Open Standards for Data Exchange in Healthcare Systems

Seminar Thesis in E-Health
at the
Faculty of Economics and Social Sciences
University of Fribourg

Prof. Dr. Andreas Meier
Department of Informatics DIUF

Vogt, Joël
Matriculation number: 03-216-553
e-Mail: joel.vogt@unifr.ch

and

Wittwer, Daniel
Matriculation number: 03-201-258
e-Mail: daniel.wittwer@unifr.ch

Submitted on: 7.12.2007
## Contents

1 Introduction .................................................................................................................. 1
   1.1 Structure ................................................................................................................. 1

2 E-Health defined ............................................................................................................. 2
   2.1 The 10 e’s of E-Health ......................................................................................... 2
   2.2 Stakeholders in the Healthcare System ............................................................... 4
   2.3 Information Management .................................................................................... 4

3 The Importance of Standardization ............................................................................... 6
   3.1 Data Integration ................................................................................................. 6
   3.2 Ontological Compatibility with Health Level 7 .................................................. 7
   3.3 Terminological Compatibility with the Unified Medical Language System .......... 11
      3.3.1 Metathesaurus ............................................................................................. 11
      3.3.2 Semantic Network ...................................................................................... 12
      3.3.3 Specialist Lexicon ...................................................................................... 12
      3.3.4 MetamorphoSys ......................................................................................... 13

4 Case Study: Mirth .......................................................................................................... 14
   4.1 Use Case .............................................................................................................. 14
   4.2 Mirth ................................................................................................................... 14
   4.3 Implementation ................................................................................................... 16

5 Integration through Service-Orientation ........................................................................ 17
   5.1 Building Blocks of a Service Oriented Architecture .......................................... 19
      5.1.1 Legacy Systems .......................................................................................... 19
      5.1.2 Services ..................................................................................................... 20
      5.1.3 Enterprise Service Bus (ESB) .................................................................... 20
   5.2 SOA and HL7 ...................................................................................................... 20
      5.2.1 Combining SOA Principles and Mirth ......................................................... 21

6 Conclusion ...................................................................................................................... 23
   6.1 Outlook ............................................................................................................... 24

List of Figures ...................................................................................................................... 25

Bibliography ....................................................................................................................... 26
1 Introduction

The objective of this seminar thesis is to illustrate the need for comprehensive information management and technical integration strategies as one of the important steps towards efficient intra- and interorganizational and IT-supported healthcare processes. Possible solutions to the topic are discussed by differentiating the need for semantically and ontologically standardized data from the standards and approaches used to exchange the information between various stakeholders and actors in medical processes.

As a matter of complexity of the field of healthcare and e-health and the various solutions already proposed by different institutions, this thesis focuses on two specific developments in the area which are Health Level 7 (HL7) and Service Oriented Architecture (SOA). The combination of the HL7 standard together with the principles of SOA can provide a basis and possible solution for automated, efficient and collaborative e-health processes.

There exists a variety of important non-technical but rather political and cultural requirements for a successful implementation of national e-health environments. These aspects are not explicitly in the scope of this thesis but are in some cases mentioned to gain a broader view of the field.

1.1 Structure

After the introduction in chapter 1, the term e-health, the stakeholders in a healthcare system and the need for information management is discussed in chapter 2.

To enable continuous medical processes, data standards must be defined and implemented by the institutions sharing medical information. As a possible approach to data standardization, the Health Level 7 (HL7) standard is discussed in chapter 3.

In chapter 4, the asynchronous HL7 messaging middleware Mirth is presented. The case study elaborates on the HL7 generation and transformation capabilities offered as well as on the underlying architectural pattern of channels.

The concept of services and the Service Oriented Architecture (SOA) is introduced in chapter 5. SOA represents a modern architectural approach for building flexible IT infrastructures, where the most anticipated values are improved integration capabilities, both technical and organizational. The chapter finishes with a proposition, how an SOA approach can be combined with the healthcare-specific requirement of HL7 content exchange.

The thesis finishes with a conclusion and outlook given in chapter 6.
2 E-Health defined

E-health is a comprehensive concept to position the use of ICT in a healthcare system. This implies that the technical feasibility is of second priority and thus the goal is not just to electronically implement existing processes. E-health initiatives rather focus on providing new ways of interaction between the stakeholders in the system allowing processes to be executed more efficient and effective. The major goal thus is to reduce complexity, save administrative costs and improve the quality of patient management and treatment. This implies organizational and political adoption as well as the introduction of nationally agreed standards, technologies and legal aspects.

Eysenbach [2001] defines e-health as “an emerging field in the intersection of medical informatics, public health and business, referring to health services and information delivered or enhanced through the internet and related technologies. In a broader sense, the term characterizes not only a technical development, but also a state-of-mind, a way of thinking, an attitude, and a commitment for networked, global thinking, to improve health care locally, regionally, and worldwide by using information and communication technology”.

2.1 The 10 e’s of E-Health

To illustrate the characteristics of the term, Eysenbach [2001] introduces the following 10 e’s of e-health. They are cited in this thesis because they define e-health in a broader sense and implicitly and explicitly show the integration needs as a basis for improved collaboration and information management across healthcare professionals and patients as well as the political and sociological issues associated.

**Efficiency** “One of the promises of e-health is to increase efficiency in health care, thereby decreasing costs. One possible way of decreasing costs would be by avoiding duplicative or unnecessary diagnostic or therapeutic interventions, through enhanced communication possibilities between health care establishments, and through patient involvement” [Eysenbach, 2001].

**Enhancing quality** “Increasing efficiency involves not only reducing costs, but at the same time improving quality. E-health may enhance the quality of health care, for example by allowing comparisons between different providers, involving consumers as an additional power for quality assurance, and directing patient streams to the best quality providers” [Eysenbach, 2001].

**Evidence based** “E-health interventions should be evidence-based in a sense that their effectiveness and efficiency should not be assumed but proven by rigorous scientific evaluation. Much work still has to be done in this area” [Eysenbach, 2001].
Empowerment “of consumers and patients - by making the knowledge bases of medicine and personal electronic records accessible to consumers over the Internet, e-health opens new avenues for patient-centered medicine, and enables evidence-based patient choice” [Eysenbach, 2001].

Encouragement “of a new relationship between the patient and health professional, towards a true partnership, where decisions are made in a shared manner” [Eysenbach, 2001].

Education “of physicians through online sources (continuing medical education) and consumers (health education, tailored preventive information for consumers)” [Eysenbach, 2001].

Enabling “information exchange and communication in a standardized way between health care establishments” [Eysenbach, 2001].

Extending “the scope of health care beyond its conventional boundaries. This is meant in both a geographical sense as well as in a conceptual sense. e-health enables consumers to easily obtain health services online from global providers. These services can range from simple advice to more complex interventions or products such a pharmaceuticals” [Eysenbach, 2001].

Ethics “e-health involves new forms of patient-physician interaction and poses new challenges and threats to ethical issues such as online professional practice, informed consent, privacy and equity issues” [Eysenbach, 2001].

Equity “to make health care more equitable is one of the promises of e-health, but at the same time there is a considerable threat that e-health may deepen the gap between the "haves" and "have-nots". People, who do not have the money, skills, and access to computers and networks, cannot use computers effectively. As a result, these patient populations (which would actually benefit the most from health information) are those who are the least likely to benefit from advances in information technology, unless political measures ensure equitable access for all. The digital divide currently runs between rural vs. urban populations, rich vs. poor, young vs. old, male vs. female people, and between neglected/rare vs. common diseases” [Eysenbach, 2001].
2.2 Stakeholders in the Healthcare System

The healthcare system consists of a wide and complex variety of different classes of stakeholders. A stakeholder in this system can be defined as an organization (e.g. a hospital) or an individual person (e.g. a patient) which is actively or passively involved in healthcare processes and thereby acting as a source, broker or consumer of specific healthcare-related information and/or goods.

As e-health initiatives target at the interorganizational exchange of electronically stored information between the participants, i.e. along the processes in the system, the complexity of the integration task becomes obvious.

Figure 2.1: Stakeholders in a Healthcare System on the basis of Haas [2006, p.245].

2.3 Information Management

The services provided and tasks executed by the stakeholders in a healthcare system rely on information (i.e. patient-related or disease-related data). As there exist processes that span over multiple actors, i.e. stakeholders in the system, information is modified and often needs to be shared between different stakeholders in the system. Regarding healthcare services, this means that the work result of a task executed by a stakeholder often relies on information previously processed by a different stakeholder. As an example, the billing process in a health insurance company needs as an input information provided by the hospital or healthcare professional on the treatment of the patient.

As this information is often produced and treated in different organizational units, i.e. institutions, it is hence often stored in different information systems with different underlying technologies, database schemes and coding formats.

Tan [2005, p.164] states that healthcare information systems have traditionally been designed as autonomous applications using widely varying formats for the description of
patient-related data “impeding [an] integration of information from different sources”. This results in inefficient “discontinuities in the care process among caregivers” [Tan, 2005, p.164]. To enable efficient and effective healthcare processes along the stakeholders which are consequently supported by information and communication technologies (ICT), it is therefore necessary to integrate the underlying information systems.

Information management strategies and information integration between healthcare institutions, i.e. stakeholders, are therefore amongst the most cited challenges of today’s national healthcare systems and the underlying information systems, i.e. applications.

The challenge for the politics and the stakeholders is to establish nationally agreed-upon and mandatory specifications regarding coding and transmission standards for healthcare-related data. The successful establishment of a national framework for communication and data standards in the healthcare environment is the very basis for a sustainable and successful implementation of a flexible and powerful e-health infrastructure. Integration initiatives in the healthcare sector to support information management and electronic data exchange among its stakeholders shall - in the long run - result in increased process efficiency and lead times, improved quality of service, a reduction of complexity, increased flexibility, lower costs as well as enabling new and advanced services (e.g. mobile health concepts).

To further elaborate on the architectural requirements and possible solutions to application integration challenges in the healthcare domain, the authors are going to provide possible answers to the following two basic questions:

- How can technologically heterogenous information systems be integrated to support electronic information sharing and enable continuous business processes among the stakeholders in the healthcare system?

- What kind of data needs to be shared between stakeholders in the healthcare system and how can this data be represented in a standardized and platform independent manner, i.e. what needs to be standardized and how can it be done?
3 The Importance of Standardization

Sharing information among healthcare professionals [Chronaki et al., 2003, p.101] [Blobel et al., 2006, p.343] and information systems in a healthcare environment [Blobel et al., 2006, p.343] is drastically gaining of importance in order to increase effectiveness and efficiency of medical services. To achieve this, information systems need to be integrated and able to use messages, i.e. to interoperate. The communication and collaboration can happen at different levels of interoperability [Blobel et al., 2006, p.343] [Beyer et al., 2004, p.269].

This chapter focuses on the integration of heterogenous information systems in healthcare environments by focusing on the data integration aspect. This chapter is structured as follows: Aspects of data integration are described in section 3.1, section 3.2 suggests a messaging standard to support ontological compatibility among healthcare information systems and section 3.3.4 presents a framework for terminological compatibility.

3.1 Data Integration

To permit information systems with different data structure to interoperate, Beyer et al. [2004, 268] identified three aspects of compatibility, namely 1. Data structure and encoding (syntactical compatibility), 2. Type level semantics (ontological compatibility) and 3. Instance level semantics (terminological compatibility) [Beyer et al., 2004, 268]. They are introduced in this section.

**Syntactical compatibility:** Different systems might use different encoding rules to structure their data; thus syntactical transformation is necessary to exchange data. Beyer et al. [2004, p.268] refer to XML as their markup language of choice, because, they argue, it puts no constraints on platform independence and, with its multiple extensions such as XSLT, offers a powerful framework for data management and conversion. This aspect of data integration is not specific to e-health and will therefore not be addressed any further in this thesis.

**Ontological Compatibility:** Different applications are likely to use different information ontologies that are semantically incompatible. An information ontology, as defined by Abecker et al. [1998, p.44], “[…] comprises all aspects of information and knowledge sources that are not content-specific”. Semantic heterogeneity on the type level occur when the semantics are, as Beyer et al. [2004, 268] puts it, “hardcoded into an application”. This for example results from encoding certain concepts into a database schema. Semantic heterogeneity is a fundamental problem in information system integration since two semantically different information ontologies cannot be integrated automatically and in some cases even requires the modification of at least one database
schema [or other type of information ontology] [Beyer et al., 2004, 268]. This problem can only be resolved if different vendors agree on domain-specific, in the context of this paper, healthcare specific standards [Beyer et al., 2004, 268].

**Terminological Compatibility:** Semantic heterogeneity on the instance level corresponds to the content, e.g. when user enter semantically incompatible data into the system. A typical approach to address this problem is the use of controlled vocabulary such as standard catalogs or classifications [Beyer et al., 2004, 269]. Translations between these different terminologies can be done by ETL translation mechanisms [Beyer et al., 2004, 269].

### 3.2 Ontological Compatibility with Health Level 7

Section 3.1 briefly introduced compatibility problems at type level and stated that, in order for two or more information systems to interoperate, they need to agree on a common information ontology. This section will describe one standard to integrate different medical information systems at the type level.

Health Level 7 (HL7) is a well established message-based standard developed by the American National Standards Institute (ANSI) accredited standards developing organization Health Level 7 Inc. It aims to develop coherent and extensible standards for the exchange, management and integration of electronic information in the clinical and administrative domain [Health Level Seven, 2007].

*Level 7* of HL7 refers to the Application Level of the OSI seven layer model. This level describes how data are exchanged and the timing of the interchange as well as the handling communication of errors [Health Level Seven, 2007]. Over 90% of healthcare facilities in the USA [Shaver, 2007] already use HL7 and it is also used internationally [Health Level Seven, 2007] [Shaver, 2007]. According to Health Level Seven [2007], HL7 version 2.5 is the most widely implemented standard for healthcare information worldwide.

Before HL7 was introduced, clinical applications were not developed with interoperability in mind, i.e. there was no collaboration between various vendors and thus for two healthcare systems to interoperate, customized interfacing systems had to be developed [Metz, 2007, p.3] [Shaver, 2007].

HL7 develops and maintains two messaging standards: HL7 version 2.x and 3.0. HL7 version 3.0 is an improvement of version 2.x [Health Level Seven, 2007] but not backwards compatible. If version 3.0 was to be compatible with version 2.x, it would have had to include legacy issues that would make change difficult [Metz, 2007, p.8]. HL7 versions 2.x evolved over the years using bottom-up approach. It tried to address individual needs using an evolving ad-hoc methodology. The result is that neither the view of the data that are sent with HL7 nor the data’s relationship to other data are consistent [Health Level Seven, 2007].
HL7 contains many optional data segments and is thus very flexible. The downside to this however is that it is impossible to guarantee standard conformance of any vendors’ implementation. This has the regrettable consequence that vendors might need more time to analyse and plan their interface to assure that the same optional features are used by both parties [Health Level Seven, 2007]. The vagueness in the standard, the lack of a consistent application data model, a formal methodology to model data artifacts and the lack of well defined application user roles were issues addressed in HL7 version 3.0 [Metz, 2007, p.7]. It uses an object-oriented development methodology based on a data model, the Reference Information Model (RIM) to create messages. RIM provides an explicit representation of semantic and lexical connection that exists between the information transferred in HL7 version 3.0 messages [Health Level Seven, 2007].

This paper looks at e-health architectures from the more practical perspective, thus the main focus will be on version 2.5.1 of the HL7 standard, which is an ANSI standard and widely adopted. Version 3.0 is yet to be standardized. HL7 version 3.0 will only be discussed briefly.

**Health Level 7 Version 2.5.1**

The current HL7 standard addresses message-based interfaces among various healthcare IT systems to send or receive patient admission/registration, discharge or transfer data, queries, medical records, resource or patient scheduling, orders, results, clinical observations, etc [Accenture, 2007, p.3]. It does not prescribe which architecture, functionality, data schema, etc. an information system must implement. Accenture [2007, p.4] state that HL7 is suited for centralized as well as for distributed environments. Among the healthcare domains supported by HL7 version 2.x are a) decision support, b) emergency medicine, c) clinical genomic and d) pediatric.

Accenture [2007, p.13] explains that HL7 is not plug and play due to 1. the lack of process conformance within healthcare delivery environment and because 2. customers want to be able to negotiate with vendors.

Thus HL7 has to be very flexible and does not guarantee to offer all the interfaces needed, hence functionality might have to be added for a specific information system [Accenture, 2007, p.14]. Metz [2007, p.14] explains that HL7 version 2.x offers about 80% of the interface framework directly and lets the remaining 20% that are specific to a given application, be added. Thus HL7 version 2.x reduces but does not eliminate time and cost for interface implementation [Accenture, 2007, p.14] [Metz, 2007, p.5].

It is worthwhile to note the issues which HL7 version 2.5.1 does not cover: It does not make any specifications about security classifications, authentication and authorization as well as encryption of messages. It assumes that these mechanisms are already in place. Version 2.5.1 also does not attempt to define or support implicit or explicit roles and relationships between roles, e.g. patients and physicians. Furthermore, the HL7 version 2.x does not use
3 The Importance of Standardization

an explicit data model or composite data dictionary. These issues are however addressed in version 3.0 of the HL7 standard [Accenture, 2007, p.15-16]. This list is not exhaustive. For a full list of features not covered by the standard, the user is referred to the official standard documentation by Health Level 7.

**Triggers and Messages**

An event in the *real world* creates the need for data to flow among systems. In HL7, a *real world* event that triggers the flow of data is called a *trigger event*. Examples of trigger events are *patient is admitted* or *an observation for a patient is available*. They result in data on a given patient being transmitted among various healthcare information systems. For the information system that triggered the event to transfer data, the transaction is termed an *unsolicited update* [Grieve et al., 2007, p.4]. When a system conducts an unsolicited update, HL7 version 2.5.1 specification specifies that a message is acknowledged at the application level, since it is necessary to know that the application has processed the data. Thus simply knowing that the underlying communication system guarantees the delivery of the message is not enough. The extended version of the HL7 acknowledgment paradigm differentiates between two modes: 1. Accept acknowledgment, when the message has been received successfully and 2. Application acknowledgments that are sent when the data has been processed by the receiving application [Grieve et al., 2007, p.5].

The HL7 standard defines messages as the smallest atomic unit of data exchanged between systems. A message has a type that defines its purpose, e.g. *patient admission*. A message consists of a group of segments in a defined sequence. A segment is a logical group of data fields. They can be optional or mandatory. A segment may occur only once or may be repeated. Each segment has a name, e.g. the message header is called MSH [Grieve et al., 2007, p.6]. Messages are encoded in plain ASCII [Accenture, 2007, p.9]. HL7 resides in the application layer of the OSI model. It requires other lower level protocols to transmit the messages over the network [Grieve et al., 2007, p.5].

**Health Level 7 Version 3.0**

HL7 version 3.0 is not a continuation of version 2.x. It follows a new approach [Health Level Seven, 2007]. Metz [2007, p.4] explains the difference between both versions of HL7 as follows: “It is critical to recognize that HL7 V2 was initially created by clinical interface specialists while V3 has been mostly created by medical informaticists”. Unlike HL7 version 2.x, that evolved ad-hoc over time and did not follow a consistent methodology and data model, version 3.0 of the HL7 standard is built around a single object-oriented development methodology [Health Level Seven, 2007], called the HL7 Development Framework (HDF) [Blobel et al., 2006, p.345] and a formal data-model, the Reference Information Model (RIM). Also unlike version 2.x, HL7 version 3.0 uses XML [Health Level Seven, 2007] instead of plain ASCII [Accenture, 2007, p.9] to encode messages. Because HL7 version 3.0
The Importance of Standardization

Artifacts are based on a common data model, they are definite and testable, hence vendors’ conformance can be certified [Health Level Seven, 2007].

A detailed discussion of HL7 version 3.0 would be beyond the scope of this seminar paper. This chapter will give an overview of HL7 version 3.0 and discuss two new important elements of HL7 that were introduced with version 3.0, namely the Reference Information Model and the Clinical Document Architecture. The HL7 Development Framework and messages will not be covered. The reader shall note that, contrary to version 2.x, they are encoded in XML and are based on RIM [Health Level Seven, 2007].

Even though version 3.0 sounds promising, critics have voiced their concerns about the new standard. Smith and Ceusters [2006] argue in their article HL7 RIM: An Incoherent Standard that the RIM, despite being developed since 1997, has still major flaws such as internal inconsistencies, amateurish documentation and problems of implementation [Smith and Ceusters, 2006, p.1-2]. INTERFACEWARE [2007], a healthcare integration provider, adds that despite the long development time, no clear solution is in sight.

Reference Information Model

The Reference Information Model (RIM) is a large clinical data model and a central part of HL7 version 3.0. It identifies the life cycles of events that trigger a message or a group of messages. Messages from all clinical domains create their messages with the RIM [Health Level 7 Inc., 2005]. The RIM contains six core classes of objects in the healthcare domain as well as associations between them. Those classes are a) Entities: physical things (or groups of physical things) or organizations, e.g. organizations and living subjects. b) Roles: socially expected behavioral patterns of people, e.g. patients and employees. c) Participations: signifies a performance, e.g. performers, authors, witnesses. d) Acts: represent an inten tioned activity in the HL7 business domain, observations, supplies. e) Act relationships: associate two acts. f) Role links: associate an entry and a role [Stevens, 2003, p.10] [Blobel et al., 2006, p.345].

Clinical Document Architecture

The Clinical Document Architecture (CDA) is a document markup standard for clinical document exchange. It is part of the HL7 version 3.0. Its current release, release 2, is an ANSI approved standard. 2005 [Dolin et al., 2006, p.30]. CDA specifies the structure and semantics of clinical documents [Dolin et al., 2006, p.31]. A CDS instance is seen as a single information object, which can contain text, images, sound and other multimedia contents. CDS documents are encoded in XML, thus they are fully machine readable. CDA is based on RIM, thus the semantics of CDA elements are defined in the RIM specifications. CDA is richly expressive and flexible to cover the clinical domain [Dolin et al., 2006, p.31]. A CDA document can be transferred within a message but can also exist outside a message.

A CDA document consists of a header and a body. The header sets the context as a
The Importance of Standardization

whole and enables the exchange and management of clinical documents. Based on the header information, CDA instances of a patient can be compiled into a lifetime electronic patient record [Dolin et al., 2006, p.34]. The body of a CDA document contains the clinical report. The body can either be an unstructured blob, i.e. a non-XML content or can be composed of structured markup [Dolin et al., 2006, p.35].

CDA documents are not HL7 messages. Messages tend to be transient, trigger-based, nonpersistent. CDA instances are persistent, are seen as a whole and have clinical authentication.

3.3 Terminological Compatibility with the Unified Medical Language System

The previous section discussed integration at type level and introduced a widely used healthcare specific standard to allow different, originally not compatible, information systems in the healthcare domain to interoperate. This section focuses on interoperability at instance level, i.e. the terminological compatibility. Beyer et al. [2004, 269] argue that, by using a controlled terminology, standard catalog or classifications, heterogeneity at instance level can be avoided. A large number of medical ontologies exist [U.S. National Library of Medicine, 2007c]. To present a concrete solution to terminological compatibility, this paper will discuss the Unified Medical Language System (UMLS). UMLS is a complete framework that provided an integrated access to a very large set of biomedical and health related vocabularies. Additionally, it offers the software tools to build or enhance healthcare information systems “that create, process, retrieve, integrate, and/or aggregate biomedical and health data and information, as well as in informatics research” [U.S. National Library of Medicine, 2006]. The UMLS framework provides the means to develop applications “that behave as if they understand the meaning of the language of biomedicine and health” [U.S. National Library of Medicine, 2006]. UMLS is not restricted for a certain application; it is customizable and can be applied to a wide range of applications that interact with medical data in a variety of medical domains, e.g. patient care, indexing, cataloging biomedical literature, health service billing [U.S. National Library of Medicine, 2007a]. The UMLS Knowledge Sources can be customized for a specific task with tools provided as part of the framework [U.S. National Library of Medicine, 2006]. UMLS consists of four elements: The first three are the Metathesaurus, the Semantic Network and the Specialist Lexicon. They are called UMLS Knowledge Sources. The fourth element consists of additional tools to access and customize UMLS. The rest of this section gives an overview of each element.

3.3.1 Metathesaurus

The Metathesaurus forms the base of UMLS. It is a very large vocabulary database that contains biomedical and other health related concepts, the relationships between the con-
The Importance of Standardization

3. Concepts and their various names [U.S. National Library of Medicine, 2007a]. UMLS is not a single taxonomy or classification but consists of a large number of thesauri, codesets, classification and lists of controlled terms from the medical domain. In UMLS these are referred to as source vocabularies of the Metathesaurus [U.S. National Library of Medicine, 2007a]. Examples of source vocabularies are ICD-10, SNOMED CT, MeSH and Gene Ontology [U.S. National Library of Medicine, 2007c]. UMLS is not a comprehensive ontology of biomedicine authored by the National Library of Medicine nor does it represent a single, consistent view of the world (except for the Semantic Network). UMLS maintains the meanings, concept names and relationships and the view of the world from the source vocabularies. It does however store everything in a single, common format. UMLS organizes concepts by meaning: alternative names and views of the same concept are linked and useful relationships between concepts identified.

The Metathesaurus is linked to other knowledge sources. To provide constant categorization, each concept in the Metathesaurus is assigned to at least one semantic type in the semantic network. Additionally, many words or multi-words from the Metathesaurus also appear in the specialist lexicon [U.S. National Library of Medicine, 2007a]. The Metathesaurus is multi-purpose. To use it effectively, it must be customized for a specific use [U.S. National Library of Medicine, 2007a].

3.3.2 Semantic Network

The semantic network is part of UMLS. It is an upper-level ontology that provides an overarching conceptual framework to consistently categorize all concepts represented in the Metathesaurus [McCray, 2003, p.80]. The semantic network contains 135 subject catalogs, called semantic types, and 54 relationships that might hold between the semantic types. The major semantic type groupings include “organisms, anatomical structures, biologic function, chemicals, events, physical objects and concepts or ideas” [U.S. National Library of Medicine, 2007b]. An extract of the semantic network is shown in figure 3.1. A concept in the Metathesaurus is assigned to at least one semantic type. Always the most specific semantic type is chosen.

The most common relationship is is a. It links a semantic type to its parent to establish a hierarchy among semantic types.

The granularity differs across the hierarchy [U.S. National Library of Medicine, 2007b]. The other five major categories of relations are physically related to, spatially related to, temporary related to, functionally related to and conceptually related to [U.S. National Library of Medicine, 2007b].

3.3.3 Specialist Lexicon

The specialist lexicon provides the information needed for Special National Language Processing (NPL). The lexicon contains lexical entries for English vocabulary, biomedical terms,
terms from the UMLS Metathesaurus and the Dorland’s Illustrated Medical Dictionary. Each entry contains the syntactic, orthographical and morphological information that a specialist NPL system needs [U.S. National Library of Medicine, 2007d].

3.3.4 MetamorphoSys

MetamorphoSys is the UMLS installation wizard written in Java. As mentioned above, UMLS covers a broad spectrum. To install and customize UMLS, MetamorphoSys is used. It comes with every release of UMLS included and runs on Linux, Solaris, Mac OSX and Windows NT (and higher versions) [U.S. National Library of Medicine, 2007e].
4 Case Study: Mirth

In the precedent chapter it was argued that data integration is a necessary precondition to integrate healthcare processes that span over several heterogenous information systems. In this chapter, the authors demonstrate by means of a simple example with the open source HL7 integration engine Mirth how two heterogenous information systems can be integrated. Mirth is by far not the only HL7 integration engine. Other examples of tools that offer HL7 integration include Iguana by iINTERFACEWARE, HL7Connect Interface Engine and IBM Websphere. This chapter is structured as follows: section 4.1 describes a simple use case that will be implemented in section 4.3. Section 4.2 gives an overview of Mirth.

4.1 Use Case

This use case demonstrates the following: 1. A patient is admitted to hospital with asthma. The receptionists enter the data into the healthcare information system. 2. The healthcare information system triggers an “Admit a patient” event and sends it to Mirth. 3. Mirth checks if the message is of the correct type and if this is the case, accepts it. 4. The legacy system, which is used to store patient data, uses ICD-10 to classify diseases. The receptionist entered the name of the syndrome the patient declared, not the ICD-10 code. Mirth accesses a Java program that looks up the ICD-10 code in the UMLS Metathesaurus and replaces the value entered by the receptionist with the correct code. 5. Mirth selects a set of fields from the message and stores them in the database of the legacy system.

4.2 Mirth

This section gives a highlevel overview of Mirth. Mirth is an open source HL7 integration engine that handles the routing, filtering and transformation of messages [WebReach, Inc., 2007]. Mirth is created, maintained and supported (also commercially) by webREACH. It is written in Java and thus runs on several operating systems. In addition to sending and receiving HL7 messages, Mirth has connectors for a variety of protocols such as TCP/MLLP, SMTP, JMS Database (e.g. MySQL, Oracle), File, SOAP and FTP/SFTP [Lang and Bortis, 2007]. Its supported message standards are HL7 version 2.x and 3.0, X12, EDI and plain XML thus Mirth can be used to integrate heterogenous information systems in the healthcare domain that due to, for example different protocols, incompatible data models, could otherwise not temperate without having each system to develop its own proprietary integration solution [Lang and Bortis, 2007]. Mirth consists of two parts: the administrator to configure the server, create interfaces and monitor the server and the server itself. The server manages the interfaces; in Mirth interfaces are called channels. Sweeney [2006]
describes channels as “tasks that I want to do with HL7 messaging”. The architecture of a channel is depicted in Figure 4.1. A channel consists of four elements: the source connector, filters and validatory, transformers and destination connectors. Connectors receive messages for a channel by either opening a port and listening for incoming messages or connecting to a system such as a database and run queries. When the message is inside the channel, it is passed to the filter and validation step. The message is transformed into XML and checked against a series of rules to determine its validity. If the message is valid, it is passed to the transformer. The transformer will execute a series of steps to transform the message (e.g. to transform it into a new datatype, replace values, etc.) or performing mapping, in which parts of the incoming messages are mapped to variables that can be used by the destination connector. The transformed message is passed to the destination connectors that connect to an external system, such as a database or another HL7 compatible information system. The filter and transformer rules are written in JavaScript. A very useful feature is that any Java class in the CLASSPATH can also be directly accessed with the Package expression, e.g. Packages.remote.GetICD will call the GetICD class.

Mirth supports different integration pattern, called channel modes:

**Integration**: the message is processed and sent to another system.

**Broadcast**: the message is broadcasted to several systems over the same connector type.

**Router**: the message is sent to several systems and uses different connector types.

**Chaining**: instead of sending the message to another system, it is sent to another channel.
4.3 Implementation

The use case has been described in section 4.1. This section discusses how it was implemented with Mirth. The channel is called *Patient Admission*. The HL7 message shown below is sent by the program HL7browser, that for this proof of concept represents a healthcare HL7 capable information system. The HL7 message is:

```
MSH|^~&|ADT-HIS||HL7INSPECTOR||20060101100000||ADT^A01|1|P|2.3
EVN|A01|20060101||PID|||4711|John Doe||19701024|M|||Mayday Str. 505 1700 Fribourg CH|||GS|EV||DG1||asthma|breaths with difficulty|ADMITTING|
```

The source connector uses a LLP Listener that binds on port 6661 and accepts HL7 version 2.x messages. Next, the source filter, verifies if the event type code is A01 (Admit a Patient). If this is the case, the message is accepted (filter returns `true`, otherwise, the filter return `false` and the message is rejected and the sender receives an error). Figure 4.2 shows part of the environment in Mirth to write transformers. The center top lists the steps executed by the transformer. The first step is written in JavaScript. It accesses a Java class that returns the ICD-10 code for a given disease and logs this step. The accepted message is passed to the transformer. This prototype uses a hash which maps five disease names to ICD-10 codes. It does not interact with UMLS. The right part of the menu contains a list of prewritten functions available and the tree of the HL7 message (not visible in Figure 4.2). The variables created in the transformer are used by the destination connector. It connects to a PostgreSQL database that represents the database of the legacy PMS and updates a table with the query:

```
INSERT INTO "pmsXP".patient (name, icd, address, diag_desc) VALUES (${name},${icd},${address},${diagnosis});
```

Figure 4.3 shows that the ICD-10 code was successfully inserted into the table.
5 Integration through Service-Orientation

As previously seen, the integration of information is the key towards a seamless digital and automated exchange of information among the interacting stakeholders in a healthcare system to support efficient and effective medical processes. As we elaborated on the aspect of the what, i.e. the format of the information exchanged, in chapter 3, this chapter shall briefly introduce current generic models and architectures as the aspect of how this information can actually be shared. The question of how stakeholders can be integrated on the technological and organizational level using the paradigm of Service Oriented Architecture (SOA) is partially answered and by giving an overview on the topic.

Sahay [2006, p.2] states that in general, “entities (people and organizations) create capabilities to solve or support a solution for the problems they face in the course of their business”. Regarding healthcare the entities cited relate to the stakeholders in the healthcare system and the problems they face can, among other things, be interpreted as the need to exchange patient-related information with other stakeholders in order to execute healthcare processes. It becomes obvious that in this scenario, one stakeholders’ “need for information needs to be met by capabilities offered by someone else; or, in the world of distributed computing, one computer agents’ requirements being met by a computer belonging to a different owner [i.e. stakeholder]” [Sahay, 2006, p.2]. The challenge of the requirements implicated is to build software systems that can bring the capabilities of one stakeholder together with the need of another stakeholder, i.e. that integrate the underlying information systems in the required manner.

Traditionally, the challenge of information exchange and thus integration of information systems was met through distinct point-to-point interfaces. This integration strategy led to complex IT landscapes because of the variety of different technologies and communication protocols in place. Sahay [2006, p.5] shows that the number of interfaces used drastically increases when proprietary formats are used and calculates that “ […] the total number of interfaces to be developed [and maintained] is $n \times (n - 1)/2$ for n applications”. This formula demonstrates that in order to leverage existing information systems - so called legacy applications - through integration and drive the development of e-health capabilities, implementing point-to-point interfaces will lead to an even more complex and hardly to maintain and overview landscape of communication protocols and integration technologies. This circumstance will make it hard to achieve the primary and high-level goals that e-health promotes in the future and demands for a national architecture of a standardized integration platform for healthcare-related information systems.

A possible approach to design and establish a platform that meets the expectations is the mindset that exists around the Service Oriented Architecture (SOA). In the paradigm of SOA, the goal is to avoid tight coupling of information systems by introducing platform-
independent services as the linchpin of the IT architecture: Carter [2007, p.49] defines a
service as a “repeatable business task supported by software deployed on network-accessible
platforms provided by the service provider [i.e. a stakeholder in the healthcare system]”. Carter
[2007, p.49] further describes a service as a “[...] self-contained, reusable software that is
independent of applications and the computer platforms on which it runs. Services have
well-defined [i.e. standardized] interfaces and allow for a 1:1 mapping between business task
[e.g. admitting a patient] and the exact IT components [e.g. patient information system]
needed to execute the task”. It is important to note, that Service Oriented Architecture is a
concept to build interoperable, loose coupled information systems to achieve a flexible and
dynamic IT landscape. From the point of view of this thesis, the main goal of the application
of SOA principles and methods is to build interoperable, i.e. integrated information
systems. According to Carter [2007, p.53] the matching entry point to SOA in healthcare
context would be to improve connectivity of existing and functionally related information
systems. Carter [2007, p.53] states the following five possible entry points to SOA:

People Achieve the goal of “improved productivity by putting the user experience within
the context of the business process” [Carter, 2007, p.53]. This can be achieved by
delivering “role-based interaction and collaboration through services” [Carter, 2007,
p.53].

Process Enable business processes to be modified faster, i.e. improving the ability for
process innovation. This is achieved by decomposing processes and “treating tasks as
modular services” [Carter, 2007, p.53].

Information Achieve better “business operations, more informed decisions and reduced risk
[...] by treating it [information] as a service” [Carter, 2007, p.53].

Reuse By enabling existing assets, i.e. legacy HIS, for services (providing and consuming),
a stakeholder can “lower risk and [achieve] faster time to market by leveraging proven,
time-tested functionality” [Carter, 2007, p.53].

Connectivity Reduce “maintenance costs and [achieve] greater reliability and consistency
through flexible, any-to-any linkages” [Carter, 2007, p.53]. This is achieved by connecting “systems, users, and business channels [stakeholders in the healthcare system]
using open standards” [Carter, 2007, p.53].

Figure 5.1 illustrates the previously cited SOA entry points by giving an overview on the
basic idea of service orientation. The building blocks and interaction mechanisms among
the elements shown in figure 5.1 are described in the following section.
5 Integration through Service-Orientation

5.1 Building Blocks of a Service Oriented Architecture

To successfully implement an architecture that rather focuses on the promotion of reusable services than on the existence of a heap of heterogeneous interfaces, several building blocks are necessary. The key building blocks and principles are shortly described in the following section. They include legacy systems, i.e. existing IT assets that are to be integrated, web services as the best-practice for machine-to-machine interaction over a network [Carter, 2007, p.76] and finally the enterprise service bus as the basic and underlying connectivity and integration infrastructure. The variation among software vendors, practitioners and consulting companies concerning the naming of SOA infrastructure elements is rather big as it is mostly part of product names. In this thesis the naming of Carter [2007] (IBM) shall be used.

5.1.1 Legacy Systems

A legacy system can simply be defined as “a system that is deployed in an organization” ora. Legacy systems often enable key business processes and are rather big IT systems that were stable in use for a long time. In the healthcare domain, hospital information systems (HIS) and electronic patient management systems (PMS) can be referred to as legacy systems. They tend to be heterogeneous regarding the technology used for implementation and are often not documented in an appropriate way. This leads to the fact that the integration of several legacy systems to provide a new business functionality on time and in a flexible way tends to be a very challenging task. This lack of integration capabilities among existing IT assets is the main driver towards a service-orientation of the IT architecture. The concept of SOA is to enable legacy systems to provide the functionalities offered as services based on open standards. By introducing this abstracting service layer, the complexity of the system providing the service can completely be hidden from, i.e. is a black box for the service
consumer. By loose connecting the services provided instead of the legacy system's code itself, a higher flexibility can be achieved on both sides by information hiding (application as a black-box): The service consumer doesn't have to react on changes due to maintenance as long as the service exposed isn't altered. And the application can be modified and the internal design can be improved without having to care for the service consumer. Furthermore, new business processes can flexibly be composed using existing services without the need of adapting the information system providing the basic process steps.

If a legacy system is not able to provide its functionalities as standardized services, an adapter is used that calls the API of the legacy system and exposes the service to the exterior. From an operational system point of view, SOA can be described as the attempt to harmonize legacy system interfaces to the exterior by introducing services as the one and only and thus standardized technology to call functionalities of an IT system.

### 5.1.2 Services

As previously explained, services are the way functionalities of information systems are exposed in a Service Oriented Architecture. The goal of the introduction of services is to build a standardized facade and thus well defined interface where service consumers, e.g. other IT systems or portals, can consume functionalities. Today, the industry's best practice method and de facto standard to realize services, is to introduce web services technologies. A web service is mainly based on open standards such as the Extensible Markup Language (XML) can be described as a messaging standard, that enables machine-to-machine conversation and remote procedure calls [Carter, 2007, p.76]. As this would be beyond the scope of this thesis, details on the technical implementation of web services are not further discussed.

### 5.1.3 Enterprise Service Bus (ESB)

The enterprise service bus is a connectivity infrastructure - often referred to as middleware - that provides several basics functionalities to enable SOA [Carter, 2007, p.85]. The following list presents the five key functionalities of an ESB according to Carter [2007, p.68]:

- Routing of messages between services.
- Converting of transport protocols between requester, i.e. consumer and service.
- Transforming of message formats between requester, i.e. consumer and service.
- Handling of business events from disparate sources.
- Ensuring of quality of service (security, reliability and transacted interactions).

Furthermore, an Enterprise Service Bus must provide a service registry facility, where available services are technically documented for the service consumers. This service repository acts as a broker between service providers and service consumers by providing service
lookup capabilities. Additional services such as the monitoring of the services and infrastructure performance and service orchestration to implement business processes can be part of ESB productlines. As this thesis focuses on information system integration, the topic of process orchestration is not further discussed.

5.2 SOA and HL7

The Service Oriented Architecture can be one possible solution to meet the integration challenges in the healthcare domain. The aim of this section is to bring the concept of HL7 together with the idea of service orientation. After shortly citing two different strategies of implementing a SOA in the healthcare domain using HL7, a possible architecture using Mirth is presented.

**HL7 used as a Content Type** In this model, a general SOA framework is implemented using common infrastructure, tools and approaches that are not healthcare-specific [Sahay, 2006, p.14]. This means that the SOA approach is implemented using common web service frameworks and XML. HL7 is simply used to define the content delivered through standardized web services. The consequence of the fact that HL7 itself defines several transmission and transport-related wrapper layers around the actual data leads, according to Sahay [2006, p.14], to a certain amount of overlap in functionality between HL7 and the web services used.

**Messaging Architecture based on HL7** The second possible approach for combining SOA principles with Health Level 7 is to build a SOA based on the messaging architecture defined by HL7 using different messaging and transports, including web services [Sahay, 2006, p.14]. The architecture is based on the messaging solution provided by HL7 and “simply uses web service technology at the transmission or transport level” Sahay [2006, p.14]. The drawback of this approach, however is that the entire architecture strongly depends on the development of HL7 and that the advanced benefits proposed by the concept of SOA, e.g. business process orchestration and advanced collaboration possibilities, may not be applied to further developments of the architecture.

5.2.1 Combining SOA Principles and Mirth

The previously introduced strategies of applying the concepts of service orientation to the healthcare domain using HL7 represent two fundamentally different approaches. Decision making institutions face the challenge of making a trade-off regarding the extent to which the concept and layers of HL7 shall be implemented for electronic communication among stakeholders in a healthcare system. Applying the entire stack of HL7 layers results in an integration platform, and thus e-health environment, where messaging and transport is arranged for medical information that can be transmitted by HL7. The drawback of this
approach is that, among the proposed building blocks of SOA, only SOAP and XML are used. This results in an environment that is neither open for different content types, i.e. those not supported by HL7, nor is the environment built on real services in the way it is proposed by SOA. Another drawback of the use of HL7 for transmission and not just as a content type is that this implies that the technological framework may be subject to the constant development in the medical standards domain [Sahay, 2006, p.14]. The requirement of building national healthcare infrastructures that rely on widely accepted and stable standards regarding the technological infrastructure implies that in order to build a future-oriented and extensible infrastructure that is based on real services transmitting medical information, the use of HL7 only as a standard for content definition is proposed. A draft of a service implementation using Mirth as HL7 generator and not as an asynchronous messaging middleware is given in figure 5.2. The architecture that is sketched uses common web service technologies for transmission as well as HL7 as the standard of the medical data being exchanged between the stakeholders in a system. For the construction of HL7 messages out of data being pulled from legacy systems, e.g. a patient records database, Mirth’s transformation capabilities shall be used. The concept of channels is used for the service interface to route messages sent to and received from Mirth for HL7 generation: a channel is not integrating two different systems, i.e a source and a destination, but offers HL7 transformation capabilities to the software unit offering the web service exposed, i.e. the service interface which is sending and receiving HL7-codified content. The integration is realized, as proposed by SOA, at the level of the services. The application adapter (sometimes referred to as SOA adapter) is needed, if a given legacy application, i.e the technology it is built on, cannot directly implement web services. As Mirth is capable of accepting web services calls, another possible solution would be to offer Mirth’s transformation capabilities as a service and directly publishing them in the service registry of the ESB.

Figure 5.2: Snapshot of deploying Mirth in a Service Oriented Architecture as indirect Service Provider.
6 Conclusion

Data standardization and technical integration strategies are of high importance for successful e-health initiatives. In order to being able to electronically share patient-related information among stakeholders in the healthcare system, decision making institutions need to agree upon common data standards and integration technologies. In this context, the definition of a national e-health infrastructure architecture including data objects is the basis necessary to implement efficient inter- and intraorganizational information management. This thesis introduced Health Level 7 as a medical data standard, Mirth as a powerful HL7 creation and transformation engine and Service Oriented Architecture as an architectural approach towards a standards-based and federated e-health connectivity platform based on services.

The authors would like to state that a combination of the above mentioned standards and technologies could be used as the building blocks of a national healthcare system. HL7 is widely adopted internationally and so far the only comprehensive medical data standard being developed. As the ongoing development is absolutely crucial to HL7's future success, early adopting institutions or organizations face the challenge of the constant need of adapting newer versions of the very comprehensive and though complex HL7 standard. Mirth's HL7 creation and transformation capabilities are offered on an opensource basis. The advantages of this opensource attempt are extensibility and cost-efficiency: HL7-handling can be implemented fast and effective because no additional licenses need to be purchased. By publishing information as services with HL7 content, information systems can be loosely integrated. The integration happens at the service level and is no more hard-coded in the legacy applications. As the authors learned from the development team just shortly before finishing this thesis, Mirth is capable of routing messages straight back to the source: this means that web services can directly be implemented using Mirth: Mirth accepts a SOAP call, queries the database (legacy application) and routes the HL7 back to the source, i.e. service consumer. This even simplifies the use of Mirth as service provider in a SOA without the need of an additional web service interface. Figure 6.1 illustrates the use of Mirth as the web service provider.

In this architecture, the address of the information service would therefore be the address of Mirth together with the port of the specific channel. The drawback of this architecture is that the granularity of the services depends on the functionalities Mirth offers. In a practical implementation a trade-off needs to be made. The service can be implemented using a web service interface together with Mirth's transformation capabilities depicted in figure 5.2. In this implementation the web service interface can be programmed and more functionalities (e.g merging of information) can be implemented. This increases the flexibility and possibilities regarding the services exposes. The drawback of this solution
6 Conclusion

Figure 6.1: Snapshot of deploying Mirth in a Service Oriented Architecture as direct Service Provider.

however is that an additional layer needs to be introduced and the complexity increases.

Practical solutions that enable legacy systems to deal with HL7 messages will have to make this trade-off regarding the placement of the transformation engine. Because of the heterogeneous information system landscape in the healthcare domain, it is likely that it will not be possible to apply the exactly same architecture to all the stakeholders integrated. However, this is not a problem or a risk, because the services exposed to external institutions will not be affected: The technologies around SOA guarantee that the services exposed are implemented using agreed-upon standards. In this environment it does not make a difference how a service is implemented behind the scenes as long as the interface and the functionalities offered remain the same. It is therefore possible that both implementation styles - and others of course - can be applied in a common infrastructure and that the best suited technology can be chosen for each business case to implement the service: this is the promise of service orientation.

6.1 Outlook

The research showed that while technological standards, communication infrastructures and software seem to be ready for e-health, there exists various political issues that need to be solved prior to e-health being able to electronically support comprehensive medical processes. One of these issues is the unique identification of individuals and organizations in the system. Distributed information storage has led to the situation that attributes (e.g. treatment-related data) of the same entity (e.g. a patient) is stored in different information systems using a variety of different identification mechanisms. In order to implement information services for patient-related data, a unique identification number is required and needs to be adopted by all the stakeholders involved.
List of Figures

2.1 Stakeholders in a Healthcare System on the basis of Haas [2006, p.245] 4


4.1 Mirth channel architecture [Lang and Bortis, 2007] 15

4.2 Mirth transformer step environment 16

4.3 Legacy PMS with the values from the incoming HL7 message 16

5.1 SOA Entry Points [Carter, 2007, p.266] 18

5.2 Snapshot of deploying Mirth in a Service Oriented Architecture as indirect Service Provider 22

6.1 Snapshot of deploying Mirth in a Service Oriented Architecture as direct Service Provider 23
Bibliography


John Quinn Accenture. HI7 messaging standard version 2.5.1 an application protocol for electronic data exchange in healthcare environments. chapter 1, 2007.


Bibliography


Jon Metz. The hl7 evolution comparing hl7 version 2 to version 3, including a history of version 2, 2007.


