HL7 Services-Aware Interoperability Framework (SAIF)

April 8, 2010

Health Level 7
# Table of Contents

1. Services-Aware Interoperability Framework ...................................................................................................... 1

1.1. Preface .......................................................................................................................................................... 1

1.1.1. Roadmap to Reading about SAIF ............................................................................................................ 1

1.2. SAIF Executive Overview .......................................................................................................................... 2

1.2.1. SAIF: An Interoperability Framework for Enterprise Architectures ......................................................... 3

1.2.2. SAIF Uses Services-Aware Principles ....................................................................................................... 3

1.2.3. SAIF Core Components and Framework ..................................................................................................... 4

1.2.4. SAIF Specifications and RM-ODP .............................................................................................................. 4

1.2.5. What is SAIF ........................................................................................................................................ 5

1.3. SAIF Introduction ......................................................................................................................................... 5

1.3.1. SAIF Overview .................................................................................................................................... 5

1.3.1.1. Roadmap to Reading about SAIF ................................................................................................. 1

1.3.1.2. The SAIF Education Series ............................................................................................................. 6

1.3.1.3. SAIF Complex System Map ............................................................................................................ 9

1.3.1.4. SAIF Introduction System Map ......................................................................................................... 10

1.3.2. Value Proposition: Working Interoperability ............................................................................................ 11

1.3.2.1. Working Interoperability: Computable Semantic Interoperability ...................................................... 12

1.3.2.2. Working Interoperability between Two Trading Partners ....................................................................... 13

1.3.2.3. Working Interoperability: Stairway to Heaven ....................................................................................... 14

1.3.3. SAIF Overall Framework ......................................................................................................................... 15

1.3.3.1. SAIF Components ............................................................................................................................... 16

1.3.3.2. What is Architecture? ....................................................................................................................... 16

1.3.3.2.1. A Complex System ................................................................................................................... 17

1.3.3.2.2. Two Architectures of HL7 ......................................................................................................... 17

1.3.3.2.3. Enterprise Architecture Definitions .......................................................................................... 18

1.3.3.3. HL7 Foundation Components ............................................................................................................ 18

1.3.3.4. Services-Aware Architecture ............................................................................................................. 18

1.3.3.4.1. Services-Oriented Architecture Definitions .................................................................................. 22

1.3.3.4.2. Services-Awareness and Working Interoperability ......................................................................... 20

1.3.4. SAIF Core Principles ............................................................................................................................... 20

1.3.4.1. Objectives of the SAIF Project ......................................................................................................... 20

1.3.4.2. Lens of SAIF .................................................................................................................................. 20

1.3.4.2.1. Lens of SAIF Graphic .................................................................................................................. 21

1.3.4.2.2. Services-Oriented Architecture .................................................................................................. 22

1.3.4.2.3. Model-Driven Architecture ......................................................................................................... 23

1.3.4.2.4. Computable Semantic Interoperability ......................................................................................... 23

1.3.4.2.4.1. Bridging Domain-Specific Models .......................................................................................... 24

1.3.4.2.5. Reference Model for Open Distributed Processing ........................................................................... 25

1.3.4.2.5.1. Five RM-ODP Viewpoints ......................................................................................................... 25

1.3.4.2.5.2. RM-ODP Multidimensional Specification Framework .............................................................. 26

1.3.4.2.5.3. Integration Points .................................................................................................................... 27

1.3.4.2.6. Integrating SOA MDA CSI and RM-ODP ..................................................................................... 27

1.3.4.2.7. HL7 Artifacts ................................................................................................................................ 28

1.3.4.2.7.1. HL7 Artifacts Inventory and SAIF Survey .................................................................................... 29

1.3.4.2.7.2. Non-Payload Message Content ................................................................................................. 30

1.3.4.2.7.3. Other HL7 Content ................................................................................................................... 30

1.3.4.2.7.4. Legacy Artifacts, Interoperability Paradigms, and the Unified Field Theory ................................. 31

1.3.4.3. Working Interoperability Requirements .............................................................................................. 31
1. Services-Aware Interoperability Framework

Services-Aware Interoperability Framework (SAIF) is an architectural framework for making healthcare systems interoperable. The various topics in the SAIF document describe the architecture for designing and achieving Working Interoperability at all levels.

SAIF was originally created as the Services-Aware Enterprise Architecture Framework (SAEAF); however, as the product evolved, it became clear that this name suggested it was a complete framework for producing enterprise architectures. It is not HL7's intent to attempt to replicate or substitute well-established Enterprise Architecture frameworks and methodologies, such as The Open Group Architecture Framework (TOGAF) or Zachman, with this work. Rather, the intent is to provide a framework and methodology to allow HL7 to produce, and for its customers to use standardized specifications for achieving Working Interoperability (WI) that have sufficient content to be readily understood, successfully implemented, and manageably evolved.

This introductory document provides a roadmap to understanding the following:

- The utility of the SAIF approach,
- A background for the application of SAIF by HL7 in developing WI specifications, and
- Information to install confidence in those who are considering, or currently vested in, HL7 standards as a foundation for the inter- and intra-organizational interoperability of their healthcare businesses, their software applications, and their healthcare professionals and service providers.

1.1. Preface

The audience for the Services-Aware Interoperability Framework (SAIF) document includes enterprise architects, business analysts, members of Standards Developing Organizations, executives, Health Level 7 work group co-chairs, and implementers.

Executives who are sponsoring interoperability initiatives need to know how SAIF will help their organizations achieve interoperability, and why SAIF is the best approach to achieving interoperability.

Architects and business analysts who are designing interoperability architecture for their organizations can use SAIF to achieve their interoperability goals.

HL7 working group co-chairs and teams contributing to the maintenance and evolution of these standards can employ SAIF approaches to contextualize their work, achieve better coherence of work products across the organization, and be confident that their efforts will result in benefits for their home organizations and HL7's customers.

1.1.1. Roadmap to Reading about SAIF

Use this roadmap to determine which sections of the SAIF document to read.

<table>
<thead>
<tr>
<th>Audience</th>
<th>What to Read</th>
</tr>
</thead>
<tbody>
<tr>
<td>If you are an executive</td>
<td>Read the Executive Overview and the beginning of each major section in the SAIF book.</td>
</tr>
<tr>
<td>If you are an enterprise architect or business analyst</td>
<td>If you are in a hurry, start with the Enterprise Conformance and Compliance Framework (ECCF) section, then read the Governance Framework (GF) section, and finally the Behavioral Framework (BF) section.</td>
</tr>
</tbody>
</table>
1.2. SAIF Executive Overview

The HL7 Services-Aware Interoperability Framework (HL7 SAIF) is the responsibility of the HL7 Architecture review Board (ArB), and is the result of a collaborative development and refinement exercise of the HL7 standards development community. The goal of SAIF is to promote the safe and effective delivery of health services to clients and patients of healthcare delivery organizations by establishing Working Interoperability within and between healthcare organizations, healthcare business processes, and health service providers.

Medical and healthcare organizations using different technologies, software, and business structures need a common framework for sharing documents, messages, and services with other organizations. An example of a document might be an invoice or a doctor's report. An example of a service might be a request for a lab order or a patient's longitudinal health record.

Often these organizations are unable to share the information because the different systems do not interoperate with each other, which increases the time and costs for providing health services, and potentially affecting client/patient outcomes adversely.

For a long time, HL7 has been developing specifications, methodologies, and extensible standards for making healthcare systems interoperable. HL7 collaborates with healthcare information users and other Standards Development Organizations, and enables domain experts from the healthcare industry to develop healthcare information standards in their areas of expertise.

SAIF seeks to meet this need by providing a framework for ensuring interoperability for documents, messages, and services between organizations. SAIF is designed not only to craft software components -- messages, computable documents, services, and applications -- that are usable in the healthcare, life sciences, and clinical research domains, but also to facilitate and enable efficient lines of communications between the various cross-functional teams.

The following figure shows an example of two healthcare enterprises, A and B, which incorporate SAIF Working Interoperability specifications into their Enterprise Architecture Framework. These enterprises have interoperability with other healthcare systems (represented by the small purple rectangles).

Figure 1. SAIF-based Working Interoperability specifications are intended to be incorporated into the architectures of healthcare organizations that need interoperability.
1.2.1. SAIF: An Interoperability Framework for Enterprise Architectures

In January 2008, the HL7 CTO and the Technical Steering Committee (TSC) asked the Architecture Review Board (ArB) to develop an Enterprise Architecture Specification (EAS) for HL7. The EAS needed to satisfy the following criteria:

- Apply to each of HL7's three interoperability paradigms (IPs): messages, documents, and services.
- Address the growing disparity between the various solution sets emerging from HL7.
- Use existing HL7 V3 and Reference Information Model (RIM) artifacts and expertise to the maximum degree possible.
- Provide support for measurable conformance and compliance.
- Define appropriate governance structures and processes.
- Provide support for directly implementable solutions.

SAIF is the ArB's response to these requirements.

After considering the assignment, the ArB determined that they could not specify a "top-down" design for the HL7 Enterprise Architecture Specification (EAS). Rather, they unanimously concluded that they should instead focus on developing an Enterprise Architecture Framework in which an EAS would emerge through the work of HL7's committees and project groups. In addition, the framework being developed would focus on aspects of intra- and inter-architecture definition, design, development, and deployment that affected Working Interoperability, a focus that was reflected in the descriptively informative name of the framework.

SAIF enables the explicit representation of the myriad of factors that collectively allow Working Interoperability to be achieved between two or more HL7 stakeholder systems and organizations. SAIF focuses on generating artifacts that are both informative and practical to stakeholders at various points in the chain from domain experts to implementers. In particular, SAIF enables concrete traceability -- including verifiable milestones and accountability -- from requirements through implementation using several interoperability paradigms: messages, documents, and services.

1.2.2. SAIF Uses Services-Aware Principles
Why use the term "services-aware"? HL7 recognizes the potential benefits of Service-Oriented Architecture (SOA) principles in achieving viable interoperability, and has expanded its scope of interoperability technology to include services.

The defining core principles for SAIF are drawn from the widespread work in services and Service-Oriented Architectures (SOA). In this sense, services awareness provides frameworks and perspectives that are necessary for the rigorous and explicit representation of both the behavioral and static semantics between systems interacting in scenarios, where the goal is *Working Interoperability (WI)*.

In particular, "services-aware analysis" allows specification developers to define appropriately layered specifications that employ a specific interoperability paradigm in a way that maximizes traceability to the overarching business processes driving the interaction.

### 1.2.3. SAIF Core Components and Framework

The HL7 Services-Aware Interoperability Framework (HL7 SAIF) consists of four core components:

- Enterprise Conformance and Compliance Framework (ECCF)
- Governance Framework (GF)
- Behavioral Framework (BF)
- Information Framework (IF)

It is important to remember that SAIF is an interoperability framework rather than an Enterprise Architecture Specification (EAS).

Together, these sub-frameworks provide an overarching framework for specifying and governing the set of artifacts that enables HL7 to develop specifications for support Working Interoperability (WI) among parties using HL7 specifications alone, or with specifications and standards that other organizations have developed. The Enterprise Conformance and Compliance Framework (ECCF) extends the historical work within HL7 on conformance, while the Behavioral Framework (BF) formalizes and extends the constructs of the HL7 dynamic interaction models to include explicit semantics around contracts and integration. The Governance Framework (GF) provides the context to understand the value of service contract patterns for interoperability, along with an understanding of how to manage and sustain interoperability over time. The GF is an essential enabler for organizations to design and implement enterprise architectures that are both robust and flexible.

### 1.2.4. SAIF Specifications and RM-ODP

After a high-level review of available models, it was decided to adopt the Reference Model for Open Distributed Systems (RM-ODP) as the common language for discussing its emerging Enterprise Architecture Framework (EAF) specification.

In particular, the adoption of the RM-ODP constructs of *conformance statements*, *conformance assertions*, and *integration points* sorted via the five RM-ODP viewpoints – Business/Enterprise, Informational, Computational, Engineering, and Technology -- provided a suitably robust framework within which to develop the SAIF deliverables.

The various specifications, thus, become the building blocks of the EAF, which enable the emergence of the EAS and also the ultimate development of HL7 products. Finally, the specification of these products provide, as part of their development, input to the conformance and compliance processes. These specifications are readily adaptable to various forms of governance, both within HL7 and between HL7 and other organizations that provide additional standards for use.

The RM-ODP viewpoints also provide an ability for consumers of Working Interoperability specifications to determine which standards are appropriate for a particular purpose, and to adapt their systems to use them...
effectively and safely.

**Note** Although SAIF uses RM-ODP to provide some of the structural and organizational aspects of ECCF, similar constructs could have been drawn from other Enterprise Architecture Frameworks, such as Zachman and TOGAF. The critical difference between SAIF and these frameworks is that SAIF focuses on only those aspects of enterprise architecture that affect the achievement of Working Interoperability.

### 1.2.5. What is SAIF

In summary, SAIF emphasizes reuse of architectural artifacts developed from a common *framework grammar* as a critical success in obtaining Working Interoperability (WI) between two or more HL7 systems or organizations using HL7 specifications.

SAIF is intended to be used comprehensively across all of the HL7 work groups to create standards that either build on or further enhance WI between health care, clinical research, and life science systems, regardless of the interoperability paradigm that these systems use to exchange information and coordinate behaviors.

**Figure 2. SAIF produces guidance for the creation of a coherent Enterprise Architecture Specification (EAS) that realized projects and operational systems can use. You can use SAIF to implement interoperability in an Enterprise Architecture implementation.**

**Note** Either the original SAIF triptych figure or a simplified figure will go here.

### 1.3. SAIF Introduction

The SAIF Introduction and Overview describes the general constructs that frame the HL7 Services-Aware Interoperability Framework (SAIF). SAIF represents a synthesis of best practices and concepts from multiple architectural frameworks.

This introduction to SAIF goes into a lot of technical depth and assumes you are already familiar with architectural standards and how the HL7 organization produces standards specifications.

Other major sections of the SAIF document describe the details of the Enterprise Conformance and Compliance Framework (ECCF), Governance Framework (GF), and Behavioral Framework (BF). In a future release, the SAIF document also will include information about the Information Framework (IF), the Implementation Guide, and SAIF examples and lessons learned.

### 1.3.1. SAIF Overview

Health Level 7 (HL7) provides a framework and standards for the exchange, integration, sharing, and retrieval of electronic health information.

The HL7 Services-Aware Interoperability Framework (SAIF) provides consistency between all HL7 artifacts, and enables a standardized approach to Enterprise Architecture (EA) development and implementation, and a way to measure the consistency of implementations of Working Interoperability (WI).

SAIF is a way of thinking about producing specifications that explicitly describe the governance, conformance, compliance, and behavioral semantics that are needed to achieve computable semantic working interoperability. The
intended information transmission technology might use a messaging, document exchange, or services approach.

SAIF is an architecture for achieving interoperability, but it is not a whole-solution design for enterprise architecture development and management.

## Related information

- Section 1.3.2. Value Proposition: Working Interoperability
- Section 1.3.3.1. SAIF Components
- Section 1.3.3.4. Services-Aware Architecture
- Section 1.3.4. SAIF Core Principles

### 1.3.1.1. Prerequisite Knowledge and Outcomes

This topic covers what you need to know ahead of time and what you will learn after reading the SAIF Introduction and Overview.

**Prerequisite knowledge:**

- **Essential:** Familiarity with the problems of achieving Working Interoperability in the healthcare, life sciences, and clinical research domains.
- **Helpful:** General knowledge of HL7.

**Outcomes:**

- Understanding of the organizational principles and context around which SAIF was developed.
- General understanding of core components of SAIF.
- Preparation for the other SAIF educational materials.

## Related information

- Section 1.3.3. SAIF Overall Framework

### 1.3.1.2. The SAIF Education Series

The SAIF Education Series consists of a series of Word documents and companion PowerPoint decks. These documents define and discuss the core components and processes, which are associated with the HL7 Services-Aware Interoperability Framework (SAIF).

The overall organization of the document and deck pairs is based on the major components of SAIF. Each of the five topics has its own document and deck module.

- SAIF Introduction and Overview
- SAIF Behavior Framework (BF)
- SAIF Enterprise Conformance and Compliance Framework (ECCF)
- SAIF Governance Framework (GF)
- SAIF Application and Implementation (including the Alpha Rollout Project)

The following summarizes the content of each module:
SAIF Introduction and Overview

- SAIF value proposition: *Working Interoperability*
- SAIF structural components
  - Introduction to the Behavioral Framework
  - Introduction to the Conformance and Compliance Framework
  - Introduction to the Governance Framework
- SAIF core principles
- Implementing SAIF – impact of change
  - Education
  - Change management
  - Alpha projects
- SAIF history

SAIF Enterprise Conformance and Compliance Framework (ECCF)

- The ECCF specification stack and associated concepts
  - Conformance (statements vs. assertions)
  - Compliance
  - Consistency
  - Certification (conformance vs. compliance)
  - Traceability
  - Compatibility
  - Localization
  - Jurisdiction
  - Provenance
- Building specifications using the specification stack
- ECCF and Governance

SAIF Governance Framework (GF)

- What is Governance in the context of interoperability
- Why is Governance important to managing interoperability
HL7 Services-Aware Interoperability Framework (SAIF)

- SAIF provides an abstract Governance model
- What products are produced using the Governance Framework
- How does the Governance Framework relate to the Information Framework, the Behavioral Framework, and the Enterprise Conformance and Compliance Framework
- How are Governance products maintained
- Governance provides four contexts of use:
  - Within an Interoperability Community
  - Within a Standards Development Community
  - Across Standards Development Communities
  - Within an Enterprise Architecture

SAIF Behavioral Framework (BF)

- Behavioral Framework overview
- Behavioral Framework essentials
- Behavioral Framework foundational concepts and models
- Using the Behavioral Framework packages
- Behavioral Patterns
- Appendix A: The Behavioral Framework and the HL7 Legacy Dynamic Model
- Appendix B: RIM-Based Services and V3 (RIM-Based) Messages
- Appendix C: References

SAIF Application and Implementation

- Examples and lessons learned
- Service Interoperability Paradigm
- Message Interoperability Paradigm
- Document Interoperability Paradigm

The document modules have been structured so that the content can be read in a linear fashion. However, a considerable amount of cross-reference and dependencies exist between concepts in the BF, ECCF, GF, and IF. Therefore, readers wanting a deep or detailed understanding of the entire SAIF may find that reading the entire series in a linear fashion prepares for a second, non-serial approach with focus on particular topics of interest and their associated inter-document context.

The HL7 ArB welcomes all input on the content, organization, clarity, and relevance of the document modules. Given the commitment of HL7 to the path defined by SAIF, producing materials that aid everyone in the organization to get up to speed on SAIF as quickly, effectively, and completely as possible is clearly a high priority for the HL7 ArB and TSC.
Educational Materials Coming in the Future

Several new educational materials will be available in 2010:

- A PowerPoint presentation on the SAIF Information Framework (IF) will be available by the May 2010 Work Group Meeting (WGM).

SAIF Implementation Guide for HL7 Developers

An educational package will describe how to implement SAIF at all levels, and how to choose the artifacts based on your role (analyst, architect, or developer).

Related information

- HL7 Development Framework
- Section 1.3.5.1.1. SAIF Alpha Project Implementation

1.3.1.3. SAIF Complex System Map

In the SAIF complex system map, the SAIF components define a framework for achieving Working Interoperability for messages, documents, and services. These components include the Enterprise Conformance and Compliance Framework, Governance Framework, Behavioral Framework, and Information Framework.
Related information

- Section 1.3.6. SAIF Project Summary

1.3.1.4. SAIF Introduction System Map

The SAIF introduction system map shows how you can use SAIF to enable Working Interoperability between healthcare systems and organizations.

The key component of SAIF is the Enterprise Conformance and Compliance Framework (ECCF). ECCF provides a specification stack (SS) template, which you can use to collect testable conformance statements. Then you would
use conformance assertions to certify these conformance statements.

Related information

Section 1.3.6. SAIF Project Summary

1.3.2. Value Proposition: Working Interoperability
This section defines and explains the term Working Interoperability (WI), and includes two examples of users (also called trading partners) who are exchanging information to achieve business goals.

HL7’s mission, as stated on its web page, is as follows:

*HL7 provides standards for interoperability that improve care delivery, optimize work flow, reduce ambiguity, and enhance knowledge transfer among all of our stakeholders, including healthcare providers, government agencies, the vendor community, other Standards Developing Organizations (SDOs), and patients. In all of our processes, we exhibit timeliness, scientific rigor, and technical expertise without compromising transparency, accountability, practicality, or our willingness to put the needs of our stakeholders first.*

To help in understanding the focus of SAIF, the HL7 mission statement was simplified and re-expressed, as follows:

*To provide the necessary specifications to enable HL7 users — referred to here as trading partners -- to exchange data and information at a computable, semantically interoperable level in the context of a specific set of transactions (referred to as a collaboration) that involves HL7-enabled systems representing the trading partners.*

**Related information**

- Section 1.3.3.4.2. Services-Awareness and Working Interoperability
- Section 1.3.4.3. Working Interoperability Requirements
- Section 1.3.4.5. Integration and Working Interoperability

### 1.3.2.1. Working Interoperability: Computable Semantic Interoperability

You can achieve Working Interoperability if you know the Computable Semantic Interoperability (CSI) between the trading partners.

*Working Interoperability (WI)* is defined as follows:

The collection of structures, processes, and components that support Computable Semantic Interoperability (CSI) between two parties ("trading partners") who are interacting (for example, exchanging information, coordinating behavior) to achieve one or more business goals. **Interoperability**, in this context, is further defined to be the deterministic exchange of data or information in a manner that preserves shared meaning. **Shared meaning** means that the details are worked out in advance so that both the sender and receiver understand the meaning in common. Figure 1 explains the difference between syntactic interoperability and semantic interoperability.

*Figure 3. Definition of interoperability*
The emphasis on enabling CSI between two or more trading partners in collaboration implies the need for both a pre-transaction agreement, such as some form of a contract that clearly details the specifics of the expected results of the transaction. Examples of transactions include the data and information to be exchanged, the details of the expected interactions between the trading partners during the transactions, as well as the capability to perform a post-transaction assessment (which ideally can be done automatically) of the degree of success of the transaction in progress or immediately following completion.

WI must be measurable in quantitative terms using business-value-measurements such as “degree of difficulty (cost)/degree of success (value).” A framework, such as SAIF which facilitates and enables WI, must provide a Conformance and Compliance Framework (preferably layered to allow for varying levels of trading partner maturity), as well as providing a set of constructs that allow for the explicit expression of the static, functional, and behavioral semantics which collectively define the “WI transactions.” Finally, the specifications for enabling WI must be defined in such a manner to be usable, useful, durable, and implementable in a variety of deployment contexts in a repeatable and understandable manner.

1.3.2.2. Working Interoperability between Two Trading Partners

Working Interoperability (WI) makes no assumptions about the size, character, or identity of the parties involved in the interaction; for example, the parties can be nations, enterprises, systems, department, and individuals. Additionally, WI makes no assumptions about the “what, why, or how” details of a given interaction.

Consequently, the concept of WI does not specifically refer to or imply a particular form of interaction, for example, messages, documents, or services.

The following figure shows a graphical representation of Working Interoperability.

Figure 4. Working Interoperability -- A set of transactions or interactions that involve the exchange and coordination of messages, documents, or services.
As described and depicted above, a given WI interaction (transaction or set of transactions) involves two parties that are trading partners. A formal approach to supporting those trading partners to assess both the difficulty (cost) and benefits (value) of a given WI interaction must be able to deal with the realities of trading partners at different levels of maturity in terms of their ability to explicitly support, send, and receive various types of semantic information.

The assessment of both difficulty and value is determined by evaluating the degree of commonality and consistency that the two trading partners have about a set of shared specifications that might or might not be standardized beyond the scope of the parties and the WI session. (For a given WI session with a specified number of trading partners, the degree of difficulty involved in achieving WI increases with the number of parties, but decreases with the degree or level of standardization of the set of specifications relevant to the WI context.)

SAIF formalizes the notion of “layering” a framework for assessing the transaction through the definition of four “levels” at which a given trading partner may be at in a given WI session.

**Related information**

- Section 1.4.2. ECCF: A Template for Working Interoperability

**1.3.2.3. Working Interoperability: Stairway to Heaven**

The SAIF "Stairway to Heaven" graphic shows the steps to achieving Working Interoperability (WI) at all levels. The greater the distance on the Stairway to Heaven, the more difficult the transforms required to achieve WI.

This figure shows six prototype trading partners who have varying degrees of semantic (behavioral, functional, and static) maturity for a particular WI session. In this figure, trading partners C and D are the most mature. However, trading partners A and F are less mature.

*Figure 5. Trading partners on the "Stairway to Heaven" engaging in a Working Interoperability session. The lightning bolt represents Implementation.*
Figure legend:

- A – F represents the trading partners.
- Trading partner E is at the Platform-Specific Model (PSM or Physical) layer.
- Trading partner B is at the Platform-Independent Model (PIM or Logical) layer.
- Trading partner F is at the Computationally Independent Model (CIM or Conceptual) layer.

Two trading partners on any two levels could decide to interoperate if they both come to the table with the same “reference” notions, they could achieve WI in a tractable, predictable manner. For example, they could agree to use SAIF, and bind to the Reference Information Model (RIM) and to the ISO 21090 standard. The two trading partners at the same level at the top demonstrate the optimal WI context.

The easiest (most automated) and highest value (most semantic) exchange WI interaction can occur between trading partners C and D. Because D is more mature than A, an exchange of equal value but of considerably greater difficulty could occur between A and D, if they are willing to work together to provide and build the required transformations necessary to move A to D’s level of maturity for the given WI session. Other combinations of trading partners result in similar discussions of cost and value, depending on the level of maturity that each trading partner brings to the WI session.

A mature specification has explicit conformance statements for the entire specification stack, backed up by reference to industry standards. An immature specification does not specify all types of semantics and leaves implementers to infer intent -- often resulting in incompatible systems that do not interface as intended to meet a shared business objective.

If system implementers do not agree, you cannot have automated interoperability and you would need custom software to bridge the gap between systems.

Related information

► Section 1.4.2. ECCF: A Template for Working Interoperability

1.3.3. SAIF Overall Framework

SAIF is the framework for developing an Enterprise Architecture Specification (EAS).

Before discussing the collective set of overarching contexts, filters, or sources of guidance that define the overall structure and content of SAIF, the following SAIF facts are presented as a “framework-to-the-frameworks of SAIF.”
1.3.3.1. SAIF Components

As previously mentioned, SAIF defines four frameworks that collectively provide the context, processes, and specific deliverables for deriving an HL7 EAS by HL7 Work Groups and participating organizations.

A high-level summary of each framework is as follows:

- **Behavioral Framework (BF)**
  - Specification of integration semantics of IT components
  - Linkage of integration semantics to real-world behavior

- **Enterprise Conformance and Compliance Framework (ECCF)**
  - Layered to enable “degrees” of conformance and compliance

- **Governance Framework (GF)**
  - Internal HL7 governance
  - Relationships between HL7 and other organizations specifying standards
  - Relationships between standards and government regulations and policies
  - Governance within an interoperability community

- **Information Framework (IF)**
  - Static semantics
  - Existing information framework artifacts

**Note:** SAIF does not include a formal static framework because mature specifications, such as the HL7 Reference Information Model (RIM), Abstract Data Type (ADT) specification, Clinical Document Architecture (CDA), already describe static semantics in HL7 specifications.

1.3.3.2. What is Architecture?

Architecture is a set of resources, relationships, patterns, practices, and processes that collectively define a system and its products-of-value.

From an HL7 perspective, the following definition also is useful:

*The collection of parties (persons, organizations, or systems), roles, responsibilities, relationships, processes, and deliverables which collectively define the “what, who, and how” of the specifications which HL7 develops which, in turn, enable instances of CSI collaborations between HL7-enabled trading partners.*

Finally, from a software engineering perspective, Martin Fowler’s more pragmatic definition also is useful in gaining perspective:

“Architecture is the stuff that’s hard to change after it’s built.”

It is beyond the scope of this document to discuss the relationship between the management of complexity and an EAS (in general).

---

(1) The Information Framework document will be developed from existing HL7 materials contextualized for use within SAIF. It will describe the how existing HL7 artifacts such as the Reference Information Model (RIM), data types, Vocabulary Best Practices, Clinical Document Architecture, Clinical Statement Pattern, and HL7 Core Principles are applied in the context of SAIF.
1.3.3.2.1. A Complex System

An example of a complex system is the healthcare, life sciences, and clinical research domain.

In the following figure, Ivar Jacobson gives a definition of a complex system. (2) The Value Proposition of an Enterprise Architecture Specification (EAS) for HL7 is based on the ability of an EAS to manage the complexity of the overarching healthcare, life sciences, and clinical research “system” in such a way that enables the delivery of the multidimensional values -- efficiency, effectiveness, and efficacy – that result from CSI collaborations between HL7-enabled trading partners.

Figure 6. Ivar Jacobson defines a complex system as "a collection of resources that are organized into a multilevel vertical hierarchy whose products of value are the result of executing horizontal processes that cross vertical organizational boundaries."

1.3.3.2.2. Two Architectures of HL7

HL7 is concerned with both internal and external architectures.

1. Internal architecture: HL7 is dedicated to producing specifications that make it easier to accomplish Working Interoperability with external architectures.

2. HL7 enterprise architecture: The collective HL7 enterprise architecture enables Working Interoperability in the healthcare, life sciences, and clinical research areas.

The internal architecture is “how HL7 does its job” and the enterprise architecture is how the healthcare systems interoperate. Each organization that considers itself an enterprise has its own architecture. Interoperating organizations have a collective architecture. The graphic in shows architectures within architectures.

HL7 specifications produced for architecture #1 support interoperability in architecture #2 for the composite enterprise. The composite enterprise is made up of all the participating enterprises in the healthcare tapestry.

SAIF is a set of sub-frameworks for producing specifications that support aspects of HL7 architecture #2. SAIF is not meant to be a complete EAS. Instead, SAIF focuses on the critical interoperability aspects of HL7 specifications in each of the three HL7 Interoperability Paradigms (messages, documents, and services). SAIF also has structural, relationship, and process implications for architecture #1, which defines the artifacts and specification semantics needed to support interoperability in healthcare, life sciences, and clinical research.

1.3.3.2.3. Enterprise Architecture Definitions

The following are Enterprise Architecture definitions, which the SAIF document uses.

Enterprise Architecture Framework:

The set of rules, constructs, and specific aspects of an overall framework that define, govern, or otherwise

(2) This figure was adopted from textual discussions in “The Object Advantage,” by Ivar Jacobson.
inform and enable to define, design, and development Enterprise Architecture components (also called architecture primitives).

**Enterprise Architecture Specification:**

The set of architecture primitives, processes, patterns, and principles that collectively define how HL7 produces products for its external stakeholders.

Enterprise Architecture focuses on producing a number of architecture primitives (the result of engineering efforts) -- which may then be assembled into composites (the result of manufacturing efforts) -- commonly referred to as products.

**HL7 Enterprise Architecture Products:**

Any specification or balloted standard that HL7 produces. Products may also be defined as "manufactured composite structures that are assembled from engineered architecture primitives."

**HL7 Foundation Components:**

Certain V3-related artifacts that have been identified as essential components of either the HL7 Enterprise Architecture or the HL7 SAIF.

### 1.3.3.3. HL7 Foundation Components

Several HL7 foundation components are being integrated into SAIF and the Enterprise Architecture (EA).

**Technical components:**

- Reference Information Model (RIM)
- HL7 Core Principles
- Data types
- Vocabulary binding
- Refinement and localization for the ECCF
- Clinical Document Architecture (CDA)

**Process documentation:**

- HL7 Development Guide (HDF)
- Facilitator's Guide
- Publishing Handbook
- V3 Guide for messaging

### 1.3.3.4. Services-Aware Architecture

This section discusses the services-aware architecture of SAIF.

**Related information**

- Section 1.3.4.2.2. Services-Oriented Architecture
- Section 1.3.4.7. SAIF and the SOA Lens
Section 1.3.5.3.3. History of services in HL7

1.3.3.4.1. Services-Oriented Architecture Definitions

Both the influence of Service-Oriented Architecture on SAIF -- as well as the degree to which SAIF is not SOA -- can be seen in the following definitions of SOA, drawn from the tens (or hundreds) of definitions that abound in the technical and Internet literature on the subject.

SOA is defined as follows:

WIKIPEDIA: In computing, Service-Oriented Architecture (SOA) provides a set of principles of governing concepts using during phases of systems development and integration. Such an architecture will package functionality as interoperable services: Software modules provided as a service can be integrated or used by several organizations, even if their respective client systems are substantially different. An implementation of SOA is called a Service-Oriented Architecture implementation.

Service-orientation requires loose coupling of services with operating systems, programming languages, and other technologies that underlie applications. SOA separates functions into distinct units, or services, which developers make accessible over a network in order that users can combine and reuse them in the production of applications. These services communicate with each other by passing data from one service to another, or by coordinating an activity between two or more services.

OASIS: Service-Oriented Architecture (SOA) is a paradigm for organizing and utilizing distributed capabilities that may be under the control of different ownership domains.

SUN: SOA is an architectural style for building software applications that use services available in a network such as the web. It promotes loose coupling between software components so that they can be reused. Applications in SOA are built based on services. A service is an implementation of a well-defined business functionality, and such services can then be consumed by clients in different applications or business processes.

SOA allows for the reuse of existing assets where new services can be created from an existing IT infrastructure of systems. In other words, it enables businesses to leverage existing investments by allowing them to reuse existing applications, and promises interoperability between heterogeneous applications and technologies. SOA provides a level of flexibility that wasn’t possible before.

INTEL: The term SOA stands for Service-Oriented Architecture and this is where the consensus over the definition stops and the debate begins. Numerous standards organizations such as W3C, DMTF, and OASIS have all made an attempt to own the definition of SOA, with varying amounts of success. Large enterprises such as IBM, HP, and SAP now offer SOA products and services along with a host of smaller ISVs. Finally, IDC estimates that the total market for SOA software and services combined will be over USD 31 billion by 2009. With so much potential, how is it possible that the definition of SOA remains a contentious battleground?

The ultimate aim of this section is to delve into the reasons why the SOA concept has many stakeholders and many definitions. Rather than exhaustively comparing and contrasting the existing definitions, we investigate the underlying foundations of the service concept and try to understand the motivations behind SOA as well as the debate around properly defining it.

Related information

(1) This definition of SOA comes from an article entitled “Service-Oriented Architecture (SOA) and Web Services: The Road to Enterprise Application Integration (EAI),” by Qusay H. Mahmoud of Sun Microsystems.

(4) This definition of SOA comes from a recent book by a group of authors from Intel, including Joe Natoli, entitled “Service-Oriented Architecture Demystified: A pragmatic approach to SOA for the IT Executive.” See: http://www.amazon.com/Service-Oriented-Architecture-Demystified-pragmatic/dp/1934053023
1.3.4.2. Services-Awareness and Working Interoperability

HL7 has increased its scope of interoperability technology to include services, which include business-process services that require alignment of policies and processes to work.

This services-aware approach focuses the SAIF on several core concepts including the following:

• Explicit expressions of behavioral semantics
• Formal, layered strategy for assessing conformance and compliance
• Formal governance

In a pure SOA environment, these constructs appear in particular architectural and technological representations and implementations. In contrast, in SAIF, these constructs are pivot points that manifest themselves somewhat differently, depending on the HL7 Interoperability Paradigm -- messages, documents, or services -- in which they are applied.

In particular, SAIF is based on the belief that an HL7 Enterprise Architecture Specification can be developed successfully only if a set of frameworks is in place to guide the emergence of the EAS. The SAIF Behavioral Framework, Enterprise Conformance and Compliance Framework, Governance Framework, and Information Framework are those frameworks.

HL7’s premise is about describing Working Interoperability and including a services technology as one of the means of achieving it. All parties that intend to achieve working interoperability need to agree on the set of semantics that are associated with achieving their mutual objective. SAIF makes those semantics explicit, rather than requiring implementers to interpret implicit semantics and then work out what they have to implement explicitly. The more implicit, the less the interoperability solution will scale. Getting the scope of interoperability to scale up to regional, national, or even the international level is the challenging part.

Note: To date, the documentation on SAIF has not defined the specific ways in which these services-aware principles are manifested in the various HL7 interoperability paradigms.

1.3.4. SAIF Core Principles

This section covers the SAIF core principles and the objectives of the SAIF project.

1.3.4.1. Objectives of the SAIF Project

To fulfill these objectives, SAIF draws on a combination of forces. In particular, SAIF is best understood through a set of lenses (perspectives), also called the "Lens of SAIF."

Specifically, this overarching lens includes Service-Oriented Architecture (SOA), Model-Driven Architecture (MDA), Computable Semantic Interoperability (CSI), and Reference Model for Open Distributed Processing (RM-ODP). In addition, SAIF brings to the "combination of forces" the considerable knowledge base that the Reference Information Model (RIM), Abstract Data Type (ADT) specification, vocabulary constructs, Clinical Document Architecture (CDA), and CSP, and so on, represent.

The overall objectives of the SAIF are as follows:

• Facilitate organization-wide development of services-aware specifications that enable computable semantic interoperability between HL7 trading partners.
• Enable "Unified Field Theory," that is, contextualization and use of SAIF principles, processes, and practices in
the development of all HL7 interoperability paradigms (messages, documents, and services).

- Define a Behavioral Framework (BF) that enables explicit specification of contract-based integration derived from service-role specifications.

- Define a layered Enterprise Conformance and Compliance Framework (ECCF) that supports full, bidirectional traceability of requirements and combines the essence of Model-Driven Architecture (MDA) with that of RM-ODP.

- Define a robust Governance Framework (GF) that explains both the internal SAIF-based governance of HL7, as well as the impact of SAIF on HL7’s relationships and interactions with other Standards Developing Organizations (SDOs), specification boundaries, regulators, and so on.

1.3.4.2. Lens of SAIF

SAIF represents a synthesis of best practices and concepts from multiple frameworks.

SAIF was developed through several “lenses” including:

- Service-Oriented Architecture (SOA)
- Model-Driven Architecture (MDA)
- Computable Semantic Interoperability (CSI)
- Reference Model for Open Distributed Processing (RM-ODP)

Each lens brings to SAIF a particular set of perspectives, approaches, and constructs, as described in the following paragraphs. The collective perspective of the various lenses is filtered through the overarching lens of HL7 (including relevant input from the Health Level 7 Development Framework (HDF) and Core Principles) to form the composite Lens of SAIF. SAIF provides the goals, artifacts, portions of a methodology, and a framework for defining the HL7 EAS as a robust, durable business-oriented set of constructs that provide extensibility, reuse, and governance.

Related information

- Section 1.3.4.7. SAIF and the SOA Lens

1.3.4.2.1. Lens of SAIF Graphic

The Lens of SAIF graphic shows the composite set of influences from various enterprise architectures on SAIF. Each of these SAIF lenses is discussed briefly.

The following figure shows a graphic of the Lens of SAIF.

Figure 7. The Lens of SAIF shows the composite set of influences from SOA, RM-ODP, MDA, and CSI that HL7 contextualizes to provide the focus for SAIF.
For more details about the influence of Reference Model for Open Distributed Processing (RM-ODP) and Model-Driven Architecture (MDA) on the Lens of SAIF, see the Behavioral Framework and ECCF documentation.

### Related information

- Behavioral Framework
- Enterprise Conformance and Compliance Framework

### 1.3.4.2.2. Services-Oriented Architecture

SAIF is *services-aware*, not SOA-specific; that is, SAIF is not just about services.

In addition, services-awareness brings to an EAS a number of architectural considerations and constructs including:

- The specification of Working Interoperability (WI) interactions using the formal constructs of contracts and service roles
- A formal discipline emphasizing the separation of concerns, or between static and behavioral semantics
- A well-understood formalism for the development of technology-independent specifications
- A well-defined notion of conformance and compliance
- A well-established need for overarching governance

Each of these points is elaborated in further detail in the Behavioral Framework and ECCF documentation.

### Related information

- Behavioral Framework
- Enterprise Conformance and Compliance Framework
- Section 1.3.4.7. SAIF and the SOA Lens
1.3.4.2.3. Model-Driven Architecture

SAIF includes aspects of Model-Driven Architecture (MDA).

The following text describing Model-Driven Architecture is taken from the web site for Object Management Group (OMG):

How Systems will be Built

OMG’s Model-Driven Architecture® (MDA®) provides an open, vendor-neutral approach to the challenge of business and technology change. Based on OMGs established standards, the MDA separates business and application logic from underlying platform technology. Platform-independent models of an application or integrated system’s business functionality and behavior, built using UML and the other associated OMG modeling standards, can be realized through the MDA on virtually any platform, open or proprietary, including Web Services, .NET, CORBA®, J2EE, and others. These platform-independent models document the business functionality and behavior of an application separate from the technology-specific code that implements it, insulating the core of the application from technology and its relentless churn cycle while enabling interoperability both within and across platform boundaries. No longer tied to each other, the business and technical aspects of an application, or integrated system can each evolve at its own pace – business logic responding to business need, and technology taking advantage of new developments – as the business requires.

Of particular importance to SAIF is a layered approach to representing the semantics of a particular specification to satisfy a particular type of Working Interoperability (WI) session.

While MDA only mentions two layers -- platform-independent and platform-specific -- in the text above, subsequent work on MDA and MDA-derived specifications has indicated the usefulness of more than two layers (for example, the Zachman Framework® identifies six layers of a specification).

SAIF has identified three core layers as the basic constructs of SAIF:

- Computationally-Independent Model (CIM), also called Conceptual
- Platform-Independent Model (PIM), also called Logical
- Platform-Specific Model (PSM), also called Physical or Implementable

These layers are most directly manifest in the Enterprise Conformance and Compliance Framework (ECCF).

However, as noted above, these layers also provide a general framework for assessing both the difficulty and value of a given WI session between two trading partners using one or more HL7 specifications.

MDA also supports “services-aware” thinking about the issues surrounding WI. Finally, an ever-increasing set of tools have been designed to support the development of the various artifacts required at each MDA layer, as well as to support the transformations between the various MDA layers. In particular, HL7 is working with the United Kingdom National Program and the Open Health Tools initiative on specific tooling to support SAIF and MDA.

Related information

- Section 1.3.4.2.6. Integrating SOA MDA CSI and RM-ODP
- Section 1.3.5.2.1. Influence of Existing OMG Specifications on SAIF
- Section 1.3.5.1.4.1. SAIF and HSSP

1.3.4.2.4. Computable Semantic Interoperability
The “Four Pillars of Computable Semantic Interoperability” (CSI) are useful in thinking about the “necessary but not sufficient” constructs that must be in place between two parties for them to achieve the computable exchange of information.

**Note:** Although well known and understood for some time in the informatics community, the formal collection and naming of the Four Pillars was first published in an article in the *HIMSS Journal of Healthcare Information Management*, (Vol. 20, No. 1, pp 72-78, in January 2006).

The following lists the original pillars -- which focused on static semantics, only implying the need for equivalent artifacts to specify dynamic and behavioral semantics):

- **Pillar #1**: Common model across all domains-of-interest
- **Pillar #2**: Pillar #1 bound to robust data type specification
- **Pillar #3**: Methodology for binding terms from concept-based terminologies
- **Pillar #4**: A formally-defined process for specifying the static and behavioral semantics of the interoperability scenario

**Related information**

- Section 1.3.5.3.1. How SAIF Got Started
- Section 1.3.2. Value Proposition: Working Interoperability
- Section 1.3.4.4. Four Pillars of CSI
- Section 1.3.4.2.6. Integrating SOA MDA CSI and RM-ODP
- Section 1.3.4.5. Integration and Working Interoperability

### 1.3.4.2.4.1. Bridging Domain-Specific Models

In a Working Interoperability (WI) context as large as that spanned by the collection of HL7 stakeholders, it is often not possible to specify a single model of common semantics.

As a result, it is often necessary to have semantic structures, which harmonize or bridge two domain-specific models that share common semantics. Examples of common semantics might include a Person, Observation, Terminology, and Data type.

As the following figure shows, these bridging models might need to be expressed in a layered fashion involving multiple levels of abstraction. In this example, two domain-specific models share common semantics for Data Type and Vocabulary Element.

*Figure 8. CSI informs SAIF to generate artifacts that express shared semantics across multiple domain models. These models are Design Information Models that subject matter experts developed in different clinical domains.*
1.3.4.2.5. Reference Model for Open Distributed Processing

The HL7 ArB chose the Reference Model for Open Distributed Processing (RM-ODP) as its basis for specifying SAIF and, therefore, the specification language for the HL7 EAS.

HL7 made this choice primarily because RM-ODP -- unlike the somewhat similar Zachman Framework or other enterprise architecture models (for example, Krutchen et al.) -- is an International Standardization Organization (ISO) standard. This distinction aligns culturally with HL7’s posture as an ANSI Standards Developing Organization (SDO) with significant ties to ISO. In addition, the focus on RM-ODP in distributed architectures makes it particularly relevant for the loosely-coupled environment in which HL7 specifications are used.

RM-ODP specifies architectures by using an ontology of terms that talk about various aspects of an enterprise architecture from the perspective of five “viewpoints, which collectively specify and implement a given instance of an EAS.

The viewpoints are not hierarchical and might cover the same area of interest (also called “non-orthogonal”). Although the viewpoints are non-hierarchical, the collection of artifacts associated with a given viewpoint usually is hierarchical so that varying levels of detail and abstraction can be managed readily within a given viewpoint.

Note: HL7 has not yet decided as to whether it should formally adopt the RM-ODP ontology. However, several other organizations using RM-ODP -- the National Cancer Institute, Open Health Tools, and the Australian National e-Health Transition Authority (NEHTA) program have either adopted the ontology “as is” or in a form customized for local usage. The design of the RM-ODP ontology facilitates human-to-human communication between the various stakeholders in a given EAS.

The figures in Section 1.3.4.2.5.1 define and summarize the RM-ODP viewpoints.

Related information

- Section 1.3.4.2.6. Integrating SOA MDA CSI and RM-ODP
- Section 1.3.4.2.7.3. Other HL7 Content

1.3.4.2.5.1. Five RM-ODP Viewpoints

SAIF uses the RM-ODP as its common language to categorize the various artifacts. RM-ODP consists of five viewpoints.

Figure 9. Tabular (top) and graphical (bottom) representations of the five RM-ODP viewpoints along with the associated synonyms for viewpoints 1-4.

---

(5) ISO Standard: RM -- ODP, ISO/IEC IS 10746 | ITU-T X.900

(6) A formal ontology is a controlled vocabulary expressed in an ontology representation language. This language has a grammar for using vocabulary terms to express something meaningful within a specified domain of interest.
<table>
<thead>
<tr>
<th>Why?</th>
<th><strong>Enterprise View</strong>: Concerned with the purpose, scope and policies governing the activities of the specified system within the bounds of that enterprise.</th>
</tr>
</thead>
<tbody>
<tr>
<td>What?</td>
<td><strong>Information View</strong>: Concerned with the kinds of information handled by the system and constraints on the use and interpretation of that information.</td>
</tr>
<tr>
<td>How?</td>
<td><strong>Computational View</strong>: Concerned with the functional decomposition of the system into a set of objects that interact with interfaces, enabling system distribution.</td>
</tr>
<tr>
<td>Where?</td>
<td><strong>Engineering View</strong>: Concerned with the distribution of both computing resources and infrastructure required to support the system, as defined in the Computational view.</td>
</tr>
<tr>
<td>True?</td>
<td><strong>Technology View</strong>: Concerned with the choice of technology used to support system distribution.</td>
</tr>
</tbody>
</table>

Note that in the upper representation of the “five viewpoints,” the Technology viewpoint differs from the other four viewpoints. The Technology viewpoint specifies a particular “technology binding” to the specifications enumerated and categorized by the other four viewpoints, and therefore acts as a “validation instance” of the EA, as specified through the other viewpoints.

The notion of a technology binding as a testable validation of assertions is fundamental to the constructs and processes described in the ECCF.

The Behavioral Framework informs the Computation viewpoint. Meanwhile, more traditional HL7 artifacts, such as RIM-derived artifacts and vocabularies, populate the Information viewpoint.

Finally, as explained in greater detail in the ECCF document, a given SAIF specification stack is made up of conformance statements and compliance validations, which are explicitly stated in the artifacts constructed as constraint patterns sorted by RM-ODP viewpoints.

**1.3.4.2.5.2. RM-ODP Multidimensional Specification Framework**
The value of the RM-ODP multidimensional specification framework provides two distinct advantages from the perspective of building an EAS.

These advantages are:

- In theory, no single viewpoint can completely specify a system as valid, which means that all viewpoints are equivalent (with the caveat on the Technology viewpoint, as mentioned in Section 1.3.4.2.5.1).
- Conformance statements (assertions) made by one viewpoint may be overridden by the others.

In practice, the specification process must ensure that one viewpoint’s assertions are not ignored, re-prioritized, or not aligned by or to the others. Each viewpoint is of equal importance when compared with the others from an Enterprise Architecture perspective. The RM-ODP-based specification framework serves as a check on the overall consistency of the collection of Conformance statements (assertions) except the Technology viewpoint, where the various conformance statements are evaluated as being True or False for the given implementation or technology binding.) This interplay between the viewpoints serves as an EAS Quality Requirement whose Fit Metric is the set of “viewpoint-specific conformance statements (assertions)” that circumscribes the EAS’s definition, design, and implementation.

### 1.3.4.2.5.3. Integration Points

RM-ODP defines four types of integration points -- critical points in a Working Interoperability (WI) context -- that serve to focus the various viewpoint specifications.

The classification of integration points is included here for future reference. The four types are:

- **Programmatic** (establishes a point of integration at which one component may be substituted for another)
- **Perceptual** (a reference point at which some interaction occurs between the physical world and the component)
- **Interworking** (a reference point that allows communication between two or more systems)
- **Interchange** (the point at which an external storage medium is introduced into a system)

As SAIF evolves and translates into specific HL7 Specification Process Guidelines, integration point types may be used to classify further the Compliance Assessment Processes. (See the ECCF section.)

### 1.3.4.2.6. Integrating SOA MDA CSI and RM-ODP

The following diagram depicts a system prototype that integrates elements of Services-Oriented Architecture (SOA), Model-Driven Architecture (MDA), Computable Semantic Interoperability (CSI), and RM-ODP.

*Figure 10. A process-independent, artifact-based, prototypical integration of these four lenses.*
1.3.4.2.7. HL7 Artifacts

HL7 is the overarching lens that contextualizes Services-Oriented Architecture (SOA), Model-Driven Architecture (MDA), Computable Semantic Interoperability (CSI), and RM-ODP.

One of the goals of SAIF is to take the enterprise architecture approaches and best practices from the other four "lenses," and contextualize them for HL7 business. One of the most critical aspects of this contextualization is to use as much as possible the existing HL7 artifacts and processes -- for example, Refined Message Information Models (RMIMs), data types and vocabulary practices -- as well as complying with and harmonizing SAIF constructs and processes with both the HL7 Development Framework (HDF) and Core Principle documents.

The following discussion outlines the current thinking as to which of the existing HL7 artifacts that can be used in the context of SAIF. It also indicates areas and specific artifacts that do either not map directly to SAIF without
modification or – in a few cases – will need to be developed as new. Given the rich history of HL7 in specifying robust static semantics, most of the artifacts “to be developed” center on the Behavioral, Enterprise Conformance and Compliance, Governance, and Information Frameworks.

Reference: For more details on the SAIF sub-frameworks, see the following sections:

**Related information**

- Behavioral Framework
- Section 1.4. Enterprise Conformance and Compliance Framework
- Governance Framework

### 1.3.4.2.7.1. HL7 Artifacts Inventory and SAIF Survey

Theoretically, the SAIF recognizes the critical and essential importance of the rigor of the frameworks, processes, artifacts, and tools.

Therefore, one of the guiding directives of the *Jump Start* project was to establish the maximum reuse of and derived value from the body of work done over the past 13+ years on HL7 Version 3. This project also applied, whenever possible, the same metrics to the work that was being done for HL7 V2.x., which was developed for specifying static semantics.

It was assumed from the outset of the *Jump Start* effort that the following artifacts (just to mention a few HL7 specifications) would be of considerable value, regardless of the Interoperability Paradigm in which SAIF was applied (services, documents, or messages):

- The Reference Information Model
- The Abstract Data Type specification
- The Electronic Health Record Functional Model
- The Clinical Document Architecture

In particular, the SAIF uses existing HL7 artifacts and specifications for static semantics in a SAIF-defined constraint pattern that provides traceability from requirements through analysis artifacts--artifacts derived from Domain Analysis Models, design artifacts (such as RMIMs and Hierarchical Message Definitions (HMDs), and finally implementation artifacts, such as XML schema definitions (XSDs).

The only caveat to this statement is that one of the main conceptual differences between a *messaging* paradigm and a *services* paradigm is that when services are modeled after *business capabilities*, considerable differences occur in the granularity of a *capability* service specification and a *message interaction* specification. In addition to the granularity issue, a services-aware behavioral framework is, by definition, very explicit about formally expressing the various aspects of a given interaction. For example, HL7 Application Roles and Receiver Responsibilities may need to be defined more explicitly than is currently required.

**Note:** Differences between *operations* specified by a services-awareness paradigm and *interactions* specified by a message-aware paradigm are not considered to fundamental hurdles for artifact reuse because preliminary experience has shown these semantics to be readily mappable between the two domains.

For a detailed discussion of the related topics, see the Behavioral Framework and ECCF documentation.

**Related information**

- Behavioral Framework
- Enterprise Conformance and Compliance Framework
1.3.4.2.7.2. Non-Payload Message Content

The interface contract assimilates many of the semantics that have historically been reserved for the non-payload content of V3 messages, including -- but not necessarily limited to -- the MessageCommunicationsControl classes, the Control Wrappers, and portions of the Transmission Wrapper.

Rarely, these semantics are lost. Usually these semantics are preserved either implicitly in the architecture or explicitly in the interface contract. Thus, in a “pure SOA environment” – when one is restricted to just services using the service HL7 Interoperability Paradigm – the wire-format needed for many of the structures found in V3 message Control and Transmission wrappers would be made unnecessary by semantics at the two interacting service interfaces.

1.3.4.2.7.3. Other HL7 Content

Apparently, the current representations of most of the other HL7 static content are largely misaligned with the representations that this semantic content can assume as part of a service specification.

Examples of HL7 static content include:

- Clinical Document Architecture (CDA)
- Vocabulary
- Clinical Context Management Specification (CCOW)
- Data types
- Electronic Health Record System Functional Model (EHR-SFM)
- GELLO(7)
- Personal health records

The current HL7 V3 artifacts that are associated with the RM-ODP Information viewpoint may not be reused “as is.” The structure and meaning would be reusable even if the representation of the semantics were algorithmically transformed to a representation that is more consistent with a services specification. This means that the unified serialized information model that the messaging paradigm uses is broken up into smaller models that are closely aligned with service operations, but no meaning is lost.

In addition, some (or all) of these specification types may benefit from SAIF practices and frameworks that surface the various dimensions of CSI and formalize notions of conformance, compliance and governance. SAIF brings an organizing principle to the development of these non-service specifications.

SAIF provides a layered, rigorous way of capturing behavioral and functional semantics that have often been implied, but never expressed, in existing HL7 work.

The current HL7 messaging dynamic model cannot represent adequately the complexity of desired business behavior. The service specification expression of behavior more appropriately captures and expresses the necessary semantics to accommodate complex interfacing behavior. One can use the service specification to produce reusable software components more directly, rather than leaving so much behavior implicit for developers and interface implementers to make explicit.

Note: In some cases, these behavioral and functional semantics amount to design decisions regarding the desired behavior involved in a particular standard – this is appropriate in certain circumstances for a Standards Development Organization (SDO), as long as the specification provides explicit enumerations of the behaviors as well as the potential policy effects of adopting the standard in question.

(7) GELLO is an object-oriented query and expression language for clinical decision support.
1.3.4.2.7.4. Legacy Artifacts, Interoperability Paradigms, and the Unified Field Theory

The HL7 SAIF intentionally builds on the successes of much prior HL7 work, including the Electronic Health Record System Functional Model (EHRSS-FM), the Healthcare Service Specification Project (HSSP), the Clinical Document Architecture (CDA R2), the Clinical Statement Pattern (CSP), various work products of the Vocabulary Work Group, and the Abstract Data Types Specification (R2).

In addition, the SAIF is designed to facilitate integration of HL7 specifications with the work of other ANSI, ISO Standards Development Organizations (SDOs), and de facto standards-making bodies, such as Clinical Data Exchange Standards Consortium (CDISC) and Organization for the Advancement of Structured Information Standards (OASIS). The lessons learned through the crafting of these artifacts -- as well as the sharing of these artifacts across various enterprise and organizational boundaries -- have served as a consistent touchstone throughout the Jump Start effort.

In particular, although this document focuses on service specifications that HL7 produces, many of the principles, conformance and compliance artifacts, and processes defined in the ECCF component of the SAIF will have application -- possibly with slight modifications -- to other HL7 interoperability paradigms, such as messages and documents.

In particular, the SAIF supports -- in the context of its explicit conformance and compliance model -- the ability to define and extend organizational governance models to incorporate specifications in any Interoperability Paradigm. In addition, SAIF specifies a metamodel for an HL7 Behavioral Frameworks and Services that aligns with two industry standards, SOA Modeling Language, referred to as SoaML (OMG) and Web Services Choreography Description Language, referred to as WS-CDL (W3C).

The SAIF aligns with the recent work within several national organizations, including the U.S. Department of Defense (DoD) and the NCI Cancer Bioinformatics Grid (caBIG® ) programs, and the Canada Health Infoway effort. In addition, much of the Jump Start work appears to align with the recent work from Bernd G.M.E. Blobel. In addition, the Open Health Tools (OHT) Architecture Project team recently adopted the ECCF as a cornerstone of the overall architecture framework for the specification of interoperable tools to support healthcare and life sciences.

This uniform application of the principles and practices of the SAIF to documents, messages, and services within HL7 is nicknamed as the Unified Field Theory for (hopefully) obvious reasons.

See the Behavioral Framework section for details.

1.3.4.3. Working Interoperability Requirements

This section covers the Working Interoperability requirements.

The requirements for the components that enable Working Interoperability (WI) include the need to define:

- Semantics and structure of the data and information to be exchanged
- Inputs, outputs, and transformations (such as functions) that enable the exchange
- Traceable mappings of functions to real-world events and business processes
- Reference terminologies or language for sorting and discussing the above processes
- Engineering and deployment topologies
- Technology bindings to achieve WI

In the context of WI, one must distinguish between syntactic and semantic interoperability. In addition, one must understand the difference between human semantic and computable semantic interoperability. These terms are defined as follows:
Syntactic Interoperability:

Guarantees the exchange of the structure of the data, but carries no assurance that all parties will interpret the meaning identically. Web pages built with HTML and/or XML are good examples of machine-to-machine syntactic interoperability because any machine with a web browser can read a properly structured page. However, the meaning of the page to a particular machine may vary substantially. This not usually deemed to be a problem since the semantics of a page’s contents are meant to be interpreted by human viewers. The ability of browsers to display HTML pages regardless of the implementation technology (such as the browser) or the content of the web page is an example of syntactic interoperability.

Human Semantic Interoperability:

Guarantees that the meaning of a structure is unambiguously exchanged between humans. Documents such as progress notes, referrals, consults, and so on, rely on the specificity of medical vocabularies and common community practice to guarantee semantic interoperability at a clinician-to-clinician level. The ability of a human being to read a clinical discharge summary formatted in multiple ways in multiple contexts and still extract the “true meaning” irrespective of its presentation is an example of human semantic interoperability.

Computable Semantic Interoperability (CSI):

Requires that the meaning of data be unambiguously exchanged from machine to machine. Note that this does not necessarily mean that all machines need to process the received data the same way, but rather that each machine will make its processing decisions based on the same meaning. CSI is often discussed by describing the Four Pillars of CSI.

Related information

▶ Section 1.3.4.2.4. Computable Semantic Interoperability

1.3.4.4. Four Pillars of CSI

This section describes the Four Pillars of Computable Semantic Interoperability (CSI) in more detail.

The pillars are as follows:

- **Pillar #1**: A common domain analysis model of shared semantics (static and dynamic)
- **Pillar #2**: The binding of each attribute in the static information model to a robust data type specification (for example, the HL7 V3 Abstract Data Type Specification or the ISO 21090 Data Types Standard)
- **Pillar #3**: A rigorous and reproducible methodology for binding static attributes to terms from concept-based terminologies
- **Pillar #4**: A formally-defined process for defining the static structures and interaction profiles that collectively define the particular information that is exchanged between trading partners

1.3.4.4.1. CSI Pillar 1: Shared Semantic Models

CSI Pillar 1 is a methodology for specifying shared semantic models.

Models of shared semantics may exist at several levels of abstraction. The most common level is that of an analysis or information model which identifies core classes, attributes, and relationships in a given domain-of-interest.

However, terminology and data type models are also examples of shared semantics. The most important notion about shared semantics is that the models that define the semantics need to be implementation free. Thus, logical

---

data models that often contain implementation details, such as primary keys, foreign keys, and indexes, are not
considered good examples of models of shared semantics. In addition, it is often desirable to have one or more models of shared behavioral (dynamic) semantics. Such models need to be clear so as not to confuse process flows and work flows, which are often not readily shareable with
contract, service role, or business capability specifications, examples of artifacts, which can productively specify
shared dynamic and behavioral semantics. The SAIF Behavioral Framework specifies a metamodel for documenting
the shared semantics of functions, behaviors, and interactions.

1.3.4.4.2. CSI Pillar 2: Computable Complex Data Type Specification

CSI Pillar 2 is a methodology for specifying a complex data type specification. SAIF expects that all artifacts required for achieving CSI need to use a shared data type representation, such as ISO 21090 or its semantically isomorphic specification, the HL7 Abstract Data Type R2 specification in one of its several implementations.

1.3.4.4.3. CSI Pillar 3: Methodology for Building Structures to Support CSI

CSI Pillar 3 is a methodology for building structures to support CSI (or, in SAIF terms, Working Interoperability). However, to achieve true WI at a computational level, formal methodologies must support the concepts of terminology binding in an area where the binding might not be as clean and simple as it would appear, because multiple representations of the same semantics exist on both sides of the binding.

1.3.4.4.3.1. Example: Penicillin Allergic Reaction

The "Grade IV Penicillin allergic reaction" example shows the ideal binding of information and terminology model instances, and the Terminfo problem.

The following example can be represented in several ways by using various combinations of Reference Information Model (RIM) structures and SNOMED-CT codes, thereby creating structures that are not computationally semantically interoperable.

The Terminfo problem illustrates the difficulty of taking the information structure that is derived from the HL7 Reference Information Model and expressing it through the SNOMED-CT codes. SNOMED-CT permits the expression of several related terms in a formal way.

Figure 11. The top half of the figure shows the ideal binding of information and terminology model instances, and the lower half represents the "Terminfo problem."
1.3.4.4. CSI Pillar 4: Methodology for Using Pillars 1 to 3 to Achieve Computable Semantic Interoperability

CSI Pillar 4 is a methodology for using Pillars 1 to 3 to achieve CSI, (or for SAIF, Working Interoperability). This pillar is the essence of much of the SAIF.

Note: Although the above framework, as traditionally applied (in object-oriented design and programming), is concerned with both static and dynamic semantics, it does not address how specific information exchanges or component interactions are described, registered, advertised, discovered, bound to particular solutions, or governed. These concerns are addressed within the specification context of services-aware EAS, and are discussed in detail in the context of "contracts" and "service role specifications" in the Behavioral Framework documentation.

1.3.4.5. Integration and Working Interoperability

The Four Pillars of Computable Semantic Interoperability or -- more specifically, Working Interoperability -- is, for implementers, a somewhat vague and "ivory tower" concept when it is used in the general sense. To implement Working Interoperability in a project, take the following steps to establish:

• Measurable goals
• "Plug and play" implementation patterns
• Incremental adoption patterns with associated incremental benefit

Normally, where more than one implementer is involved in a given project, the specifications (whether they are developed by SDOs, other organizations, internal to the implementer's organization by third parties, or by the implementer), must be implementable. In turn, this means that for shared specifications, there must be governance regarding specification details. These details must focus -- from the implementer's perspective -- on making specifications testable. For example, a Conformance and Compliance model must fit the way that the implementer's organization uses and tests software components and WI.

The Implementation Guides help the implementer answer critical questions like, "Are you ready? How does this work with our new ABC Interface Engine?" Developers of specifications must remember the following important points:

• Any single instance of integration is simple, though not necessarily easy.
• Developers of any single system can understand it well enough to allow integration with any other single system.

• Achieving component integration is, in the end, a manufacturing or implementation issue; it “simply” involves the deployment of computational components.

• On the other hand, achieving Working Interoperability is an engineering or architectural issue that enables the creation of multiple, context-specific engineering integration solutions. (John Zachman).

• Engineering for Working Interoperability is an architectural concern that enables the creation of multiple context-specific integration solutions.

• The complexity and high change of healthcare requires Working Interoperability, which, in turn, requires an Enterprise Architecture.

• Enterprise Architecture is more than just an engineering or technology perspective.

1.3.4.6. Enterprise Integration Paradigms

SAIF brings a "services-awareness" perspective to the development of specifications. The result and proof of this perspective is most notably present in the context of Enterprise Integration Patterns.

In particular, a "services-awareness" perspective forces the specification developer to express explicitly the following Integration Dimensions semantics:

• Static (the data)

• Functional (actions performed, data in and data out)

• Behavioral (observable results of functions and contracts)

• Business context (who is interacting where and why?)

Note: The semantics of the Integration Dimensions map to the Informational (static), Computational (functional and behavioral), and Business/Enterprise (business context) RM-ODP viewpoints. The Engineering, Deployment, and Technology viewpoint is only indirectly involved in the Integration Dimension specification, as it is invoked in the context of RM-ODP integration points, where an implementer must provide testable code written to enforceable test criteria (RM-ODP conformance statements).

1.3.4.7. SAIF and the SOA Lens

This section summarizes the influence of Service-Oriented Architecture (SOA) and services-awareness on SAIF. Service principles are essential for an enterprise-level service specification.

For the SAIF, the idea of services has been treated primarily as an abstract notion that provides the following:

• Encapsulate the integration semantics in abstract interfaces that are constrained and profiled.

• Provide a layered stack of specifications that provide traceability from Analysis Specifications, encapsulating many conceptual definitions of the integration semantics, through to platform-bound specifications.

• Push some of the integration semantics into the collaborative portions of the interoperability services contract specification; that is, the contracts that realize the interaction agreement between trading partners.

In other words, the SOA oval has not played a dominant part in the SAIF, except as a useful abstraction under which to specify capabilities. However, services -- even as abstractions -- have many attributes that clearly show some gaps.
in the HL7 pantheon of work products, and suggest some potential evolutions of current work that can help to fill these gaps and clarify other parts of the HL7 content.

The following shows the SOA lens. The red dot represents SAIF.

*Figure 12. The SOA lens.*

- How to answer the hard questions around building services (like granularity, exposure, and conformance)?
  - Service principles
  - service classification system
  - specification stack
- How to specify collaborations? Use the Behavioral Framework
- How to classify work for different levels of accountability? Use a service classification system.
- How to fit an HL7 standard into an organization?

These include explicit placeholders and characterizations for functional and dynamic semantics, the important community acceptance of SOA in fitting into various organizations' system designs, and a means of defining separated concerns and unities of purpose both in the creation of standards and in supporting standardized components.

These elements represent a service-oriented approach to various components in the HL7 Enterprise Architecture (EA), and in some ways, count on the foundation represented in the SAIF before they can be crafted. Nevertheless, these elements may represent future normative portions of the HL7 EA, and potentially in the SAIF. Some of these specific components are listed below:

Because services cannot directly leverage the object-oriented principles (encapsulation, polymorphism, and inheritance), it is necessary to construct an equivalent set of principles that support service-based definition and design. Once these service principles have been defined and fully specified, they can be explicitly referenced during HL7 specification development. These principles are required because in their absence, several critical interaction details are left to assumption, inference, or interpretation. Therefore, in the context of the HL7 Enterprise Architecture, service principles play a critical role in successful service design.
Essential Service Principles

SAIF has been developed with the belief that useful constructs for Working Interoperability can be drawn from services-awareness. These constructs are necessary for integrating systems across the enterprise. Services and services-aware perspectives focus on several service principles. These principles are essential for an enterprise-level service specification, which must explicitly define testable, verifiable multidimensional service contracts.

- Virtualization
- Composition
- Unity of purpose and separation of concerns
- Parsimony
- Technology independence
- Specifications supporting layered conformance and compliance

1.3.4.7.1. Virtualization

This section provides a definition, rationale, and implication for the service principle of virtualization.

Definition: Virtualization refers to the notion of providing a reusable abstraction within an architecture that embodies certain integration and contractual semantics. The semantics should encompass different layers of specifications, from the high-level, requirement-driven terms of integration down through quality-of-service agreements expressed through the choice of deployment platform. Thus, virtualization is about “contextual simplification,” because a virtualized service “hides” those features of the services behavior that are not relevant in the current context. When virtualization is used as a service principle, virtualization should not be confused with notions of physical presence or availability. (The notion of exposure and hiding of irrelevant details has the same general sense as virtualization in the more specific, physical distribution sense of the term.)

Rationale: Because services are intended to provide mechanisms for business-level integration, they must support some level of virtualization. However, because services are intended to support the explicit reuse of an integration context, only behaviors that are entirely within the integration context should be completely virtualized.

Implications: The boundaries of a service are described, in part, by the semantic components that support integration and, in part, by the intended purpose of the service. (For example, a service that handles notification is generic in functionality and can be reused in many scenarios, whereas a business-level service is scoped precisely to a business need of the organization).

1.3.4.7.2. Composability

This section provides a definition, rationale, and implication for the service principle of composability.

Definition: Services should be reusable and callable by other services in contexts and interactions that are validated at runtime with contract semantics.

Rationale: Service operations should be composed behind other services or solutions. For example, a service operation, "Admit Patient" may call an operation on another service to "Validate whether the patient exists in the system."

Implications: Services should be built in such a way that the behaviors and information that are used to describe the interface are deterministic, unambiguous, and semantically significant in terms of service use and composability.

1.3.4.7.3. Unity of Purpose and Separation of Concerns

This section provides a definition, rationale, and implication for the unity of purpose and separation of concerns.

Definition: Services should be created with the goal of providing as little redundant functionality as possible. The
result of this concern leads to the equivalent of unity of purpose and separation of concerns. In addition, separation of concerns, as a guiding service principle, leads to recognition and definition of various aspects of behavioral patterns, domain requirements, business needs, and technological convenience.

**Rationale:** Certain services are more granular than others are. A service might provide fine control over some business entity (for example, "Person") or business function (for example, “Manage Lab Order”). Other services may aggregate these basic services to provide reusable functionality to the enterprise. Still other services may embody a business process that requires a high degree of specification ("Manage Patient Admission").

**Implications:** As different classifications of purpose exist within a distributed architecture, the need for a classification scheme and taxonomy of services exists. Because service specifications support unity of purpose, they also provide an architectural approach -- rather than a design approach that object-oriented programming might manifest-- for separating concerns between components. Thus, a single service can provide an authoritative source-of-record for managing and accessing domain classes. From a design point of view, aggregating all of the various concerns regarding concurrency behind a single virtualization boundary is both effective and efficient from an architectural perspective, which enables cleaner communication between components. From an organizational (business) standpoint, separation of concerns allows services to be tied to jurisdictional boundaries. For example, a cancer center might be expected to set up a Person Service as a source-of-record for their people. This can aid in aligning an organizational map with an IT map.

### 1.3.4.7.4. Parsimony

**Definition:** "Make everything as simple as possible, but not simpler." (A. Einstein)

**Rationale:** The creation of service-oriented architecture should be governed by a desire to:

- Reduce redundant functionality.
- Create reusability.
- Create usable interoperable interfaces.

**Parsimony** is a principle that encourages the creation of simple components to provide solutions to complex problems.

**Implications:** Parsimony provides that architecture guide the design of systems rather than the other way around. Organizations that adopt a Service-Oriented Architecture (or individual services) in the context of the ECCF should exhibit parsimony in how and where they define their service structures.

### 1.3.4.7.5. Technology Independence

This section provides a definition, rationale, and implication for technology independence.

**Definition:** Services should be specified in a technology-neutral fashion, that is, more than one technology binding can be used to validate the multidimensional assertions associated with a given service specification.

**Rationale:** Services should be realized using multiple technologies that are derived from a single specification. This is due to the nature of technology as a meeting a set of requirements through its own capacities (for example, bandwidth utilization, performance).

**Implications:** Services that are bound to a technology may be bound to a service level agreement (SLA), which may be bound to a particular consumer or collaboration. SLAs are an essential part of the service contract that can specify quality of service levels and other contextual elements of the implementation of the service, but should not define the architectural requirements that a service meets.
**1.3.4.7.6. Support for Layered Conformance and Compliance Framework**

This topic provides a definition, rationale, and implication for the support for the *layered conformance and compliance framework*.

**Definition:** A robustly specified service describes its static and behavioral semantics sufficiently to allow for the quantitative assessment of implementations of the service in terms of the implementation’s compliance to a formal conformance framework with the goal of enabling trading partners to assess quantitatively their degree of interoperability for particular service implementations.

**Rationale:** Because service semantics can be realized in a variety of technologies, they provide a logical means of measuring compliance to a published set of conformance statements. In particular, a specific technology binding for a given service provides a testable implementation of that service's multidimensional service contract specification. Interoperability is often not seamless. It must be measurable without complete technological agreement.

**Implications:** Usage of a service implies conformance and acceptance of the multidimensional semantics of integration and any service specifications to which a given implementation adheres. Given that these semantics should be explicit, it may be necessary for different trading partners to agree at different levels of compliance to a set of conformance statements. Consequently, organizations must carefully select and specify which services are exposed externally. For the same reason, it is imperative for architects to make integration semantics explicit instead of implicit to promote appropriate use and reuse.

---

**1.3.4.7.7. Services and Services-Awareness Summary**

In summary, the notion of *services and services-awareness* provides a coherent set of concerns, constructs, artifacts, and processes that collectively embody the SAIF:

- A defined set of artifacts that express the semantics required to achieve Working Interoperability.
- A framework that allows these semantics to be specified in a technology-neutral manner.
- A framework in which formal conformance and compliance may be readily defined.

SAIF is not technology-specific. Rather, SAIF is using critical notions from the world of SOA as a framework for approaching the problem of WI. In particular, provides SAIF with a set of constructs that are useful in describing the semantics of loosely-coupled distributed sharing of information and functionality. SAIF is not defining “services” per se.

The term *services* is not equivalent to Web Services. Web Services are a specific technology. Services are more general; they might be implemented with Web Services or they might be other types of distributed components that use messaging for integration.

For more information, see the Behavioral Framework information in the SAIF book.

---

**1.3.5. Introduction Appendixes**

The Introduction appendixes include background information about SAIF, such as the implications for HL7, relationships with other Standards Development Organizations, and a history of SAIF and services in HL7.

**1.3.5.1. Appendix A: Implications for HL7 -- Impact of Change**

This section describes the SAIF Alpha project, and the impact of change to HL7 work groups and other organizations.
At the September 2008 Working Group meeting, the HL7 Technical Steering Committee (TSC) and the HL7 Board of Directors (BoD) formally adopted SAIF as the "path forward" to specifying an HL7 Enterprise Architecture.

Both governance bodies (as well as the ArB) are aware that adopting and implementing the SAIF can be expected to have impacts to both HL7 itself, as well as between HL7 and other organizations.

However, the ArB, TSC, and BoD see the continued development and initial deployment of the SAIF as needing to start “sooner than later” even though not all of the details have been worked out. In other words – given that all involved realize the complexity of the task -- the development and deployment of the SAIF will proceed using an iterative and incremental approach similar to a software engineering Best Practice in the development of complex systems. As such, the TSC began outlining the basics of an Alpha SAIF Deployment Project shortly after the September 2008 meeting. The project is now staffed and in progress. The project focuses on implementing the SAIF processes and generating the required SAIF artifacts in at least one project for each of the three HL7 interoperability paradigms (messages, documents, and services). The ArB has identified candidate projects.

The following is a discussion of the basic outline of that project, contextualized with the ArB’s thinking about the suggested next steps in SAIF development as well as its assessment of the impact of SAIF deployment both within and outside HL7. Central to any thinking about impact are issues, such as backwards compatibility of artifacts, process, governance, and scalability. In turn, the overarching theme that covers these topics is education and change management.

### 1.3.5.1.1. SAIF Alpha Project Implementation

The HL7 Board of Directors (BoD) and Technical Steering Committee (TSC) believe that SAIF is now stable enough to begin a set of “pilot” implementations that focus on the following activities:

- Produce SAIF-compliant message, document, and service specifications
- Use and provide feedback for SAIF-centric education
- Perform under SAIF-triggered change management processes
- Report to the TSC or its designate

The ArB is currently developing the content and core educational materials that define and explain SAIF. The TSC is focusing on the topics of change management, additional education (particularly organizational change and process integration), and oversight of the Alpha Project to ensure that each of the three HL7 interoperability paradigms is adequately addressed by and integrated into the SAIF.

### 1.3.5.1.1.1. SAIF Rollout Timeline

The Technical Steering Committee (TSC) has the primary responsibility for the implementation of the SAIF.

The following figure shows a high-level proposed SAIF rollout timeline as of September 2009, as defined and approved by the TSC. The timeline will become more detailed and possibly change as the SAIF is circulated, discussed, evolved, and positioned for adoption throughout the organization.

*Figure 13. Proposed SAIF rollout timeline.*
1.3.5.1.1.1. Alpha Project Launch

This SAIF document is one of the primary deliverables for the TSC-sponsored “SAIF Alpha Project” that is currently being developed. The Alpha projects include several trial projects.

The intent of the Alpha projects is to determine which artifacts are useful for various topics, and to consider what changes to processes would be necessary to ensure alignment of artifacts across specifications. The participants in the Alpha project might also identify quality criteria for artifacts.

As of March 2010, seven Alpha projects are working with SAIF:

- National Cancer Institute (NCI)
- National Health Service (NHS)
- Clinical Document Architecture (CDA) R3
- Privacy and Security Services (PASS)
- Canada Infoway
- Orders and Observations Lab (HL7)
- Implementable Technology Specifications work group

For details of the Alpha project, see the ArB wiki for the meeting minutes.

1.3.5.1.2. Best Practices and Patterns

SAIF should follow Enterprise Architecture best practices and patterns.

A good place to read about Enterprise Architecture best practices is the Institute for Enterprise Architecture Development web site. This web site contains useful information about research, best practices, measurement tools, guidelines, and development and implementation programs for enterprise, application, and technology architectures.
1.3.5.1.3. Existing Interoperability Paradigms

This section discusses the HL7 intra-organization implications of implementing SAIF using existing interoperability paradigms.

1.3.5.1.3.1. Analysis Specification -- Domain Analysis Model

The HL7 Development Framework introduced the Domain Analysis Model (DAM) several years ago. The DAM specifies information content and the expected exchange patterns.

One of the implications of adopting SAIF, when it is rolled out, all specifications must have DAM content that describes the requirements that the design artifacts are meeting.

The analysis specification is a complete Domain Analysis Model (DAM) that includes both static and dynamic semantics of the business context that an HL7 service (or potentially other specifications) can support. The ArB considers the analysis specification to be an essential success factor in achieving Working Interoperability. This analysis specification provides a technology- and design-independent view of the semantics of the particular interoperability paradigm component being specified. The main reasons for placing a DAM in this central and pivotal position are as follows:

- Provides direct traceability between requirements and logical (design) and implementation artifacts (a statement equally applicable regardless of whether the specification in question is for a service, message, or document).
- Binds together the conformance statements for the Information and Computational viewpoint to ensure their consistency.
- Provides essential placeholders for the remaining RM-ODP viewpoints.
- Provides placeholders for the policy and governance discussions, which arise both within and from a given specification.

The figure provides a high-level graphical summary of the above points.

1.3.5.1.3.2. Process for Developing Interoperable Applications

Analysis specifications also provide the initial entry point for collaborating organizations to assert conformance. The first level of the analysis specification is named conceptual compliance. This section also describes the specifications for the interoperability paradigms (services, documents, and messages).

While it is true that implementation details at this level are typically somewhat scarce, grounding an integration effort on an explicit set of conformance statements that include difficult notions such policy and governance is a critical success factor on the road to achieving Computable Semantic Interoperability (CSI).

Figure 14. Differences and similarities between the specifications delimited by the HL7 interoperability paradigms (messages, documents, and services)
Appropriately, the logical specifications of services messages and documents differ from the single analysis-level specifications in three important ways: functional, informational, and in the encapsulated behaviors of the structures. Despite the apparent separation, this specification pattern provides a “Unified Field Theory” for HL7 artifacts. With the addition of appropriately contextualized functional and behavioral semantics (in part within an appropriate Behavioral Framework), a given specification can adequately express the semantics of integration for consumers.

The ArB believes that the common analysis-level artifacts that the SAIF specifies across each of the three major HL7 interoperability paradigms provide a solid foundation for the “Unified Field Theory.” The ArB believes that the rigorous adoption of analysis-level artifacts as part of the SAIF will facilitate the adoption of HL7 standards, promoting reuse and contextualizing much of the important analysis that often lies behind a given standard.

Note: As implied in the above figure, it is possible to choose one or more interoperability paradigms (services, documents, or messages) from the analysis specification. This choice does not destroy the ability to integrate as organizations can still collaborate through mutual adoption of a given analysis specification, with appropriate transformations and adapters written to insure appropriate collaboration. In addition, differing degrees of CSI may be applicable (for example, human vs. computational semantic interoperability vs. syntactic interoperability).

### 1.3.5.1.3.3. Informational Systems Example

In particular, the Clinical Document Architecture (CDA)-based notion of “less-informational and highly-informational systems” has a more formal framework that can be deployed outside the domain of CDA.

The following figure shows an example of highly informational systems versus less informational systems.

*Figure 15. Highly Informational Systems and Less Informational Systems*
1.3.5.1.4. HL7 Work Groups

HL7 has undergone a transformation that aligns well with the SAIF. The responsibility for Information viewpoint components falls to the following domain groups:

- Construct Domain Information Model (DIMs)
- Constrained Information Model (CIMs)
- Localized Information Model (LIMs)

The responsibility for architecture, design, and technical issues falls to the Foundations Steering Division (SD).

However, the focus on services as first-class, business-linked services, (rather than simple imports of web service-related standards into HL7) imply the need for groups within the Foundations Steering Division that can define, manage, and evolve the infrastructure for services, as well as realigning several components with other interoperability paradigms. Tasks of this group (which also could be the ArB) include development of a SAIF-compliant normative standard for service specifications in general, with particular focus on a normative specification for contracts and services roles.

Related information

- Behavioral Framework

1.3.5.1.4.1. SAIF and HSSP

Following discussions with HSSP representatives, the ArB affirms that the HSSP framework is in conceptual alignment with the HL7 SAIF for both processes and artifacts.

In particular, the Model-Driven Architecture (MDA)-based process, such as the HSSP Service Specification Framework, produces a Service Functional Model, a Platform-Independent Model, and a Platform-Specific model that are, in principle, in alignment with the HL7 SAIF. Note that this alignment is between the overarching HSSP process and artifacts, which by definition include at least two participating organizations (for example, HL7 and OMG), and HL7 as the sole producer of SAIF-compliant processes and artifacts.

This distinction is important because it is likely that the SAIF framework will result in processes...
and artifacts produced completely within HL7, which are not necessarily defined in the HSSP, thereby resulting in some degree of non-alignment. As the SAIF artifacts and processes mature, an open question remains about how (or if) any variations between the SAIF and the HSSP are to be addressed.

### 1.3.5.1.4.2. Health Services Specification Project and SAIF

Initial work on SAIF was informed by the experience of several key organizations, for example, the National Cancer Institute and Health Services Specification Project (HSSP).

#### 1.3.5.1.4.2.1. HSSP Service Specification Framework

This table gives an overview of the main activities within the HSSP Service Specification Framework (SFF), which is the process for producing a service interface specification.

**Note:** The process is created specifically for producing standard specifications for service interfaces. This process is NOT for the internal implementation of services OR the process or collaborations in which they may take part, although the Service Functional Model (SFM) includes sample, non-normative descriptions of processes and interactions that may use the services.

**Table 1. HSSP Process (Service Specification Framework)**

<table>
<thead>
<tr>
<th>HSSP Process</th>
<th>HSSP Artifact</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Define Service Functional Model (SFM)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Set up project (any HL7 work group)</td>
<td>Acceptance for SOA work group and HSSP Support (informal)</td>
<td>Specifies an informal process by which the work group agrees to follow the HSSP process and use the latest Service Functional Model (SFM) template.</td>
</tr>
<tr>
<td>Produce SFM (any HL7 work group)</td>
<td>Service Functional Model (SFM)</td>
<td>The work group produces the SFM and usually sets up weekly conference calls. No formal process exists for this phase.</td>
</tr>
<tr>
<td></td>
<td>• Service scope</td>
<td>Specifies the statement of scope.</td>
</tr>
<tr>
<td></td>
<td>• Service business case</td>
<td>Specifies textual business background and context for the Service.</td>
</tr>
<tr>
<td></td>
<td>• Business process</td>
<td>Specifies descriptions of scenarios in which the service may be used. These are examples because services are intended to be used in many different scenarios. These are storyboards but they also can use more formal mechanisms,</td>
</tr>
<tr>
<td>HSSP Process</td>
<td>HSSP Artifact</td>
<td>Description</td>
</tr>
<tr>
<td>--------------</td>
<td>--------------</td>
<td>-------------</td>
</tr>
<tr>
<td></td>
<td>• Functions and capability descriptions</td>
<td>Lists functions, with preconditions, postconditions, inputs, outputs, errors, all expressed in conceptual and business terms.</td>
</tr>
<tr>
<td></td>
<td>• Conformance profile</td>
<td>This set of functional and semantic profiles is combined to give a complete, coherent set of capabilities against which conformance can be claimed. The profile should be versioned.</td>
</tr>
<tr>
<td></td>
<td>• Functional profile</td>
<td>Specifies a named list of a subset of the operations defined within this specification that must be supported to claim conformance to the profile.</td>
</tr>
<tr>
<td></td>
<td>• Semantic profile</td>
<td>Identifies a named set of information descriptions (semantic signifiers) that are supported by one or more operations. For HL7-based profiles, they provide cross-references to the appropriate RIM-based domain models. Services usually do not create their own new information models, but will cross-reference to existing Construct Domain Information Models (DIMs), Common Message Element Types (CMETs), and so on. In many cases, the services may be general purpose and may consume multiple different models by using semantic profiles.</td>
</tr>
<tr>
<td></td>
<td>• User scenario details</td>
<td>Specifies an optional section that includes dynamic model information. An example would be sequence or collaboration diagrams that demonstrate using the service functions in the previously identified business scenarios.</td>
</tr>
<tr>
<td></td>
<td>• Technical assumptions and recommendations</td>
<td>Provides guidance and constraints on the following technical specifications, which should be based on business requirements and needs.</td>
</tr>
<tr>
<td>Ballot SFM in HL7</td>
<td></td>
<td>The usual approach is to ballot as a DSTU to allow for time for the</td>
</tr>
<tr>
<td>HSSP Process</td>
<td>HSSP Artifact</td>
<td>Description</td>
</tr>
<tr>
<td>----------------------------------</td>
<td>--------------------------------</td>
<td>--------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>technical specification to be produced and for the lessons learned fed back into the normative version of the SFM.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2. Produce request for proposal (RFP)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Produce RFP (OMG Healthcare Domain Task Force (DTF))</td>
<td>RFP for technical specification</td>
<td>This translation is of the information from the HL7 SFM into an OMG RFP. This step is fairly simple and mechanical, although it does allow some decisions on prioritization of features to be carried out. (This allows for specification of mandatory and optional Requirements, as well as discussion items and evaluation criteria and preferences).</td>
</tr>
<tr>
<td>Produce LOIs</td>
<td></td>
<td>Potential responders to the RFP produce Letters of Intent, which are formal documents requiring signature of a senior company official.</td>
</tr>
<tr>
<td>3. Produce technical specification</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Produce technical specification (OMG RFP responders)</td>
<td></td>
<td>Normally carried out in several stages, that is, one or more initial drafts are produced by individual or groups of vendors who submitted LOIs. Either from the beginning, or after initial drafts, the vendors form a joint submission team. Note that acceptance requires that at least one commercially viable implementation be available at the time of ratification of the standard.</td>
</tr>
<tr>
<td>Tech spec -- Platform Independent Model (PIM)</td>
<td></td>
<td>Specifies a technology neutral, but fully-specified interface specification, usually in Unified Modeling Language (UML). Includes:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Interfaces</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Operations</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Signatures (inputs, outputs)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Exception conditions</td>
</tr>
<tr>
<td>HSSP Process</td>
<td>HSSP Artifact</td>
<td>Description</td>
</tr>
<tr>
<td>---------------</td>
<td>---------------</td>
<td>-------------</td>
</tr>
</tbody>
</table>
| Tech spec -- Platform Specific Model (PSM) | Specifies at least one technology-specific specification of the interface (as requested by the RFP), usually Web Services, at a minimum. Includes:  
  • Interfaces  
  • Operations  
  • Signatures (inputs, outputs)  
  • Exception conditions | |
| Additional description | The RFP response also includes a set of commentary by the responders relating to the requirements and evaluation criteria stated in the RFP. | |

Ballot technical specification

Voted on by Healthcare DTF, then OMG Architecture Board and then the OMG Technology Committee. Any OMG Members can sign up to vote, it is one company one vote, unlike HL7 ballots.

Finalization

Once approved by the vote, a finalization process occurs under the auspices of the OMG Architecture Board and Technical Committee that finalize any outstanding issues and ensure that the implementation is available.

### 1.3.5.1.4.2.2. Mapping from HSSP Process and Artifacts to SAIF

This table shows the one-way mapping from the HSSP process and artifacts to the HL7 SAIF. It does not map the other way, or consider additional steps or artifacts.

<table>
<thead>
<tr>
<th>HSSP Process</th>
<th>HSSP Artifact</th>
<th>HL7 SAIF Process</th>
<th>HL7 SAIF Artifact</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Define SFM</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Set up project (any HL7 work group)</td>
<td>Acceptance for SOA work group and HSSP Support (informal)</td>
<td>Refer to HL7 Development Framework (HDF) -- formal project</td>
<td></td>
<td></td>
</tr>
<tr>
<td>HSSP Process</td>
<td>HSSP Artifact</td>
<td>HL7 SAIF Process</td>
<td>HL7 SAIF Artifact</td>
<td>Comments</td>
</tr>
<tr>
<td>--------------</td>
<td>---------------</td>
<td>------------------</td>
<td>-------------------</td>
<td>----------</td>
</tr>
<tr>
<td>Produce SFM (any HL7 work group)</td>
<td><strong>Service Functional Model (SFM)</strong></td>
<td>Service role specification (analysis and blueprint)</td>
<td>Equivalent. (SAIF more formally defines dynamic model but SFM is more explicit on business context and narrative.)</td>
<td></td>
</tr>
<tr>
<td><strong>• Service scope</strong></td>
<td>Role type description</td>
<td>Not explicitly defined in SAIF.</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>• Service business case</strong></td>
<td>Role type description</td>
<td>Not explicitly defined in SAIF.</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>• Business process</strong></td>
<td>Storyboard</td>
<td>Equivalent.</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>• Functions and capability descriptions</strong></td>
<td>Behavior type</td>
<td>Equivalent.</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>• Conformance profile</strong></td>
<td>Unknown</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>• Functional profile</strong></td>
<td>Behavior profile type</td>
<td>Equivalent.</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>• Semantic profile</strong></td>
<td>Static Information Model Reference</td>
<td>Equivalent.</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>• User scenario details</strong></td>
<td>Collaboration, interaction, activity, exchange</td>
<td>SEAEAF defines a more formal dynamic model structure.</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>• Technical assumptions and recommendations</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Ballot SFM in HL7</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Ballot analysis specification in HL7</strong></td>
<td></td>
<td>Not sure of intent, but balloting SFMs as they stand seems within the spirit of SAIF, possibly with more formalization of dynamic model at the discretion of the work groups.</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>2. Produce request for proposal (RFP)</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
### HL7 Services-Aware Interoperability Framework (SAIF)

<table>
<thead>
<tr>
<th>HSSP Process</th>
<th>HSSP Artifact</th>
<th>HL7 SAIF Process</th>
<th>HL7 SAIF Artifact</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>Produce RFP (OMG Healthcare DTF)</td>
<td>RFP for technical specification</td>
<td>Not equivalent. The SAIF would be used when the HL7 workgroup continues the process itself.</td>
<td>N/A</td>
<td></td>
</tr>
<tr>
<td>Produce LOIs</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td></td>
</tr>
<tr>
<td><strong>3. Produce technical specification</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Produce technical specification (OMG RFP responders)</td>
<td></td>
<td>Produce conceptual design and implementation design.</td>
<td></td>
<td>Process and governance are very different, basic outputs are similar.</td>
</tr>
<tr>
<td>Technical Specification -- Platform Independent Model (PIM)</td>
<td></td>
<td>Service role specification (conceptual design)</td>
<td></td>
<td>Equivalent. HSSP does not formally define the collaborations and dynamic model.</td>
</tr>
<tr>
<td>Technical Specification -- Platform Specific Model (PSM)</td>
<td></td>
<td>Service role specification (implementation)</td>
<td></td>
<td>Equivalent. HSSP does not formally define the collaborations and dynamic model.</td>
</tr>
<tr>
<td>Additional description</td>
<td>N/A</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Ballot technical Specification</td>
<td>HL7 ballot</td>
<td></td>
<td></td>
<td>Similar, although the voting rules differ.</td>
</tr>
<tr>
<td>Finalization</td>
<td>HL7 ballot reconciliation</td>
<td></td>
<td></td>
<td>OMG process provides an additional level of architecture validation. Not sure if we have anything equivalent in HL7.</td>
</tr>
</tbody>
</table>

### 1.3.5.1.4.3. SAIF and HSSP: Service Functional Model

The Service Functional Model (SFM), defined by the HSSP is a requirements specification for a service.
It defines the service from a business perspective, including its basic operations and functional profiles, contextual applications of the services that use context-appropriate subsets of the service’s available operations. For each functional profile, the SFM specifies the pre- and post-conditions, as well as the semantic profiles that define the static data that will be exchanged during an interaction between a provider of the service and its client. In particular, SFM semantic profiles are expected to be drawn from the reservoir of healthcare static semantic standards; for example, the RIM, BRIDG, and terminologies, and so on (in particular). Thus, the HSSP SFM is a conceptual semantic artifact and not a platform-independent (analysis or design) artifact. As such, it maps easily to the Computationally Independent Model (CIM) layer of the SAIF in general and the SAIF ECCF in particular, providing information for the Enterprise/Business, Informational, and Computational viewpoints.

1.3.5.1.4.4. SAIF and HSSP: Process Compatibility

Considerable overall process compatibility exists between SAIF and HSSP, at the conceptual level. In addition to the fact that HSSP is restricted to service specifications only, whereas SAIF applies to specifications of messages and documents as well as services, differences exist in the organizational boundaries of the two approaches. In particular, SAIF places the complete repertoire of artifact generation and associated processes under the aegis of HL7, whereas HSSP splits the responsibilities between HL7 and OMG. HSSP has the definition of the Platform-Independent (logical) and Platform-Specific (physical) artifacts occur within OMG, while SAIF provides a framework for these to be fully developed by HL7. In addition, the HSSP process is less prescriptive in its specification of artifacts than SAIF. The ArB believes that this level of prescriptiveness is essential if HL7 is to achieve scalable, loosely-coupled WI across each of its three Interoperability Paradigms.

Note that recent conversations with OMG leadership have indicated a strong interest in aligning HSSP and SAIF at a more detailed artifact and process level, an activity that is strongly supported by both the HL7 ArB and TSC. It also is important to remember that SAIF does not require that all SAIF-compliant artifacts be developed within HL7 or that HL7 work groups execute all SAIF-compliant processes. SAIF is primarily a framework for achieving Working Interoperability. SAIF specifies the component Frameworks – Behavioral, Conformance and Compliance, and Governance – which enable that goal to be achieved, regardless of the organizational topography.

1.3.5.2. Appendix B: Relationship with Other Standards Development Organizations

As the acknowledged leader in data and information exchange in the healthcare domain – a domain that includes regulatory, clinical, preclinical, and research interests, and stakeholders – HL7 has fundamental responsibilities to and relationships with other Standards Development Organizations (SDOs) and standards organizations. These organizations include ANSI SDO, World Wide Web Consortium (W3C), and Organization for the Advancement of Structured Information (OASIS). Some of these responsibilities have been formalized in various ways (for example, MoUs), while others are implicit through the adoption by HL7 of non-HL7 work products. As these relationships influence the enterprise architecture of HL7, they should be made explicit and named as dependencies on the development or implementation of a given HL7 specification or SDO standard. This work can be expected to occur within existing and newly-created work groups, or as the result of reorganization of existing work group boundaries and responsibilities. The scope of these redefined boundaries and responsibilities have an impact not only within HL7, but also extend to impact external organizations with whom HL7 interacts.

Related information

► Section 1.5. Governance
► Section 1.3.5.1.4. HL7 Work Groups

1.3.5.2.1. Object Management Group
Of particular relevance to HL7 regarding the HL7 SAIF is the Object Management Group (OMG), the owner of Unified Modeling Language (UML), Common Object Request Broker Architecture (CORBA), and a host of other standards, including the existing partnership between HL7 and OMG around the Health Services Specification Project (HSSP).

This pioneering work was, in part, responsible for the current SAIF effort. Ideally, SAIF should increase the value -- and possibly speed up the implementation rate as well -- of the current agreement. Hopefully, the analysis models (or their equivalents) -- which contain more robust semantics than the currently specified requirements-based Service Specification Model that is defined under the original HSSP process -- will be able to serve as input to the OMG’s request for the proposal (RFP) process. (Currently, the content of these analysis documents are part of the RFP process rather than input into it.) In summary, the HL7/OMG relationship will continue to be an important and effective pathway for HL7-based service specification and implementation that should be encouraged and strengthened as a viable way for HL7 service specifications to gain traction in the healthcare, life sciences, and clinical research domain.

### 1.3.5.2.1.1. Influence of Existing OMG Specifications on SAIF

Several OMG standards have influenced HL7 and recent SAIF work.

The following is a description of these OMG standards:

- **Model Driven Architecture (MDA)** – MDA provides a framework for constraining models from more abstract to more concrete models. MDA is at the heart of the HL7 service specification process (for example, CFM, Platform-Independent Model (PIM), and Platform-Specific Model (PSM). Ultimately, HL7 should expect its standards to be able to take advantage of rigorous transformations between levels of the models, including, where appropriate, automated tool support and, when appropriate, code generation.

- **SOA Pro and UML Profile and Metamodel for Services (UPMS)** – The SOA Pro project and the UPMS has informed the ArB metamodel for services, conformance, and the Behavioral Framework. Of particular interest is the notion of a collaboration between partners. As these specifications mature, they will become more tightly integrated into SAIF notation and artifact generation.

- **Business Process Definition Metamodel (BPDM)** - Efforts are underway within OMG to harmonize between the various orchestration and choreography languages that exist in the industry. Examples include Business Process Execution Language (BPEL) and Web Services Choreography Description Language (WS-CDL). This effort has informed the development of SAIF by looking at the trajectory of these dynamic model expression languages in the industry.

The following is a definition of BPDM from Wikipedia:

> The Business Process Definition Metamodel (BPDM) is a standard definition of concepts used to express business process models (a metamodel), adopted by the OMG (Object Management Group). Metamodels define concepts, relationships, and semantics for exchange of user models between different modeling tools. The exchange format is defined by XSD (XML Schema) and XMI (XML for Metadata Interchange), a specification for transformation of OMG metamodels to XML. Pursuant to the OMG’s policies, the metamodel is the result of an open process involving submissions by member organizations, following a Request for Proposal (RFP) issued in 2003. BPDM was adopted in initial form in July 2007, and finalized in July 2008.

- **Business Process Modeling Notation (BPMN)** – BPMN is of interest to HL7 as a potential candidate for capturing requirements and storyboards in a rigorous, model-driven way. Some question exists as to whether the notation adequately captures the semantics of interactions, roles, and responsibilities, but insofar as BPMN (and BPMN2) informs BPDM, this essential work is emerging.

- **SOA ML** – See the Behavioral Framework information for an analysis.
1.3.5.2.2. Component Based Development and Integration SOA Consultancy

Component Based Development and Integration (CBDI) has been a primary force in the SOA community for the last few years.

The CBDI Meta-Model for SOA has informed and validated many of the SAIF concepts, as well as driving the discussion on a classification system for services.

1.3.5.2.3. World Wide Web Consortium

The World Wide Web Consortium (W3C) represents a major source of standards emerging from the industry. Many of these standards should be incorporated into HL7 specifications, both as dependencies and as contextualized references within specifications.

The following standards from the W3C informed the SAIF work:

1. **WS-* standards** – These wide-ranging groups of specifications provide for several pieces of infrastructure that are related to message reliability that serve as a foundation for many Implementable Design Specifications to speak to the need (or lack thereof) for acknowledgements, as part of a given specification.

2. **Web Services Choreography Description Language (WS-CDL)** – The Choreography Description Language heavily informed the Behavioral Framework in terms of semantics captured and in the coupling of these things.

At the Orlando meeting, the senior representatives from OMG, HL7, and W3C’s Healthcare Life Sciences Task Force made a formal commitment to collaborate on enterprise-architecture issues related to healthcare which are of mutual interest; for example, ODM, XMI, data types, and so on.

1.3.5.2.4. Organization for Advancement of Structured Information Standards

OASIS is a not-for-profit consortium that drives the development, convergence, and adoption of open standards for the global information society. The OASIS SOA Reference Architecture, Version 1 informed SAIF in terms of architectural views, service-related principles, conceptual definitions, and relationships between parties.

**Note:** While the OASIS Reference Architecture has its own specific definitions of certain notions (such as *service*), these notions served as a starting point rather than the authority. The ArB plans to continue to monitor the OASIS normative SOA architecture to inform SAIF from both a services-specific and services-aware perspective.

1.3.5.2.5. Joint Interoperability Council

The Joint Interoperability Council (JIC) consists of representatives from HL7, ISO, European Committee for Standardization (CEN), and CDISC.

The group was formed to coordinate the standards activities of the four organizations to ensure that redundant, potentially incompatible work was kept to a minimum, as well as to harmonize existing work whenever possible. To date, a major product-of-value from the JIC is the ISO Data Types Specification for Health Data Types (ISO 21090), a specification which incorporated much of the work of HL7 and its Abstract Data Type Specification (R2) into a standard that is more implementation-friendly than the ADT R2 specification was, and is semantically isomorphic with the ADT R2 specification.

Pillar 2 of the CSI Four Pillars specifies the need for a computationally robust data type standard and 21090 provides that standard. All data-type-aware SAIF artifacts are expected to be bound to this important standard.
Moving forward, HL7 anticipates informing much of the JIC dialogue around SAIF and its processes, artifacts, and value.

### 1.3.5.2.6. National Institute of Standards and Technology

The National Institute of Standards and Technology (NIST) is testing HL7 standards and using conformance statements to verify technology bindings and implementations.

Additionally, NIST has provided a mirrored registry for HL7 artifacts and, as such acts as a model repository for HL7 V3 artifacts. HL7 anticipates introducing the ECCF into NIST-based HL7 conformance and compliance testing.

### 1.3.5.2.7. Tooling and Open Health Tools Initiative

The Open Health Tools (OHT) Architecture Project voted to use the SAIF ECCF as a core component to specifying the artifacts that are required for tool developers wanting to provide tools for the four core member organizations.

#### 1.3.5.2.7.1. Tooling Requirements

The continuing advancement in the application of standards in software engineering, integration, and deployment has been predicated on the ability of end user requirements that are expressed using tools that produce highly specific, structured artifacts that can be input to software design tools and from there into interoperability configuration and testing tools.

The less reliance on manual intervention and the greater degree of automated transformation to meet the different areas of concern, the more flexible and durable the entire environment becomes.

Traceability from requirements through to deployed software capability is one essential quality criteria, as are formalized test cases and integration testing. The more dependent the success of an organization is on the software deployed to support its mission, the more formal this process needs to become. Specifically, it is essential that tooling reflect an architectural framework rather than defining one.

We are entering an era in which tooling supports the specification of information exchange standards and all the functions necessary to deploy standards-based automated components, whether the interoperability channel chosen is messaging, documents, or services. The emerging trend is toward an open-source tooling environment. A notable example can be found in the Open Heath Tools initiative and its associated Architecture Project team, whose representatives include tool developers and potential tool customers from the UK National Program, Australia NeHTA, Canada Infoway, and the National Cancer Institute. The OHT focuses on providing open-source software to support the building of software components that enable Working Interoperability. At a recent meeting, the AP team agreed on the critical importance of architecture in achieving standards-based interoperability and, to this end, adopted the following charter statement:

To enable the adoption, development, and deployment of an evolving set of interoperable tools. These tools support the development and deployment of software that enables computable semantic interoperability in the health-care, life sciences, and clinical research domains. Tools are defined as any software component that is not a clinical end-user application, although such software components may form part of an end user application. Tools are intended to be useful and usable for their intended purpose and to interoperate with each other.

#### 1.3.5.2.7.2. OHT Architecture Project

The OHT Architecture Project operates under the Open Health Tools Development Policy and Process and is chartered by the Board to articulate and adopt a formal architecture framework, architecture principles and best
practices that are focused on relevant interoperability and the use of standards.

As its initial goal, the Architecture Project is developing an architecture framework that will itself enable the evolutionrary development of an OHT Platform architecture consistent with the various enterprise architectures built and deployed by OHT stakeholders.

Deliverables include:

- A set of architecture principles
- A tool classification mechanism that enable stakeholders to contribute and have access to architecture artifacts that underlie the tools
- A set of templates, patterns, and processes that enable interoperability of OHT tools
- Identification of potential barriers to interoperability and recommendations to overcome them
- A recommendation to the board of a governance process to assist in the harvesting, categorization, and custodianship of architecture artifacts
- An executed internal and external communication plan

Related information

► Open Health Tools Development Policy and Process

1.3.5.3. Appendix C: A Brief History of SAIF

This section describes how SAIF got started, the Jump Start project, and the history of services in HL7.

1.3.5.3.1. How SAIF Got Started

In January 2008, the HL7 CTO directed the HL7 Architecture Board (ArB) to develop a robust Enterprise Architecture Specification (EAS) for HL7.

The CTO's request to the ArB was driven by the HL7 Board of Directors' commitment to the following three-part premise:

- HL7 produces specifications to enable Computable Semantic Interoperability (CSI) between users of systems implementing those Specifications.
- Instances of CSI between two or more HL7-based systems may cross department, enterprise, and/or national boundaries.
- An HL7 EAS is required if HL7 is to produce durable specifications that enable CSI in an effective, efficient, and scalable manner.

This approach is based on the ArB’s overarching belief that an EAS must emerge from both bottom-up and middle-out efforts rather than from a top-down specification.

Related information

► Section 1.3.5.3.2. Jump Start Project
► Section 1.3.5.1.1. SAIF Alpha Project Implementation

1.3.5.3.1.1. May 2008 Working Group Meeting
Subsequently, during the May 2008 Working Group Meeting, the CTO asked the ArB to provide a draft architectural framework describing how core concepts from Service-Oriented Architecture (SOA)--also called a "services-aware" perspective -- could be integrated into the core EAS. This architecture framework would focus specifically on how SOA would facilitate the core specification processes of HL7 messages, documents, and services. These two requests (for an HL7 EAS and that the EAS identify ways in which the existing HL7 specification processes might be improved in the context of an EAS) were proposed as the first pieces of the larger strategic vision and architecture trajectory for HL7.

1.3.5.3.1.2. Summer 2008 Jump Start Meetings

In response to this charge, the ArB launched the Jump Start project over the summer of 2008, holding three 3-day meetings in June, July, and August. The HL7 HQ announced each of the meetings in advance and held these meetings in Rockville, MD. The spirit of the "Left Side of the RIM" meetings (1998) inspired the concept and flow of the meetings. The series of multi-day face-to-face meetings allowed members of the ArB -- as well as other HL7 members interested in contributing to the effort--to dedicate some concentrated time to consider in depth a single topic: The initial definition of and content required to produce the first components of a draft Enterprise Architecture Specification (EAS) for HL7.

1.3.5.3.1.3. September 2008 Working Group Meeting

The CTO asked that a first draft of the EAS be delivered at the Vancouver Working Group meeting of September 2008, a request which the ArB successfully fulfilled with the draft being entitled "A Services-Aware Interoperability Framework for HL7 (SAIF)."

At the Vancouver meeting, following a review of the draft by several committees and the HL7 Technical Steering Committee (TSC), the TSC made a decision to adopt formally the draft SAIF specification, and to launch an "enterprise architecture roll-out project." As part of that effort, the ArB committed to complete the still-unfinished portions of the draft SAIF document, and to develop a companion set of education presentations. The ArB is using these presentations to help both HL7 membership and interested external parties understand better the implications and adoption challenges of SAIF structure and processes from both a short- and longer-term perspective.

1.3.5.3.2. Jump Start Project

The goal of the Jump Start project was to produce a draft Enterprise Architecture Specification (EAS) for HL7.

1.3.5.3.2.1. Jump Start Participants

The participants in the Jump Start Project collectively brought a wide range of perspectives and expertise to the discussion and effort.

<table>
<thead>
<tr>
<th>HL7 ArB:</th>
</tr>
</thead>
<tbody>
<tr>
<td>Yongjian Bao, GE Healthcare</td>
</tr>
<tr>
<td>Jane Curry, Health Information Strategies</td>
</tr>
<tr>
<td>Grahame Grieve, Jiva Medical</td>
</tr>
<tr>
<td>Anthony Julian, Mayo</td>
</tr>
<tr>
<td>John Koisch, National Cancer Institute (NCI)</td>
</tr>
<tr>
<td>Cecil Lynch, OntoReason</td>
</tr>
<tr>
<td>Charlie Mead, CTO NCI</td>
</tr>
<tr>
<td>Nancy Orvis, Department of Defense (DoD) MHS</td>
</tr>
<tr>
<td>Ron Parker, Canada Health Infoway</td>
</tr>
<tr>
<td>John Quinn, Accenture, CTO HL7</td>
</tr>
</tbody>
</table>
1.3.5.3.2.2. Jump Start Deliverables

On the first day of the first Jump Start meeting, the ArB, along with input from the CTO, listed the primary deliverables for the project.

These deliverables answered the overarching question, "How should services in particular--and, in general, other instances of interoperability paradigm artifacts; for example, messages and documents--be defined, specified, implemented, and governed within HL7?"

The ArB determined that it could best fulfill its charge for the Jump Start sessions by producing, at a minimum, the following deliverables:

- An EAS solution for the SAIF
- An inventory of current HL7 artifacts that is suitable for migration into an evolving EAS
- An inventory of critical, but currently missing, components that an EAS needs. The ArB identified these three components as being the primary focus of the Jump Start project:
  - A Behavioral Framework to express interaction semantics.
  - A layered Conformance and Compliance Framework to support service integration and runtime assessment of Computable Semantic Interoperability (CSI).
  - A Governance Framework to oversee the development and implementation of specifications for services and other HL7 Interoperability Paradigms.
- A proposal for implementing the SAIF
- The implications of adopting an EAS using SAIF to HL7, its customers, and its partners include:
  - Organizational impact within HL7 and between HL7 and other organizations
  - Tooling impact

On the final day of the third Jump Start meeting in August 2008, the CTO and the ArB reached unanimous consensus that they had been successful in delivering a framework that contained a draft of the critical, but missing, components of an EAS for HL7. In addition, the consensus was that most (if not all) of the work done during the Jump Start meetings was indeed consistent with a "Unified Field Theory of HL7 Interoperability."

Related information

- Section 1.3.4.2.7.4. Legacy Artifacts, Interoperability Paradigms, and the Unified Field Theory

1.3.5.3.2.3. Services-Aware Approach for SAIF
The CTO asked the HL7 ArB to develop an Enterprise Architecture Framework that was not specifically HL7 developing a Services-Oriented Architecture (SOA).

Rather, the ArB focused on the guiding principles from SOA that could be extended to each of HL7's three Interoperability Paradigms, such as messages, documents, and services.

SAIF is services-aware, but not services-specific. SAIF is a precondition for a well-designed Services-Oriented Architecture (SOA). SAIF is a component of SOA.

The SAIF draws on service-aware principles, such as:

- Contracts, roles, and collaborations (behavior)
- Conformance and compliance (ECCF)
- Governance

### 1.3.5.3.2.4. Reuse of Existing HL7 Artifacts

SAIF uses and reuses existing HL7 artifacts.

The artifacts include:

- Reference Information Model (RIM)
- Data types
- Clinical Document Architecture (CDA)
- Refined Message Information Models (RMIMs)
- Vocabulary
- Electronic Health Record System Functional Model (EHRS-FM)

### 1.3.5.3.2.5. Caveats and Acknowledgments

SAIF is the first step on the road to defining an Enterprise Architecture Specification (EAS) for HL7. The ArB is continuing to work on producing a draft Enterprise Architecture (EA).

Note: The CTO and the ArB acknowledge that the SAIF and the associated effort involved in defining and implementing an Enterprise Architecture Specification (EAS) for HL7, as a prerequisite for realizing CSI for HL7 customers on the scale to which HL7 is committed is neither simple nor easy.

In response to that acknowledgment, the SAIF is meant to be "industrial-strength." SAIF is designed not only to craft software components -- messages, computable documents, services, and applications -- that are usable in the healthcare, life sciences, and clinical research domains, but also to facilitate and enable efficient lines of communications between the various cross-functional teams. These teams are creating fully specified interoperable components that can be integrated and targeted to fulfill specific business needs for HL7 and other healthcare organizations.

The CTO and Chair of the ArB would like to thank all those persons and organizations who have contributed to this initial effort in defining the SAIF. In particular, the SAIF received specific contributions from the Health Services Specification Project (HSSP) and HL7 Service-Oriented Architecture (SOA) Technical Committee, the National Cancer Institute (NCI) and its Center for Biomedical Informatics and Information Technology, the Department of Defense, Canada Health Infoway, IBM, Siemens, and Bernd G.M.E. Blobel. Most important, however, was the hard work and dedication that the various contributors to the Jump Start effort showed -- each of whom sacrificed valuable summer time and energy to provide their own experience and expertise to the SAIF document and
1.3.5.3.3. History of services in HL7

This section explains how HL7 is building on the IT industry’s ideas about services. HL7’s customers asked HL7 to define services specifications.

HL7 technically became services-aware in 2004 with the development of a web services “wrapper” for V2 and V3 messages through a formal HL7 Web Service Profile, as well as through a series of discussions with the Electronic Business using eXtensible Markup Language (ebXML) task force. True services-awareness within HL7 is due, in large part, to the work done over the past several years of the small group of people within HL7 who formed what was initially a “Birds of a Feather” group to talk about how service-centric work could become part of the HL7 dialogue. In 2005, members of this group formed the Health Services Specification Project (HSSP) through a collaborative Memorandum of Understanding (MoU) with the Object Management Group (OMG). The HL7 members participating in the HSSP were officially recognized within HL7 as the SOA Special Interest Group (SIG), a group whose activities soon led them to be promoted to the status of SOA Technical Committee. This group is now the SOA Work Group and continues to work within HL7 to collaborate with OMG via HSSP activities.

The HSSP’s MoU with the OMG specified several artifacts and processes, some of which the HL7 branch of HSSP generated, and some of which the OMG side of the relationship developed. The overarching structure of the MoU envisioned that, in general, HL7 would provide the domain-specific healthcare requirements, while OMG would use its established mechanisms for request for proposals (RFPs) for technical implementations of candidate OMG standards to provide the artifacts and processes that were necessary to move a service’s requirements specification to a final implementation.

Historically, the group has had a close affiliation with the Electronic Health Record (EHR) Technical Committee. It produced – and successfully balloted as Draft Standard for Trial Use (DSTUs) – several Service Functional Model (SFM) specifications (service requirements documents), including the Entity Identification Service, the Resource Location and Update Service, and the Decision Support Service. In 2008, a Service Functional Model for the Clinical Research Filtered Query (CRFQ) service was developed and successfully balloted in collaboration with the Regulated Clinical Research Information Management (RCRIM) Technical Committee. Several other service specifications are in the process of being developed.

1.3.6. SAIF Project Summary

The following is a summary of the SAIF effort as of March 2010.

- The HL7 Services-Aware Interoperability Framework identifies and defines the language, processes, and artifacts for:
  - Developing an Enterprise Architecture appropriate for an Standards Developing Organization (SDO) that is using services or services-aware thinking as a backbone concept.
  - Developing and deploying standards to enable Working Interoperability.
- Services (and SOA) are not technology per se. Rather, they are a framework for approaching how to design distributed capabilities (information and functionality sharing).
- Services are not equivalent to Web Services (although service constructs and other interoperability paradigms, may be realized using Web Services technology.
- SAIF is a precondition for a well-designed Service-Oriented Architecture (SOA). SAIF is a component of SOA.
- SAIF is “services-aware” and therefore meant to apply to all of HL7’s interoperability paradigms (messages, documents, and services) within the context of the goal of Working Interoperability.
HL7 Services-Aware Interoperability Framework (SAIF)

- SAIF consists of four sub-frameworks:
  - Behavioral Framework
  - Enterprise Conformance and Compliance
  - Governance
  - Information Framework

- SAIF is not a replacement for, or an alternative to HL7’s existing products, engagements, or offerings.

- SAIF is an effort to reframe, encompass, and support existing HL7 work streams, and to focus them around a more explicit framework for expressing interoperability semantics.

- SAIF is the result of a shared belief by the HL7 CTO, ArB, and TSC that the Health Enterprise requires this level of specificity and rigor to achieve scalable, agile interoperability. Given HL7’s historical leadership in defining healthcare standards, it is appropriate that HL7 is establishing a leadership position in the dialogue about healthcare enterprise architecture.

- SAIF defines specific artifacts and a formal constraint pattern that provide traceability from requirements through analysis to design and implementation.

- SAIF-compliant specifications align with conformance levels, as specified in the ECCF, which enable HL7 consumers to adopt HL7 specifications in different contexts at different levels of interoperability.

SAIF represents a synthesis of best practices and concepts from multiple frameworks, and was developed through several "lenses," as the following figure shows.

Figure 16. The Lens of SAIF.

In the six months following the September 2008 HL7 Working Group meeting in Vancouver, there has been considerable dialogue outside of HL7 regarding the possibility of other organizations adopting the basic components of SAIF. In particular, the HL7 SAIF aligns with strategic direction of several national health-centric organizations including:
In addition, HL7 SAIF aligns with other industry standards, including WS-CDL, SOA Pro, and Health Information Systems Architecture (HISA).

**Related information**

- Section 1.3.4.2. Lens of SAIF

### 1.4. Enterprise Conformance and Compliance Framework

This document discusses in detail the motivation for, as well as the structure, content, and use of the Enterprise Conformance and Compliance Framework (ECCF).

As discussed in *SAIF Components*, the Services-Aware Interoperability Framework (SAIF) consists of four core sub-frameworks:

- Information Framework (IF)
- Behavioral Framework (BF)
- Enterprise Conformance and Compliance Framework (ECCF)
- Governance Framework (GF)

#### 1.4.1. ECCF Goals

The major goal of the Enterprise Conformance and Compliance Framework (ECCF) is enabling Working Interoperability between different users, organizations, and systems. The ECCF is manifest in a structure called the *ECCF specification stack*.

The ECCF specification stack (SS) identifies, defines, organizes, and relates a set of artifacts that collectively specify the relevant semantics of a software component specification or other system-of-interest.

The ECCF SS provides an organizational framework in which inter-related artifacts are sorted by content – for example, business rules, information constructors, behavioral contracts, and level-of-abstraction. For example, Standards Developing Organizations (SDOs) can use external standards and specifications in the ECCF specification stack.

**Note to reviewers:** I’ve included just the ECCF Overview here. For the entire ECCF document, see the Enterprise_Conformance_Compliance_Framework_peer_review_20100312.pdf document.

#### 1.5. Governance

The Governance Framework (GF) oversees the development and implementation of specifications for services and other HL7 Interoperability Paradigms.

The following is a high-level summary of the Governance Framework (GF):

- Internal HL7 governance
- Relationships between HL7 and other organizations specifying standards
1.6. Behavioral Framework

The Behavioral Framework (BF) provides a set of constructs for defining the behavioral semantics of specifications, which enable Working Interoperability (WI).

As a result, the focus of the BF is accountability – a description of “who does what when.” Accountability describes the perspective of the various technology components that are involved in a particular instance or scenario designed to achieve Working Interoperability. The BF is technology-neutral and, therefore, can be used within model-driven specification stacks, such as the Services-Aware Interoperability Framework (SAIF) Enterprise Conformance and Compliance Framework (ECCF).

This discussion assumes that the BF is one of the sub-frameworks of SAIF. As explained in the SAIF Overview, each sub-framework is a grammar or set of metamodels, which enables one to describe particular aspects of a specification that is associated with the specified component’s involvement in an instance of Working Interoperability.

In particular, the BF specifies the grammar that is used to construct the essential artifacts necessary to comprehensively specify the various aspects of the Computational (and, to a lesser degree, the Information) viewpoints of the ECCF’s specification stack instances for a given organization’s implementation of SAIF.

Behavioral Framework Essentials provides a detailed description of the fundamental concepts and constructs of the BF. Behavioral Framework Foundational Concepts and Models presents the various models that collectively define the BF at each of the three levels of the ECCF:

- The Computationally Independent Model (CIM)
- The Platform-Independent Model (PIM)
- The Platform-Specific Model (PSM)

Using the Behavioral Framework Packages and Behavioral Patterns discuss overall usage guidelines and BF patterns that are essentially implementation-neutral. Following are Behavioral Framework Appendixes that present the mapping of the BF to the HL7 Legacy Dynamic Model and a more detailed discussion of the differences between Reference Information Model (RIM)-based HL7 Version 3 messaging and services, as examples of two different interoperability paradigms.

Note The BF is informed and to a large (but not exclusive) extent scoped by the Reference Model for Open Distributed Processing (RM-ODP) Computational viewpoint and draws on the terminology of RM-ODP whenever possible.

1.6.1. Behavioral Framework Overview

The Behavioral Framework (BF) describes both the functional decomposition of systems and the means by which they interact with their environment and with other systems. The BF also provides the associated static semantics, which are bound to various specified behaviors.

Thus, the BF focuses on the specifics of the actual run-time behavior of software running at a computational node in a deployed architecture, e.g. a software component’s interface. More specifically, a “focus on behavior of a node” means that the quantitative, unambiguous specification and documentation of the details of conversations or interactions between nodes, which collectively create business value. Examples of such “conversations or
interactions” include everything from simple push messaging, to publish-subscribe distributions, and to longer-running, multiparty transactions.

The BF combines notions of a loosely coupled event-driven architecture – and is thus compatible with a traditional message-based environment, such as HL7 V2 or V3 – with inter-component procedural activities to achieve three overarching goals and capabilities:

- Documentation of human-mediated interoperability patterns, such as those present in healthcare IT solutions.
- Documentation and encapsulation of automated interoperability patterns.
- Documentation of the definitional characteristics of the technological structures (for example, components and interfaces) that assume roles within a deployed architecture in a manner that enables the definition and validation of accountability at a per-component granularity.

Because the BF is intended to be used in the context of the ECCF, it facilitates the development of testable and certifiable conformance statements, which denote conformance points at which a given implementation can make pairs of conformance assertions.

**Note**: The ECCF document defines and discusses the concepts of conformance statement, conformance assertion, and certification.

**Important**: The BF should not be confused with a given architecture specification’s formalisms, which are used to express conformance statements within a given ECCF specification stack instance, but rather should be seen as a grammar for expressing these statements.

This document is primarily concerned with defining the syntax and semantics of the BF rather than providing an explanation of how it is applied. Each organization adopting SAIF will develop the specifics of using the various SAIF grammars in an organization-specific SAIF Implementation Guide. As such, this discussion provides relatively few concrete examples of BF applications, specific artifacts, and so on. However, when it is helpful in defining a specific syntactic or semantic point in the definition of the BF, a brief example is included.

**Note**: By necessity, certain formalisms are required to express the BF’s core concepts. When a particular formalism constitutes a normative choice, the text will note that choice. Similarly, certain components, concepts, and constructs of the framework persist through specifications irrespective of their content or context, and the discussion will note those situations.

The formal models associated with the HL7 Behavioral Framework, which are included in Behavioral Framework Foundational Concepts and Models, are published at: http://www.ncientarch.info/hl7_bf/hl7_bf/

**Related information**

- Section 1.3.4.2.5. Reference Model for Open Distributed Processing
- Section 1.4.4. ECCF Foundational Concepts
- Section 1.8. Implementation Guide

**Note to reviewers**: I’ve included just the Behavioral Framework Overview here. For the entire BF document, see the Behavioral_Framework_for_peer_review_20100312.pdf document.

### 1.7. Information Framework

The Information Framework (IF) is a SAIF-compliant recasting of existing HL7 expertise regarding the specification of static semantics.

The Information Framework will draw on the information available from the following sources:
HL7 Services-Aware Interoperability Framework (SAIF)

- Storyboards
- Domain Analysis Models (DAM)
- Reference Information Model (RIM)
- Vocabulary concepts
- HL7 Core Principles

A PowerPoint presentation on the SAIF Information Framework (IF) will be available by the May 2010 Work Group Meeting (WGM). More information on the Information Framework will be coming soon.

Note: The Information Framework and Implementation Guides are not part of the current peer review. This material will be made available for Peer Review during the summer, in advance of the Fall 2010 meeting.

1.8. Implementation Guide

The SAIF Implementation Guide is the revised version of the HL7 Development Framework (HDF) document. The experiences of the Alpha project participants will shape this document, which will be available soon.

Related information

► HL7 Development Framework (HDF)

1.9. SAIF Examples and Lessons Learned

The information for the SAIF examples and lessons learned will come from the Alpha projects, and the documents from the National Cancer Institute (NCI).

In early 2010, a new SAIF presentation for new developers at the National Cancer Institute (NCI) will be available. This presentation will give them enough information to get up and running with SAIF.

An educational package will describe how to implement SAIF at all levels, and how to choose the artifacts based on your role (analyst, architect, or developer).

More information on the SAIF examples and lessons learned will be coming in a few months.

1.10. Glossary

This glossary provides definitions of the core concepts involved in discussing the SAIF and its component parts.

**Architectural style**

A family of systems that is defined in terms of a pattern of structural organization. More specifically, an architectural style defines a vocabulary of components and connector types, and a set of constraints on how they can be combined. [Shaw and Garlan]

**Architecture**

Is a set of resources, relationships, patterns, practices, and processes that collectively define a system and
Artifact
Any portion of a specification that is controlled and can be versioned. For example, an artifact can be a model, a document, or an XML schema.

Candidate component
An application, service, tool, or infrastructure component that is evaluated for compatibility with the National Cancer Institute (NCI) CBIIT Enterprise Architecture Specification (EAS).

Compliance assessment
A process which results in a single descriptor – a compliance level -- applied to a specific application, tool, service, or infrastructure component in the context of an Enterprise Architecture Specification (EAS).

Compliance is a function of:

- A set of static conformance statements collected as a conformance specification; and
- An accompanying set of dynamic processes that define the responsibilities and activities involved in performing conformance measurements. The compliance of a given application, tool, service, and infrastructure component within an Enterprise Architecture Specification is expressed as a compliance level. In cases where different components of an overall system are assessed to be compliant at different compliance levels, the lowest level of compliance is used for description of the overall system.

Compliance level
A classification of the degree of compatibility of a given application, tool, service, and infrastructure component to a conformance specification. Compatibility levels establish categories of compatibility which reflect NCI CBIIT requirements for semantic interoperability both internally (such as within the caBIG™ community), as well as with other organizations.

Conformance assertion
A statement that is made against technology that must be tested to be validated.

Conformance measurement
A binary-valued variable that reports whether a particular technology binding’s implementation of an application, tool, service, and infrastructure component satisfies a particular conformance statement. For example, is conformant or non-conformant based on one or metrics defined by the conformance statement as being essential for conformance to the assertion.

Conformance specification
A collection of Conformance statements made in the context of an Enterprise Architecture Specification (EAS).

Conformance statement
A testable, verifiable statement made in the context of a single RM-ODP viewpoint. In particular:

- Conformance statements may be made in four of the five RM-ODP viewpoints -- Enterprise, Information, Computational, and Engineering.
- The Technology viewpoint specifies a particular implementation technology binding that is run within a test harness to establish the degree to which the implementation conforms to a given set of conformance statements made in the other RM-ODP viewpoints.
- Conformance statements are conceptually non-hierarchical. However, conformance statements may have hierarchical relationships to other conformance statements within the same viewpoint (that is, being increasingly specific). They are not, however, extensible in and of themselves.
- Extensibility of a given conformance specification in the context of an EAS is defined through additional conformance statements from the Enterprise, Information, Computational, Engineering and...
Technology viewpoints (most commonly the latter two), as manifested in specific instances of Platform Specific Models.

- Localization indicates custom modifications or other alterations (including ignoring) specific conformance statements in a local context, which results in non-compliance to the overarching EAS. Specific localization semantics are identified in the context of the conformance and compliance assessment process.

Constraint
A limitation or restriction. A constraint is an additional factor that limits the semantics when compared to the original conformance statement to which must be complied.

Constraint pattern
A set of constraint processes or details that can be repeatedly applied to multiple appropriately-similar situations. For example, the National Cancer Institute (NCI) applies a particular constraint pattern that reduces the set of Person attributes and associate ISO 21090 data type properties for all specifications that use the RIM Person class.

Enterprise Architecture Framework (EAF)
The set of rules, constructs, and specific aspects of an overall framework that define, govern, or otherwise inform and enable to define, design, and development Enterprise Architecture components (also called architecture primitives).

Enterprise Architecture Products (EAP)
Any specification or balloted standard that HL7 produces. Products may also be defined as "manufactured composite structures that are assembled from engineered architecture primitives."

Enterprise Architecture Specification (EAS)
The set of architecture primitives, processes, patterns, and principles that collectively define how HL7 produces products for its external stakeholders.

Enterprise Architecture focuses on producing a number of architecture primitives (the result of engineering efforts) -- which may then be assembled into composites (the result of manufacturing efforts) -- commonly referred to as products.

Functional profile
A profile that specifies subsets of expected operations from a computational viewpoint.

HL7 foundational components
Certain V3-related artifacts that have been identified as essential components of either the HL7 Enterprise Architecture or the HL7 SAIF.

Integration point
A physical realization of a conformance statement that allows verification of the assertion. Compliance testing for conformance and non-conformance to a particular Enterprise Architecture Specification occurs exclusively at defined integration points.

Interoperability
The ability of two parties, either human or machine, to exchange data or information. (The HL7 interoperability paradigms are messages, documents, and services.)

Logical coherence
Logical or natural connection or consistency. The quality or state of cohering, especially a logical, orderly, and aesthetically consistent relationship of parts.

Pattern
A common solution to a common problem in a given context.

**Profile**
A subset of the conformance statements that are useful and might be implemented independently.
Functional subsets are from a computational viewpoint and semantic profiles are from an informational viewpoint.

**Semantic profile**
A profile that specifies subsets of the overall information content that supports the functional profiles.

**Semantics**
Pertaining to, or arising from the meaning of different words or other symbols.

**Services-aware**
The inclusion of services, such as business processes, in the HL7 interoperability standard, as well as the exchange of messages and sharing of documents.

**Services**
An activity or series of activities of intangible nature that normally, but not necessarily, take place in interactions between customers and service employees and/or physical resources or goods and/or systems of the service provider, which are provided as solutions to customer problems (Gronroos, 1990).

**Software Engineering Process (SEP)**
Any process that integrates the efforts of software development teams to craft implementable code and to deploy that code. Examples include RUP, Agile, UPF, Scrum, and others.

**Specification stack**
A set of instances focused on a business-appropriate topic. Each stack is contextualized by the notions of conformance and governance, and is internally consistent across various viewpoints.

**Technology component**
A logical unit that conforms to one or more conformance statements, as stated in a specification or standard. An RM-ODP viewpoint in which a specific technology binding asserts conformance via conformance assertions to a set of conformance statements.

**Trading partners**
Two parties who want to interoperate in some manner to accomplish a common goal.

**Working Interoperability**
The collection of structures, processes, and components that support Computable Semantic Interoperability (CSI) between two parties (“trading partners”) who are interacting (for example, exchanging information, coordinating behavior) to achieve one or more business goals. **Interoperability**, in this context, is further defined to be the **deterministic** exchange of data or information in a manner that **preserves shared meaning**.