインターフェースを実装した。IPrescriptionインターフェースの定義を表2に示す。また、図2は、これらのインターフェースを使って、アプリケーションで処方箋のRIMオブジェクトツリーを組み立て、XMLメッセージを生成する例である。例中で使用されているPatientModelやAuthorModelクラスといったモデルクラスは、インターフェース型の規約に従ってアプリケーション側で実装されたクラスであり、ライブラリはこれらのクラスの実装には関知しない。

4. 考察

本ライブラリによるアプリケーションの開発効率の評価に関しては、現在のところ定性的な評価しか行なうことができないが、本ライブラリで提供するRIMやデータタイプの上位層のインターフェースを利用することにより、開発者はRIMやメッセージインタラクションの詳細を知ることなく規約に準拠したメッセージを生成することが可能であり、開発効率の向上が期待できる。

また、HL7バージョン3の仕様は、現在も開発中であり今後も仕様の改変が予想される。しかし、RIMクラスやデータタイプを個別のクラスとして忠実に実装し、各クラスのXMLメッセージングの実装を

隠蔽することで、今後規約が変更された場合においても、ライブラリの変更箇所を最小限に抑えることができる。

しかしながら、RIMクラスやデータタイプを個々のC#クラスとして実現することにより、一つのメッセージについて生成されるオブジェクト数が増加するため実行効率への影響が懸念される。そこで、今回試作した処方オーダーアプリケーションを使用し、メッセージ生成処理の実行効率を評価した。例として1項目につき2薬品の処方指示を含み、計10項目から構成される処方オーダーを入力し、その入力からRIMオブジェクトを作成しXMLメッセージに変換するまでの実行時間を測定した。Mobile Pentium3-M 866MHzメモリ512MB搭載のPCで測定した結果、平均実行時間は約80msであった。これより個別のクラスとして実装した場合の実行効率への影響は実用上問題ないと考える。

さらに、実運用を考えた場合、送信されるHL7バージョン3メッセージには電子署名を付加することが要求される。本ライブラリでは、我々が並行して開発しているXML電子署名ライブラリを用いて、メッセージに対して電子署名を容易に付加することも可能である⁴⁾。

5. 結論

本稿では、電子カルテアプリケーション開発のためのHL7バージョン3メッセージングライブラリについて述べた。本ライブラリを用いることにより、HL7バージョン3に関する詳細な知識を持たないアプリケーションドメインの開発者でも、インターフェース型として定義される契約に従うことにより、規約に準拠したアプリケーションを容易に開発することが可能となる。今後は、本ライブラリに基づき、アプリケーション間でHL7バージョン3メッセージ交換を行なうためのメッセージングフレームワークの構築へと発展させるとともに、実システムへの開発に適用し、規模適応性や開発効率についての評価を行なう予定である。

参考文献

- [1] HL7 Version 3 Standard, Health Level Seven Inc., Available at <http://www.hl7.org/>, 2002.
- [2] 電子保存された診療録情報の交換のためのデータ項目セットの作成,(財)医療情報システム開発センター, 2002.
- [3] N. Sakamoto et. al. "A New Approach for Unification of Healthcare Information Exchange Protocols Through HL7 RIM," Japanese Journal of Medical Informatics, Vol 21. No. 1, pp. 13-22, 2001.
- [4] 石戸是亘他、HL7v3メッセージに対応したXML電子署名ライブラリの開発、第23回医療情報学連合大会、2003.

3-A-3-2 標準化/一般口演: 標準化2

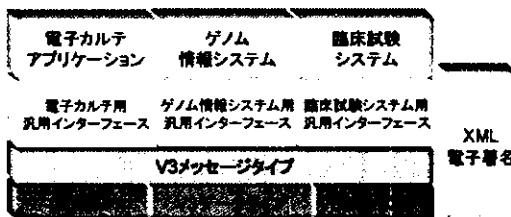


図1 ライブラリ構成の概念図

```
// 処方オーダーインタラクションに関する情報を提供するプロバイダオブジェクトを生成
InteractionProvider interactionProvider = new PORXING02122InteractionProvider();
MessageModel orderMessage = new OrderMessage();
orderMessage.SetSenderDeviceId(1001);
orderMessage.SetSenderDeviceName("client001.tri-kobe.org");
...
// IAuthorとIPatient インターフェースを実装するモデルオブジェクトを生成
AuthorModel author = new AuthorModel();
author.SetValues("TR1001", "神戸", "太郎", "こうべ", "たろう",
    AdministrativeGender.Singleton.Male, new DateTime(1960, 1, 1));
...
PatientModel patient = new PatientModel();
patient.SetValues("F00100", "臨床", "花子", "りんしょう", "はなこ",
    AdministrativeGender.Singleton.Female, new DateTime(1970, 3, 10));
...
// IPrescription インターフェースを実装する処方箋モデルオブジェクトを生成し、患者と
// 作成者を生成
PrescriptionModel prescription = new Prescription();
((IPrescription)prescription).Author = (IAuthor)author;
((IPrescription)prescription).Patient = (IPatient)patient;
...
// 処方項目を作成し、処方箋モデルオブジェクトに追加
PrescriptionItem item = new PrescriptionItem();
...
prescription.Add((IPrescriptionItem)item);
...
// メッセージタイプ (RIM オブジェクトツリー) を生成
PORXMT002122MessageType messageType =
    InteractionProvider.CreateMessageType((IMessage)orderMessage);
messageType.ConstructPayload((IPrescription)prescription);
...
// RIM オブジェクトツリーをXMLに変換
XmlDocument xmlDocument = messageType.ToXmlDocument();
```

図2 ライブラリの使用例

表1 処方オーダーのインターフェース型

インターフェース型	説明
IPrescription	一つの処方箋情報に対応するインターフェース。処方箋の作成者、患者情報、作成日時や処方項目へのアクセスを提供。
IPrescriptionItem	処方箋の項目に対応するインターフェース。処方される薬品の情報と検査法へのアクセスを提供。
IMedicinalPackage	一つの医薬品に対応するインターフェース。薬品名や医薬品コードに対するアクセスを提供。
IMedication	処方箋の用法に対応するインターフェース。用法名や検査量、服用期間へのアクセスを提供。
IAuthor	処方箋の作成者に対応するインターフェース。作成者のIDや氏名へのアクセスを提供。
IPatient	処方箋の患者に対応するインターフェース。患者IDや氏名に対するアクセスを提供。

表2 IPrescription インターフェース型の定義

```
string Oid [get] : 処方箋のOIDを取得するプロパティ
string Id [get] : 処方箋のIDを取得するプロパティ
DateTime IssuedDateTime [set, get] :
    処方箋発行日時を取得または設定するプロパティ
ArrayList PrescriptionItems [get] :
    処方箋項目一覧を取得するプロパティ
IAuthor Author [set, get] :
    処方箋の作成者を取得または設定するプロパティ
IPatient Patient [set, get] :
    処方箋の患者を取得または設定するプロパティ
int Add(IPrescription item) :
    処方箋項目を追加するメソッド
void SetId(string oid, string id) :
    ID情報を設定するメソッド
```

HL7v3メッセージに対応したXML電子署名ライブラリの開発

石戸 是亘¹⁾ 西脇 清行²⁾ 増田 剛¹⁾ 坂本 憲広³⁾

財団法人先端医療振興財団 臨床研究情報センター¹⁾ 株式会社 ハーバー・ソフトウェア²⁾
神戸大学医学部附属病院 医療情報部³⁾

A Development of a XML Digital Signature Application Library for HL7v3 messages

Yoshinobu Ishido¹⁾ Kiyoyuki Nishiwaki²⁾ Gou Masuda¹⁾ Norihiro Sakamoto³⁾

Translational Research Informatics Center, Foundation for Biomedical Research¹⁾

Harbor Software, Co. Ltd.²⁾ Department of Medical Informatics, Kobe University Hospital³⁾

Abstract: The HL7 Technical Committees promote the standards of the medical informatics systems such as electronic medical records systems. XML in HL7 is used as data format and these xml data is sent and received on the internet. However, security is talking stage in HL7, so it is not described in detail at present yet. There are only descriptions that they shall all consider security issues(confidentiality of patient information, authentication and integrity etc.) while developing the HL7 Data Model and Version 3 messages. In this thesis, I would like to focus on signature and announce a development of a XML digital signature application library. We developed it for electronic medical records systems using HL7v3 messages. We implemented it based on "XML-Signature Syntax and Processing" which is W3C Recommendation. Therefore, we use C# to develop the library so that it could be used in many existing systems. As a result, you can easily add the function of XML digital signature to your system that is developed using such a programming language as VB. Moreover, I want to report the problems when we use the XML digital signature of W3C in this announcement. With regard to the signature given to the medical information, it is often required to remain valid over long periods. But it is not described in "XML-Signature Syntax and Processing" how to extend a valid time limit of signature. So, there are many cases that are not appropriate to use this library. I will report our future plans to solve these problems as our measures to be taken.

Keywords: security, HL7

1. はじめに

電子カルテなどの医療情報システムは、HL7として標準化が進められている。そのメッセージは、XML化されネットワーク上で送受信されることを想定されている。しかし現時点において、HL7でのセキュリティは検討段階であり、明文化された規定はない。HL7v3の Message Development Frameworkにて、患者情報の機密保護、認証、扱うデータの完全性については、検討するとあるのみである。本論文では、データの完全性について焦点を当て、XML電子署名モジュールの実装について発表する。HL7による電子カルテアプリケーションの実装に併せ、汎用的なXML電子署名モジュールを開発した。XML電子署名については、W3C Recommendationとなっている、XML-Signature Syntax and Processingを実装した。そのため、開発言語は、多くの既存の電子カルテアプリケーションにおいても活用可能な.NET Visual C#とした。これにより、VB等をベースとして開発された既存の電子カルテアプリケーションにおいても、簡単にXML電子署名機能の拡張が可能となる。また、本発表では、W3CのXML電子署名を電子カルテに適用した場合の問題点及びその対策についても報告したい。医療情報に施された署名は、長期に保存されることが要求されるデータがあり、この点において

W3Cの規格のみでは十分ではない利用方法も多々発生することが予想される。実装モジュールの今後の検討課題及びその対策として、その将来設計を報告する。

2. 開発環境及び依存ライブラリ

本ライブラリの実装環境は以下のようなものとした。

2.1 開発環境

- 1) オペレーティングシステム
Windows 2000 Professional
- 2) 開発ツール
Microsoft Visual Studio . NET 2003
- 3) 主な開発言語
C#

2.2 依存ライブラリ

1) CAPICOM
CryptoAPIのCOMインターフェースを提供するActiveXコントロール 主に鍵情報へのアクセスのために使用。

2) WSE (Web Services Enhancement)
公開鍵証明書から公開鍵を取得するために使用する。WSEの利用は、.NET標準のX509Certificateクラスには、公開鍵取得のためのメソッドがないため、WSEのMicrosoft. Web. Services. Security. X509. X509Certificateクラス

のPublicKeyというメソッドで取得。

3) ICカード

公開鍵証明書及びその秘密鍵のホルダーとしてのICカードには、standard-9(三菱仕様)、Cryptoflex(Schlumberger製)を用いて開発した。

4) CAPICOM

検証時におけるパスの構築(Chainクラスのbuildメソッド)、証明書検証(CertificateクラスのisValidメソッド)及び署名時におけるキーストア処理部分、証明書型変化に利用

3. ライブラリ設計

.NET Framework クラスライブラリには、XML電子署名のためのクラスがいくつか用意されている。主なクラスは、SignedXmlクラスである。但し、このクラスはW3Cの仕様に記述された、Enveloping型、Enveloped型署名を直接生成するクラスとはなっていない。また、署名生成等の処理をICカード内の鍵ペアに対応させる機能はない。そのため、本ライブラリでは、この2点を解決するため、SignedXmlクラスを使い、新たなクラスライブラリの開発を行った。クラス構成は以下のようなものとした。

3.1 署名クラス

1) XmlSigner

XML電子署名生成用の抽象クラス。(図1参照)

2) EnvelopingXmlSigner

Enveloping型XML電子署名を生成するための、XmlSignerの実装クラス。(図1参照)

3) EnvelopedXmlSigner

Enveloped型XML電子署名を生成するための、XmlSignerの実装クラス。(図1参照)

3.2 キーストアクラス

1) SigningKeyStore 秘密鍵、公開鍵証明書の格納されたキーストアをオープン、クローズし、署名用証明書を選択するための抽象クラス。(図2参照)

2) SigningKeyStoreWithCapicom

CSP(Cryptographic Service Provider)を通して、ICカードに格納された秘密鍵、公開鍵証明書にアクセスするSigningKeyStoreの実装クラス。但し、Standard-9(三菱製)用CSPは一部機能が対応していないため、Standard-9(三菱製)には対応していない。(図2参照)

3) SigningKeyStoreWithStd9m

CSPを通して、Standard-9(三菱製)ICカードに格納された秘密鍵、公開鍵証明書にアクセスするSigningKeyStoreの実装クラス。一部、pkcs#11にて実装。(図2参照)

3.3 検証クラス

XmlVerifier

署名タイプを指定して、XML電子署名の検証を行うクラス。証明書の失効状態の確認のタイプとして以下の3つのタイプを用意し、柔軟な対応を可能とした。

1) CRLDistributionPointを確認

2) ストアに登録されたCRLを確認

3) 失効状態の確認をしない。

4. 今後

4.1 署名ライブラリの拡張

今回開発したライブラリは、ICカード内に署名用証明書を一つだけ格納した場合に対応したものである。今後、一つのICカード内に複数証明書が格納されることが予想される。署名用証明書、認証用証明書、暗号用証明書等、鍵使用目的の異なる証明書或いは、異なるドメインから発行される証明書等様々なケースが考えられる。KeyUsage、Issuer、Subjectといった証明書フィールドから適切な鍵ペアを選択し、署名を施すように対応を考えていきたい。

4.2 署名アーカイブシステム

このライブラリを利用した仕組みとして、電子署名をアーカイブするシステムの検討も進めている。既存、新規を問わず様々なシステムで生成された電子署名を受け、システム間での署名の流通を仲介する仕組みである。仲介させるための基盤としては、B to Bにおけるデータ、アプリケーションの統合サービスであるMicrosoft BizTalk Serverを考えている(図3参照)。異なる医療情報システムを連携させる仕組みについて、電子署名という視点から模索していく。

4.3 署名の長期保存

このライブラリは、W3CのXML-Signature Syntax and Processingに沿ったXML電子署名を、ICカードに格納された秘密鍵、公開鍵証明書を使い容易に作成するために開発した。しかし、この仕様は単純にXMLデータに電子署名を施すに過ぎず、署名が失効した場合や有効期限を過ぎて署名が保存される場合については対応できていない。現在、署名データの長期保存の仕様として、ETSI(the European Telecommunications Standards Institute) TS 101 733、RFC3029 DVCS(Data Validation and Certification Server Protocols)が存在し、この仕様をベースとした電子署名長期保存に関するレポートや商用サービスはいくつか見受けられる。しかし、これらはASN.1による電子署名の長期保存に関するものである。XML電子署名の長期保存に関する仕様としては、ETSI TS 101 903(XAdES)が存在するが、これに関する取り組みは、レポート、実装ともに現時点であまり見受けられない。平成13年度からは電子署名法が施行され、電子署名は法的な有効性を立証する一つの指標となった。長期に審議される裁判では、その公判に用いるデータの信憑性が長期間保証されることが求められる。そこで用いられる電子署名も必然的に長期保存対応が求められる。病院で用いられるカルテ等は、法的に数十年単位でのデータの保存が要求されることがある。XML化が進む電子カルテを考えた場合、

XML電子署名の長期保存は重要課題の一つといえる。但し、長期保存されるデータに対して、XAdESのように全てXML形式で長期保存する必要があるか否かは議論の余地がある。リアルタイムに流通されるのであれば、現在主流となりつつあるXML形式で署名を作成する意味は大きい。しかし、数十年後にデータがXML形式で扱われている保証はない。上記を踏まえた上で、今後の署名ライブラリ的设计を検討していく。

4.4 署名検証

一般に署名検証処理は容易ではない。署名検証ポリシーにより差異はあるが、処理内容として、署名検証、証明書有効性検証、証明書パス検証といったことが必要となる。電子署名の長期保存の場合、さらに署名検証は複雑となる。署名の長期保存技術は複数存在するが、ETSITS 101903で見ると、署名検証、証明書有効性検証、証明書パス検証、タイムスタンプの確認、タイムスタンプに署名がついている場合(RFC3161)は、その証明書の有効性検証及びパス検証を行う。さらに全体に施されたタイムスタンプ署名の検証をする。全体へのタイムスタンプ署名は更新されており、その更新毎に検証処理が必要となる。これらの証明書有効性検証は、検証時の失効情報ではなく、その証明書で署名された時点での失効情報を元に判断する必要がある。検証者は、こうした処理をオプション事項含め、署名検証ポリシーに沿って実施しなければいけない。署名検証ポリシーによって検証動作が変化すること等を考慮すると、実運用モデルとして、検証処理はVA (Validation Authority)と呼ばれる第三者に代行させるのが望ましいのかもしれない。RFC3029として、DVCS (Data Validation and Certification Server Protocols) が、その標準として存在するがXMLに対応した仕様とはなっていない。XMLに対応したVAを考慮したシステムについては、今後ユースケースを想定して議論する必要がある。XKMS (XML Key Management Specification) といった、公開鍵証明書情報の有効性検証サービス及び公開鍵証明書情報の管理サービスについての仕様は存在する。XML電子署名の検証サービスについてはまだドキュメントはほとんどなく、OASISのDigital Signature Services TCにて取り組みが始まったばかりである。今後標準化動向を見ながら検討を進める。

4.5 署名ポリシー

署名、検証は、各ドメインに規定されたポリシーに従って行う必要がある。そのポリシーはシステムに組み込むこととなるが、ドメインをまたがったシステム間で署名を流通させる場合、ポリシーの差異が問題となる。署名、検証ポリシーの標準フォーマットとして、CMS

ベースのETSI TS 101 733やXMLベースのETSI TR 102 038が存在するが、これらの仕様はまだ開発途中である。特に、ETSI TR 102 038は、ETSI TS 101 733をXMLに単純に変換したものである。ポリシーについても標準化動向を調査しながら検討を進める。

4.6 タイムスタンプ

公開鍵証明書には有効期限があり、鍵は有効期限前に失効されることがある。そのため、署名がいつ行われたのかを明確することは電子署名を運用する上で欠かせない。XML電子署名にタイムスタンプを付与するための標準的な仕様は現在ないが、その検討は、OASISのDigital Signature Services TCにて進められている。これは、RFC3161をベースに検討されている。そのため、タイムスタンプ自体に署名の有効期限が存在し、その長期保存も考える必要がある。これとは別に、タイムスタンプを実現するものとして、リンキングプロトコルが存在する。タイムスタンプ局が発行したタイムスタンプをリンクさせることにより、改竄を困難にするものである(一つのタイムスタンプを改竄する場合、リンクされた全てのタイムスタンプを改竄する必要があり、実質的に不可能に近い)。タイムスタンプは、時刻を付与したいデータのハッシュ値、時刻情報、リンク情報等から計算される。Digital Signature Services TCにおいても、リンキングプロトコルについての検討はされている。今後、OASISを中心に、その他の標準化動向を見ながらタイムスタンプの扱いについて検討を進める。

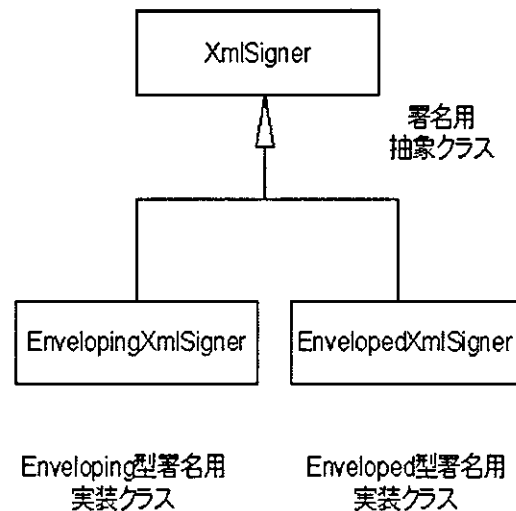


図1 署名クラス図