HL7 FHIR: An Agile and RESTful Approach to Healthcare Information Exchange

Duane Bender, P.Eng.
Dept. Electrical and Computer Eng. Technology
Mohawk College
Hamilton, ON, L8N 3T2, Canada
duane.bender@mohawkcollege.ca

Kamran Sartipi, PhD, P.Eng.
Dept. Electrical, Computer, and Software Eng.
University of Ontario Institute of Technology
Oshawa, ON, L1H 7K4, Canada
Kamran.Sartipi@uoit.ca

Abstract

This research examines the potential for new Health Level 7 (HL7) standard Fast Healthcare Interoperability Resources (FHIR, pronounced "fire") standard to help achieve healthcare systems interoperability. HL7 messaging standards are widely implemented by the healthcare industry and have been deployed internationally for decades. HL7 Version 2 ("v2") health information exchange standards are a popular choice of local hospital communities for the exchange of healthcare information, including electronic medical record information. In development for 15 years, HL7 Version 3 ("v3") was designed to be the successor to Version 2, addressing Version 2's shortcomings. HL7 v3 has been heavily criticized by the industry for being internally inconsistent even in it's own documentation, too complex and expensive to implement in real world systems and has been accused of contributing towards many failed and stalled systems implementations. HL7 is now experimenting with a new approach to the development of standards with FHIR. This research provides a chronicle of the evolution of the HL7 messaging standards, an introduction to HL7 FHIR and a comparative analysis between HL7 FHIR and previous HL7 messaging standards.

Keywords: eHealth; Interoperability; Healthcare; Standards; FHIR, HL7 v3; Agile, RESTful; Informatics.

1 Introduction

Interoperability is widely acknowledged as a key requirement for the success of healthcare information systems. Successful interoperability delivers economic value - the economic cost benefit of interoperability of healthcare information systems in the USA alone has been estimated at USD$77.8B [17]. Experts have publicly demanded proven interoperability as a means of increasing public safety.There is obvious clinical value in having the best information available when treating patients.

Interoperability is achieved in part by the use of consistent standards that define the syntactic and semantic meaning of information. Consistent, implementable standards reduce the risks, costs and timelines of health technology development projects.

Industries such as the financial industry have achieved high degrees of integration, interoperability and automation in part due to adherence to standards set by organizations such as SWIFT [4]. Organizations such as HL7 have facilitated a degree of integration in healthcare but widespread interoperability has not yet been achieved. Although tremendous resources have been invested to date by industry and jurisdictional health programs around the world, the goal of interoperability has remained elusive in the healthcare industry.

HL7 has recently embarked in the development of a new standard referred to as Fast Healthcare Interoperability Resources or "FHIR". FHIR is a departure in both process and product from previous HL7 messaging standards such as HL7 v2 and v3. The FHIR development process itself employs an incremental, iterative approach to develop the standard reflective of today's industry best practices for complex systems design. There is a deep focus by the FHIR development team on the usability and fitness for purpose of the end-product. The authors are contributors to the development of HL7v3 and FHIR and have built reference implementations of each platform. The v3 experience was discussed previously in [8]. FHIR implementation experience on the Apple iOS platform will be discussed in another paper currently under development.

This paper provides a chronicle overview of the development of the HL7 messaging standards from v2 to v3 and to the newly proposed HL7 FHIR. We provide an introduction to FHIR and its related dependant technologies and we contrast FHIR with previous versions of HL7 messaging standards.

2. Related Work

Several authors report on the advantages of HL7 v3 over v2 in terms of support for semantic interoperability and model driven aspects. However, they also report on the complexity of v3 message development process and the high-cost of its message communication. Mead [13] presents the differences between v2 and v3 in terms of: cross-domain information model, data type specification, and top-down development model of v3. Vernadat [1] proposes an interoperable enterprise system with focus on services and processes. The author investigates interoperability at technical, semantic, and organizational levels and identifies the necessity of using standard ontology to reach semantic interoperability.

There is an increasing trend towards standard-based integration of legacy healthcare systems that leverage emerging technologies such as web services, service oriented architecture, and Enterprise Service Bus (ESB) [12], [6]. Liu et al. discuss an integration project based on HL7 v2 to establish interoperability between a hospital information system (HIS) and a picture archiving and communication system (PACS) based on DICOM [12]. They propose an information exchange gateway between DICOM and HL7 v2. Mirth is an open source healthcare messaging integration engine developed by WebReach Inc. [5]. Mirth is based on client-server and enterprise service bus architecture and consists of connector, filter, and transformer modules to send/receive, parse, and transform messages from HL7 v2 to legacy formats. Sujansky et al. [16] report on a work for providing a constrained HL7 implementation guide for ambulatory Electronic Health Record (EHR) systems. A group of 15 representatives from ambulatory EHR vendors, clinical laboratories, physician organizations, governmental agencies, and HL7 target to reduce the complexity of the HL7 standard by constraining the optionality in HL7 v2.

The integration projects in the healthcare domain scale from integration of a hospital's subsystems to proposing general communication framework in distributed environments [18], [11], [9], [6]. However the architectures and configurations of the proposed models vary from project to project. Our research team have been involved in several HL7 interoperability projects, as follows. Sartipi and Yarmand [15] present the challenges in following the standard HL7 v3 message development platform to integrate legacy healthcare information systems. They present semantic interoperability of a Clinical Decision Support System (CDSS) with the EMR of a specialist. Jayaratna and Sartipi [10] use semantic web technology and developed a tool called TAMMP to identify HL7 v3 messages that perform healthcare transactions in an integration scenario. The technique provides a new categorization of HL7 v3 message functionality according to a set of message contexts extracted from HL7 v3 information hierarchies and messaging infrastructure. These contexts allow for mapping the key terms in a healthcare scenario to the corresponding HL7 v3 messages. Dehmoobad and Sartipi propose to use the core of HL7 v3 messages and expand it to provide cross-domain interoperability between healthcare and insurance organizations [14].

A number of HL7 based tools have been developed to assist different stakeholders such as researchers, healthcare professionals, industrial organizations, and the public. These tools are specialized for different tasks. Some of these tools are as follows. R-MIM Designer is a graphic designer that supports design of HL7 static information models [3]. V3 Generator accepts the XML expressions of an HMD and generates HL7 artifacts, including: static schema, interaction schema, HTML table views, and MIF files for static models, data types, and vocabularies [3]. HL7 v2 & v3 mapping is an Eclipse-based tool for semantic mapping and data transformation, with extensions for HL7 v2 and v3. It allows to map, translate, and test HL7 v2 or v3 to each other and to other formats [3]. TAMMP translates high-level healthcare scenarios into proper HL7 v3 messages [10].

3. Interoperability Standards in Healthcare

A single, well-adopted standard for healthcare information exchange has been an elusive goal. Many competing standards exist today and even more are in development. Although no single standard has emerged as a clear winner, Health Level 7 (HL7) Standards are a popular choice for both local communities such as local integration networks and hospitals, as well as regional and national eHealth initiatives. Several organizations play various roles in the development of interoperability standards in healthcare. There are standards development organizations such as HL7 and DICOM, but also groups such as Integrating Healthcare in the Enterprise (IHE) which aims to provide coordinated, consistent use of the various standards available, as opposed to the creation of new standards.

Health Level 7 (HL7)
Health Level 7 (HL7) is a not-for-profit standards development organization that was established in 1987 to develop standards for hospital information systems. Today, HL7 is an international community of health information experts that collaborate to develop standards for the exchange of health information and health systems interoperability.

HL7 produces both electronic system to system messaging standards as well as other standards such as electronic document structure and content standards to support systems interoperability. The messaging standards are available as HL7 v2 and HL7 v3. This research focuses on the messaging standards available from HL7.

HL7 Version 2

HL7 v2 was initially developed in 1989 in an ad hoc fashion to integrate various hospital systems together such as administrative and clinical systems. For example, hospitals quite often purchase separate billing and laboratory systems which cannot communicate automatically upon installation. HL7 v2 is a well adopted standard in hospitals and local communities, and is supported by most vendors of health information systems in North America.

The ad hoc nature of HL7 v2 meant that it did not scale well into larger multi-party environments such as jurisdictional information systems; for example, HL7 v2 lacks inherent support for global enterprise identifiers. HL7 v2 also has a heavy reliance on local customization through the use of so-called ”Z-segments”. System interfaces are designed to be 80% defined through the HL7 specifications and 20% customized by the local implementation. A second drawback of HL7 v2 is the lack of a formal ontology unifying the concepts being exchanged across various messages and interfaces.

HL7 Version 3

The development of HL7 v3 was started in 1995 to address the shortcomings of HL7 v2. HL7 v3 was not an incremental improvement to v2, rather, it was a decided departure. HL7 v3 introduced the HL7 Development Framework (”HDF”) development process and a central information model called the Reference Information Model or ”RIM”. The RIM defined the structure upon all of the semantic and lexical elements of the HL7 v3. The process makes use of a top-down Model Driven Architecture techniques to develop the clinical information models and attempted to automatically generate the messaging artifacts. The process is decidedly platform agnostic and decoupled from implementations. The documentation of the HDF development process alone is 144 pages.

The published HL7 v3 standards are not directly implementable and require substantial tooling for the generation of executable software systems. Although generated XML schemas are bundled with the distribution of the HL7v3 standards they are official not ”normative” to the standard. Although partially useful for the verification of an XML instance message based on the standard, they are not suitable for other implementation tasks such as the automated generation of software classes. Implementation of the HL7 v3 clinical models requires complex model transformations into platform specific models, a task similar to that of a custom compiler. No such tool is provided by HL7. The author led the development of such a tool which took approximately 18,000 man hours.

Implementation of HL7 v3 requires a detailed understanding of the RIM model. The RIM seeks to structure all information as: “*entities*” in “*roles*” “*participating*” in “*acts*”. Model development in HL7 v3 is a process of ”design by RIM constraint”. This is quite abnormal in the practicing software industry. Design by constraint (reduction) does not translate well to implementation languages such as object oriented platforms which are typically ”design by composition” (addition) paradigms. HL7 architects have commented that v3 added design rigour to v2, however v3 flows all design rigour to the transport level in complex messaging of model information. This makes for complex wire format transmissions.

From the HL7 website: ”The HL7 Reference Information Model (RIM) is single artifact that is the foundation from which all V3 and SAIF logical and implementable information models SHALL be derived. It serves to unify the information content of HL7 V3 standards, affording more consistent representation of models and simpler implementation of models from multiple topics. The RIM is a class model that includes the state machines, associations and attributes of those classes.”

The ”universal” edition of the HL7 v3 standard is not designed to be directly implemented. The philosophy is that the models are starter templates that will then be constrained for use by the implementing region or party. This places a large burden of work on the implementer. It also virtually guarantees that no two real-world implementations will be compatible and blocks international software vendors and solution providers from being able to offer consistent solutions across regions and languages.

The contextual flow of HL7 v3 through it’s top-down modelled approach implied the context for interaction and was therefore elegant in its high level descriptions of roles and responsibilities of transaction partners (called ”application roles”). This pure top-down approach failed at the implementation stage though due to the explosion of complexity as the model transforms from the abstract to the concrete implementation artifacts.

HL7 v3 was adopted by several national governments (e.g., Canada, United Kingdom, Australia, Denmark, Germany, etc.) as a basis for national EHRs but was most notably not selected for use in the US as part of the HITECH act and thus did not qualify for Meaningful Use investment programs. The complexity of implementations was mentioned in large project failures such as the UK National Health Services EHR program and the elimination of HL7 v3 messaging from the US Food and Drug Administration clinical study data acquisition program.

Adding to implementation complexities, HL7 v2 and v3 are not directly interoperable between themselves. Interchange of HL7 v2 and v3 data requires the use of sophisticated translation software.

Although it claimed to be platform independent, HL7 v3 is modelled using an object oriented approach and the process output generates clinical information formats that are designed to be used in a message oriented middleware fashion. These design choices in a practical sense predetermine much of the way that the clinical models can be used.

Finally, HL7 v3 claims to offer semantic interoperability, but ironically due to the syntactic complexity, in the field there have been dramatic examples of misinterpretations leading to unaligned implementations and subsequent significant rework. One example of this is the incompatible implementations of the Ontario and Alberta Provincial patient registries and their dynamic behaviour with respect to identity merge operations, even though they were developed using the same standard specification.

Representational State Transfer (REST)
Representational State Transfer (REST) systems as described by Fielding [7] (often referred to as RESTful architectures) have recently been widely adopted as the dominant information abstraction of the WorldWideWeb . Resource orientation and the statelessness of the HTTP protocol are responsible for the decentralized highly scalable nature of the web. Recently major platforms of Internet services have migrated to RESTful services and away from other architectural choices such as SOAP and WSDL approaches. In particular an implementation of REST follows four basic design principles of: i) using HTTP methods explicitly; ii) being stateless; iii) exposing directory-structure like URIs to resources; and iv) transferring XML or JSON or both as resource representations.

The practical advantages of RESTful architectures include light-weight interfaces that allow for faster transmission and processing of data structures, more suitable for mobile phones and tablet devices. RESTful interfaces also facilitate faster development cycles through their simpler structures.

Resources and their base operations very closely resemble the operation of a relational database - information structures are defined, operations on creating, updating, deleting and retrieving data is defined but the meaning of the data in an application is defined at a higher level.

Agile Methods
Agile Methods are a group of software development methods that favour incremental and iterative approaches to the development of software and systems. Many specific processes have been developed as implementations of the agile approach.

Although agile methods have been traced back many decades, the modern agile development has been influenced by the publication of the Agile Manifesto by a group of prominent software engineers including Martin Fowler [1].

The AgileManifesto website states that the principles of Agile Development are: ”Individuals and interactions over processes and tools”; ”Working software over comprehensive documentation”; ”Customer collaboration over contract negotiation”; and ”Responding to change over following a plan”. Agile methods build quality directly into a product by subjecting the product to continued and constant consume-ability, usability and defect testing by early adopters.

4 FHIR: Fast Healthcare Interoperability Resources

Based on pressure from the lack of implementations of HL7 v3, including a lack of eligibility for US HITECH funding, in January 2011, the HL7 Board of Governors initiated a ”fresh look task force” to examine how HL7 messaging standards could be improved. This inspired an independent group of HL7 architects to begin discussing a new approach to healthcare information exchange which they initially called ”Resources for Health - RFH”, which was later renamed to “Fast Healthcare Interoperability Resources - FHIR” [2]. This new approach was based on the RESTful principles described by Fielding [7].

The FHIR effort aims to simplify and accelerate HL7 adoption by being easily consumable but robust, and by using open Internet standards where possible. Using an easily consumable format for the standard avoids the need for complex custom tooling. The standard contains implementation examples for all artifact and reference implementations for several platforms, including live test servers available over the Internet. The authors claim that HL7 v3 was not thrown away but that FHIR was created through the learning of v3.

FHIR Development Process
Whereas HL7 v2 messaging followed an ad hoc development process and v3 had followed a rigidly defined top-down model-driven process referred to as the HDF, FHIR uses an incremental and iterative approach to the development of the standard. Incremental development refers to the development of small pieces at a time. The incremental pieces are implemented and tested and the learning are fed back into the design process very quickly. This is done over and over, i.e., the iterative part of the process.

FHIR Artifacts
FHIR aims to define the key entities involved in healthcare information exchange as resources. Each resource is a distinct identifiable entity. The FHIR specification describes the following attributes of resources:

- Resources should have a clear boundary, that matches one or more logical transaction scopes.
- Resources should differ from each other in meaning, not just in usage (e.g., different ways to use a lab report should

not result in different resources).
• Resources need to have a natural identity.
• Resources should be very common and used in many different business transactions.
• Resources should not be specific or detailed enough to preclude support for a wide range of business transactions.
• Resources should be mutually exclusive.
• Resources should use other resources, but they should be more than just compositions of other resources; each resource should introduce a novel content.
• Resources should be organized into a logical framework based on the commonality of the resource and what it links to (see resource framework below).
• Resources should be large enough to provide meaningful context; resources that contain only a few attributes are likely too small to provide meaningful business value.

Example resources include: *Patient, Device* and *Document*. At the time of this writing there are 32 resources defined with many more under consideration. The development team estimates that there will be approximately 150 resources defined in total. The designers of the FHIR process have stated that "FHIR attempts to maintain design rigour of v3 but make resource representation and transport simple". In HL7 v3 common reusable class structures were defined as *Common Message Element Types* or CMETs. CMETs are somewhat comparable to Resources in FHIR. The number of CMETs in the 2010 Normative Edition of the HL7 v3 specification was 194, while the number of Resources defined in FHIR as of Feb 2013 (after roughly 18 months of development) is approximately 32. This represents a drastic reduction in the number of concepts requiring implementation.

Criticism of FHIR

As a resource oriented environment FHIR allows for very simple implementation of base artifacts, their transmission and persistence. However there is very little guidance as to how these base resources are to be constructed into larger collections and relationships. There is also no support for workflow and dynamic behaviour beyond base CRUD operations. This may become an area of divergence leading to a lack of interoperability.

Collections and feeds of resources are defined in the base standard but sophisticated resource aggregations (e.g., invoices) and how they are to behave is not defined. There is a grey area between resources aggregations and documents that is not yet defined.

A hybrid approach

A chasm exists between information modellers and systems implementers. Coming from the top down approach modelling the universe, complexity explodes at the concrete level. Starting from the concrete and trying to move up to the meta-model in the abstract creates a case-by-case design. It seems to be ideal to incorporate the strength of HL7 v3 such as RIM and messages into FHIR, and propose a "process" for it, i.e., similar to HL7 message development framework.

Statements from stakeholders in public presentations claim that FHIR has a "strong ontology" behind the scenes that links RIM and vocabulary. However, there is no clear documentation describing such an ontology or process being used in the public documentation. FHIR actually appears quite "granular" in the sense that resources are individually defined as discrete elements; however there is no obvious cohesion to a unified model. This allows for the ultimate flexibility but it is a case by case example model with no meta-model to draw from.

Mapping of FHIR resources to the HL7 v3 RIM is mentioned as a key component of the FHIR methodology in the specification, and that the design rigour of HL7 v3 is being maintained in the FHIR development process. The rationale for this is unclear as there are few if any real-world implementations of HL7 v3. There is no clear description of what the abstract RIM mapping means or how it is to be used in the concrete implementations of FHIR. In the current specification at the time of this writing many of the RIM mappings in the FHIR specification are entered as "N/A" or resemble OCL constraints. The author questions the usefulness of these mappings. The base specification lacks the ability to model complex workflows and declines to describe more than a few events and dynamic behaviour patterns. A few events are defined (9 at the time of writing) but this is primitive and inadequate to implement in the real world. Table 1 provides a comparison of the salient properties of the HL7 v2, v3, and FHIR standards.

5. Conclusion

This paper presents a chronicle study of the evolution of the HL7 standards for interoperability among healthcare systems. The attributes of these standards (HL7 v2, v3, and FHIR) were studied with an analytical theme. A comparative table was provided that highlights the salient properties of each standard that allows to compare the strengths and weaknesses of each standard. While still under development, the FHIR standard has attracted a lot of attention from the relevant community due to its simplicity (following the RESTful architecture), bottom up and consumer-based development model (as opposed to model-based). There are still ambiguities on how the HL7 FHIR standard should utilize the strengths of its predecessors (HL7 v2 and v3). However, clearly the authors of FHIR will take advantage of the vast experiences gained from the implementation of its ancestors, which will drastically improve the state of information communication among healthcare systems. Fu-

Table 1. Comparison of FHIR with HL7 V2 & V3

Property	**HL7 v2**	**HL7 v3**	**HL7 FHIR**
Year initiated	1987	1997	2011
Development Process/methodology	Bottom up / ad hoc	Top-down, MDA	Iterative and incremental
Architectural paradigm	Message, Fields and records	Message-Oriented	RESTful
Semantic Ontology	No	Yes	Yes?
Learning overhead	Order of weeks	Order of months	Order of weeks
Specialized tooling required?	Yes - parser	Yes - model compiler	No
Directly consumable?	yes	no	yes
Order of size of specification	hundreds of pages	thousands of pages	hundreds of pages
Implementation examples in specification	yes	minimal	yes
Reference implementations available from HL7	no	no	yes
Industry and community support	strong	weak	n/a - too new
Inherently suitable for mobile devices	no	no	yes
Number of message types	?	450	30
Degree of adoption	Very high	Very low	n/a
Information model type	ad hoc	constrained RIM	?
International character support	no (ASCII)	conceptually yes	yes (UTF8)
International message format support	single global standard	localized by realm	single global standard

ture work includes completing a reference implementation of FHIR on the Apple iOS platform. The architecture and performance of this framework will be discussed in a paper currently under development.

References

[1] Agile software development . http://martinfowler.com/agile.html [20 February 2013].

[2] FHIR: Fast healthcare interoperability resources. http://hl7.org/implement/standards/fhir/ [20 Feb 2013].

[3] HL7 official website - list of tools . http://www.hl7.org/participate/toolsandresources.cfm?ref=nav.

[4] SWET: The global provider of secure financial messaging services. http://www.swift.com/standards [20 Feb 2013].

[5] G. Bortis. Experiences with Mirth: An open source health care integration engine. In *International conference on Software engineering*, pages 649–652, 2008.

[6] T. H. Y. et al. A scalable multi-tier architecture for the national taiwan university hospital information system based on HL7 standard. In *Symposium on Computer-Based Medical Systems*, pages 99–104, 2006.

[7] R. T. Fielding. *Architectural Styles and the Design of Network-based Software Architectures*. PhD thesis, University of California, Irvine, 2000.

[8] J. Fyfe, D. Bender, and H. K. Edwards. Everest: A framework for developing HL7 v3 applications. *ACM SIGHIT Record*, 2(1):24–24, March 2012.

[9] S. Hsieh et al. Middleware based inpatient healthcare information system. In *Bioinformatics and Bioengineering*, pages 1230–1234, 2007.

[10] P. Jayaratna and K. Sartipi. HL7 v3 message extraction using semantic web techniques. *Inernational Journal of Knowledge Engineering and Data Mining*, 2(1):89–115, 2012.

[11] L. F. Ko et al. HL7 middleware framework for healthcare information system. In *e-Health Networking, Applications and Services*, pages 152–156, 2006.

[12] B. Liu, X. Li, Z. Liu, Q. YUuan, and X. Yin. Design and implementation of information exchange between HIS and PACS based on HL7 standard. In *Information Technology and Applications in Biomedicine*, pages 522–555, 2008.

[13] C. N. Mead. Data interchange standards in healthcare it-computable semantic interoperability: Now possible but still difficult, do we really need a better mousetrap? *Journal of Healthcare Information Management*, 20(1):71–78, 2006.

[14] K. Sartipi and A. Dehmoobad. Cross-domain information and service interoperability. In *ACM iiWAS*, pages 25–32, 2008.

[15] K. Sartipi and M. Yarmand. Standard-based data and service interoperability in ehealth systems. In *IEEE International Conference on Software Maintenance (ICSM)*, pages 187–196, 2008.

[16] W. V. Sujansky et al. The development of a highly constrained Health Level 7 implementation guide to facilitate electronic laboratory reporting to ambulatory electronic health record systems. *Journal of the American Medical Informatics Association*, 16(3):285–290, 2009.

[17] J. Walker et al. The value of health care information exchange and interoperability. *Health Affairs*, pages w5–10 – w5–18, 2005.

[18] W. Wang, M. Wang, and S. Zhu. Healthcare information system integration: a service oriented approach. In *Services Systems and Services Management*, pages 1475–1480, 2005.