Standards Australia

Information in fields should be filled in if known or left as is.

|  |  |
| --- | --- |
| E‑health Interoperability Framework | |
| Part Title: |  |
| Designation: | SA HB 137—2013 |
| Part Number: |  |
| Supersedes Standard No: |  |
| AustralianORJoint: |  |
| Creation Date: |  |
| Revision Date: |  |
| Issue Date: |  |
| Committee Number: | IT-014 |
| Committee Title: | Health Informatics |
| Subcommittee Number: | IT-014-09 |
| Subcommittee Title: |  |
| Project Manager: |  |
| PMs Email Address: | @standards.org.au |
| Shadow Project Manager: |  |
| Shadow PM’s Email Address: |  |
| WP Operator: |  |
| Project Office ID Number: |  |
| Project Number: | 101128 |
| Combined Procedure?: |  |
| Committee Doc No.: |  |
| Stage: | WORKING DRAFT |

**DRAFT – FOR STANDARDS DEVELOPMENT & REVIEW ONLY**

|  |  |
| --- | --- |
| Committee Reps: | Aged Care Association Australia  Allied Health Professions Australia  Australasian College of Health Informatics  Australian and New Zealand College of Anaesthetists  Australian College of Nursing  Australian Commission on Safety and Quality in Healthcare  Australian Healthcare and Hospitals Association  Australian Information Industry Association  Australian Institute of Health & Welfare  Australian Institute of Radiography  Australian Medical Association  Australian Private Hospitals Association  Commonwealth Department of Health and Ageing  Consumers Federation of Australia  Consumers' Health Forum of Australia  CSIRO e-Health Research Centre  Department of Health (Vic.)  Department of Health (WA)  Department of Human Services  Edith Cowan University  Engineers Australia  GS1 Australia  Health Informatics Society of Australia  Health Information Management Association of Australia  Integrating the Healthcare Enterprise Australia  Medical Software Industry Association  National ICT Australia  National E-Health Transition Authority  NSW Ministry of Health  Queensland Health  Royal Australian College of General Practitioners  Royal Australian College of Medical Administrators  Royal College of Pathologists of Australasia  The Pharmacy Guild of Australia  The Royal Australian and New Zealand College of Radiologists  University of Western Sydney |
| Additional Interests: | ACT Health  Central Queensland University  Deontik  Department of Health (NSW)  Department of Health (SA)  DH4  e-Health Research  Flinders University Northern Territory  Global Informatic Health  HL7 Systems and Services  ISN Solutions  Kestral Computing  Lantana Group  Laughing Mind  Llewelyn Grain Informatics  Mater Children's Hospital  Montage Systems  Ocean Informatics  Semantic Identity  Smart Health Solutions  University of Wollongong  Victoria Avenue Medical Centre |
| Product Type | Product type (eg: AS, ASAM, HB, JHB, JS, JSAM) |
| Pathway | Pathway Type (Standards Driven/Committee/Externally Funded) |
| Sector | Sector (SA PM to Fill) |

Amendment

|  |  |  |
| --- | --- | --- |
| Amendment No.: | |  |
| Amdt Published Date: | |  |
| Std Published Year: | |  |
|  |  | |

PREFACE

This Australian Handbook was prepared by the Standards Australia Technical Committee IT-014, Health Informatics.

The Standards Australia Technical Committee IT-014 recognizes the work of the Standards Australia Sub-committee IT-014-09, EHR Interoperability, and the initial work of the National E-Health Transition Authority (NEHTA) in the preparation of this Handbook.

This Handbook describes an E-health Interoperability Framework (EHIF) that provides a set of concepts, principles and approaches to support the building of interoperable e‑health solutions that span organizational boundaries. The framework is applicable at the local, jurisdictional and national level to solutions such as e-discharge summaries, e-referrals, e‑medications and the reporting of clinical findings.

This Handbook supplements existing enterprise architecture frameworks by recommending components for specifying interoperable e‑health solutions and guidelines for assessing interoperability capability. The proposed approach aims to facilitate structured discourse among stakeholders and consistency of interoperable e‑health solutions.

The framework adopts relevant concepts from the *ISO/IEC/ITU-T Reference Model of Open Distributed Processing (RM-ODP)*[[1]](#footnote-1) and is influenced by the HL7 Service Interoperability Framework (SAIF)[[2]](#footnote-2) as well as previous NEHTA architecture experience and related publications, including NEHTA’s *Interoperability Framework,* version2.0.[[3]](#footnote-3)

The framework captures the key concerns of those involved in specifying, building, integrating and supporting the evolution of interoperable e‑health solutions, including components of the national e‑health solutions. It describes a common approach to interoperability that supports information being exchanged and understood, securely and safely, by all parties in an e‑health endeavour. It is intended to promote shared conversation among the following stakeholder groups:

1. E‑health specification and standards developers—by providing architectural foundations for specifying and building interoperable systems.
2. System and software developers and systems integrators—by promoting delivery of interoperable e‑health infrastructure and e‑health solutions.
3. System testers—by promoting vendor solutions that are of high quality and readily testable.
4. Policy makers and regulators—by providing a structured framework for capturing legal and policy requirements and identifying the benefits and risks of e‑health solutions.
5. Healthcare providers—by providing a structured framework for clinical engagement, and capturing healthcare processes, leading to the design of fit-for-purpose e‑health solutions.
6. Organizational management—by providing a structured framework to specify business processes and identify e‑health governance requirements.

The term ‘informative’ has been used in this Handbook to define the application of the appendix to which it applies. An ‘informative’ appendix is for information and guidance only.

This publication has been developed with assistance from the Australian Government Department of Health and Ageing. The Australian Government makes no representation or warranty that the information in this publication is correct and accurate.

Standards Australia wishes to thank the Department of Health and Ageing for its continued financial support in helping to develop this Australian Handbook.

CONTENTS

*Page*

FOREWORD 8

1 Scope and general 10

2 Interoperability principles 13

3 Framework description 16

4 Using the framework 39

5 Recommended document content 41

6 Interoperability capability 53

APPENDICES

Appendix A Background to the framework 59

Appendix B NEHTA instantiation of the framework 63

Appendix C Service taxonomy 65

Appendix D Technical interoperability capability and existing approaches 66

Appendix E CDA conformance levels 68

Bibliography 69

FOREWORD

Australian and international experience has shown that inadequate ability to interoperate across human-to-human, human-to-system and system-to-system dimensions has caused significant costs to society—both financial and human related. Such costs are also highlighted in the *National E‑Health Strategy,*[[4]](#footnote-4) which recommends the implementation of a world class e‑health capability in Australia to support an environment where ‘consumers, care providers and health care managers can reliably and securely access and share health information in real time across geographic and health sector boundaries’(p. 1). Further, a recent US report, *Health IT and Patient Safety,*[[5]](#footnote-5) identifies interoperability as a key factor contributing to patient safety (along with usability and clinical workflows).

The intention of this Handbook is to address part of this interoperability problem by identifying key concepts, principles and approaches that can contribute to the common language supporting discourse about the specification and building of e‑health systems capable of interoperating at local, jurisdictional or national level.

The framework supports two communities of practice known for their use of complex and difficult languages: medicine and computer science.[[6]](#footnote-6) It accommodates the needs of both communities by incorporating both technical and organizational perspectives.

Effective interoperability needs to address the exchange of information both between systems and between organizations and also to recognize the ongoing need for the development and support of e-health solutions.

NOTE: The emphasis of the e‑health interoperability framework is on supporting predictable and consistent interactions between organizations, irrespective of the frameworks or methodologies used by either party. It is not an enterprise architecture framework or a solution architecture framework; instead, it supports emerging properties of complex e‑health system-to-system interactions.

It is hoped that use of the e‑health interoperability framework will promote the adoption of common concepts, principles and patterns in the specification of interoperable e‑health solutions by Australian organizations—and bring downstream benefits in the design, procurement and implementation of e‑health solutions.

As in several other industries, there are numerous benefits of interoperability in the health sector—for providers, patients and society as a whole.[[7]](#footnote-7) The adoption of interoperability solutions in e‑health does involve cost and effort, but the benefits are predicted to far outweigh the costs.[[8]](#footnote-8)

To support international e‑health alignment, SA HB 137—2013 reflects the latest e‑health standardization, in particular SAIF[[9]](#footnote-9)and the HL7/OMG Healthcare Services Specification Project.[[10]](#footnote-10) The SAIF in turn leverages standards, such as HL7, RM-ODP and various service-oriented architecture approaches.

The e‑health interoperability framework described in this Handbook also takes into account relevant aspects from the *Australian Government Architecture Reference Models*.[[11]](#footnote-11)

1. Appendix A provides further detail on the background to this Handbook.

Australian Handbook

1. Scope and general
   1. Scope

This Handbook describes a set of principles and Standards-based approaches for specifying and achieving interoperable e‑health systems. It identifies—

1. guiding principles, goals, approaches and requirements for achieving e‑health interoperability;
2. a Standards-based framework and related concepts for specifying e‑health interoperability requirements;
3. recommended document components for specifying interoperable e‑health solutions and the role, content and uses for each of these components; and
4. guidelines for defining the capability of health care organizations to interoperate.
   1. Application

This Handbook is intended to be read in conjunction with SA HB 138—2013, *E‑health Architecture Principles.*

The framework and specifications identified in this Handbook are intended primarily for use in cross-organizational contexts, but some of the core principles and approaches may also be applicable in an individual organization or unit. Interoperability is one of the key factors that should guide the specification, development, acquisition, implementation and use of e‑health systems.

* 1. Intended audience

This Handbook is aimed at organizations involved in the design, building, integration and use of e‑health solutions or developing e‑health components of national significance. This includes—

1. e‑health specification and standards developers;
2. system and software developers and systems integrators;
3. system testers;
4. policy makers and regulators;
5. healthcare providers; and
6. organizational management.
   1. Referenced documents

Documents referred to in this Handbook are listed in the Bibliography.

* 1. Definitions

For the purposes of this Handbook, the following definitions apply.

* + 1. Interoperability

The ability of two or more systems to interact with one another and exchange information according to a prescribed method in order to achieve predictable results.

* 1. ‘System’ covers IT systems but is broader in scope.
  2. In the context of this Handbook, this ability needs to persist for as long as the need to exchange information between the systems continues to exist.

[Adapted from ISO 16056-1:2004, Clause 3.24.]

* + 1. Terminological resource

Coding system (ISO 17115:2007, 2.7.3), classification (ISO 17115:2007, 2.7.1) or terminology (ISO 1087-1:2000, 3.5.1).

* 1. Acronyms

| Term | Description |
| --- | --- |
|  |  |
| BPMN | Business Process Modelling Notation |
| CDA | Clinical Document Architecture, a standard produced by HL7 International for representing clinical documents in electronic form |
| ECCF | Enterprise Conformance and Compliance Framework |
| EHAP | E‑health Architecture Principles |
| EHIF | E-Health Interoperability Framework |
| EHR | Electronic health record |
| ETP | Electronic Transfer of Prescriptions |
| HL7 | Health Level Seven International Inc |
| HL7 RIM | HL7 Reference Information Model |
| HPI-I | Health Provider Identifier – Individual (as issued by the Healthcare Identifiers Service) |
| HPI-O | Health Provider Identifier – Organisation (as issued by the Healthcare Identifiers Service) |
| HSSP | Health Services Specification Project (a joint activity of HL7 International and OMG) [Ref. 22] |
| LOINC® | Logical Observation Identifiers Names and Codes[[12]](#footnote-12) |
| NEHTA | National E-Health Transition Authority |
| ODP | Open Distributed Processing |
| OMG | Object Management Group |
| OWL | Web Ontology Language |
| PCEHR | Personally Controlled Electronic Health Record |
| PKI | Public Key Infrastructure |
| RACGP | Royal Australian College of General Practitioners |
| REST | Representational State Transfer |
| RIM | Reference Information Model |
| RM-ODP | ISO/IEC/ITU-T Reference Model of Open Distributed Processing [Ref. 35] |
| SAIF | Service Aware Interoperability Framework |
| SNOMED CT® | Systematized Nomenclature of Medicine Clinical Terms[[13]](#footnote-13) |
| SoaML | Service-Oriented Architecture Markup Language |
| UML | Unified Modelling Language |
| WSDL | Web Service Definition Language |

1. Guiding principles for interoperability
   1. Overview

The principles outlined in Clauses 2.2 and 2.3 below can be used to guide the e-health solution development efforts of e‑health organizations.

The aim of these principles is to facilitate consistency of national e-health architecture approaches and to support the building and operating of consistent and interoperable e‑health systems.

These principles are divided into two categories:

(a) General principles, which reflect commonly used business and ICT best practices, drawn direct from SA HB 138—2013, Section 2. These principles are listed in Clause 2.2.

(b) A suite of principles specific to interoperability in e‑health, and amplifying the general e‑health architecture principle: ‘Ensure e‑health solutions support interoperability’ [SA HB 138—2013, *E‑health Architecture Principles,* EHAP 3, Clause 2.4]. These principles are listed in Clause 2.3.

* 1. General principles

The following general e-health architecture principles from SA HB 138—2013 are also relevant to the definition, implementation and governance of e-health interoperability solutions:

1. EHAP 1: Improve the safety and quality of healthcare.
2. EHAP 2: Improve the efficiency of healthcare services.
3. EHAP 4: Ensure solutions are fit for purpose.
4. EHAP 5: Support service-oriented approaches.
5. EHAP 6: Comply with legislative and policy requirements.
6. EHAP 7: Re-use e‑health components.
7. EHAP 10: Maintain security.
8. EHAP 11: Assess whole-of-life costs.
9. EHAP 13: Manage information quality.
10. EHAP 16: Express policy compliance as business rules.
11. EHAP 17: Support loose coupling.
12. EHAP 20: Ensure supportability, sustainability and continuity.
13. EHAP 21: Manage change.
    1. Interoperability principles

2.3.1   General

This document expands on EHAP principles 3, 9 and 18 from SA HB 138—2013. The titles of Clause 2.3.2 , 2.3.3 and 2.3.4 are the original EHAP principles as listed in SA HB 138—2013. The text in each clause expands on the principle listed in the heading.

**2.3.2   EHAP 3: Ensure e‑health solutions support interoperability**

This principle means the following:

* 1. *Universal participation*

All health stakeholders should be able to exchange health information, irrespective of the level of their technical capability.

* 1. *Enabling interoperability*

E‑health stakeholders need to define and publish the levels of capability they are capable of supporting.

* 1. *Policy compliance*

Interoperability solutions need to comply with applicable policies in all jurisdictions and organizations within which they operate.

* 1. *Resolution of policy conflicts*

Where applicable policies conflict, it is the responsibility of the interoperating parties to achieve a workable outcome.

* 1. *Observance of standards*

Interoperability needs to be achieved through rigorous adherence to applicable standards.

* 1. *Conformance and compliance*

Systems and interfaces for interoperability need to be tested against agreed interoperability conformance requirements.

* 1. *Governance of change*

Those providing interoperable solutions need to institute collaborative processes for governing, managing and communicating changes affecting e‑health interoperability, including changes to exchanged information, exposed service interfaces and business rules.

* 1. *Agreement on common semantics*

Effective, safe e‑health interoperability requires the interoperating parties to have a common understanding of the concepts embodied in policies, business services, terminologies and data definitions.

* + 1. **EHAP 9: Engage with all relevant stakeholders**

This principle highlights the need for stakeholder engagement*.* Interoperability solutions should be developed and maintained through active engagement with stakeholders who have applicable end-user, business and technical perspectives, and take into account national e‑health policies and solutions.

* + 1. **EHAP 18: Express policy in technology-independent terms**

This principle centres on the need for separation of business rules*.* Where possible, the design of systems should separate the expression of business rules from the applications and technology that interpret them, thereby maximizing flexibility and minimizing the cost of accommodating changes in business rules.

1. Framework description
   1. Summary
      1. Structure

The E‑health Interoperability Framework described in this section identifies a commonly used set of e‑health concepts, structured according to five architecture viewpoints and three design abstractions as influenced by HL7 SAIF [Ref. 20]. The structure is represented as a partially completed matrix, as shown in Figure 1. The cells of the matrix are intended to be populated with specification artefacts—this is its primary purpose, as explained in Clause .1. The matrix can also be used to show the different stakeholder concerns, as explained in Clause .

The key cells are shown with solid lines, and the optional cells with dashed lines.



Figure 1 E‑health Interoperability Framework structure

The five columns cover the architecture viewpoints of the ISO RM-ODP standard (these columns are further described in Clause ). The viewpoints in the columns reflect the concerns of the different stakeholder groups involved in the architecture development process, beginning with strategic planners and clinical/subject matter expert roles (enterprise viewpoint), and continuing via information/solution/system architects and developers (information /computational/engineering viewpoints) to testers and system integrators (technology viewpoint).

The three rows provide an additional level of refinement. These refinements are views of the system from the perspective of different sub-groups of stakeholders, and are expressed in terms of conceptual, logical and implementable perspectives or design abstractions.

Consider, for example, the information viewpoint, which is the focus of those interested in the information used in and exchanged by a system. This group of stakeholders can be subdivided into—

1. those interested in the conceptual view of information, such as clinicians and other subject matter experts, with their preferred ways of representing information, such as conceptual maps;
2. those interested in the logical view of information, in terms of information models and terminology binding, such as clinical information modellers, clinical terminologists and information architects who use more formal representation approaches such as UML; and
3. those interested in the implementable perspective, such as developers who use specific sets of datatypes and terminologies to develop solutions, which can, in turn, be implemented using specific technologies or vendor products.

NOTE: While the empty parts of the matrix may not be relevant to an interoperability specification, they may be relevant to aspects of an organization’s enterprise architecture, such as its need for policies governing technology platforms and operational processes, or service level agreements.

* + 1. Relevance of architecture viewpoints to stakeholders
       1. Overview

Although the primary role of the matrix is to provide a systematic approach to developing e‑health specifications (explained further in Clause ), it can also be helpful as a guide to the roles that need to be involved in the development of various types of e-health interoperability specifications.

An example set of such roles is shown in in Figure 2.

Each organization will have its own set of roles. A small organization may have the roles of software developer, system tester and CEO, while a larger healthcare provider may have many more roles defined.



FIGURE  2   FRAMEWORK STRUCTURE BY STAKEHOLDER VIEWPOINT

* + - 1. Enterprise viewpoint

The enterprise viewpoint is concerned with describing the scope and purpose of the IT system. This viewpoint is relevant to the strategic aspects of the system, which are of concern to senior executives, and other managers, as well as owners of business processes, subject matter experts responsible for business policies and procedures and end users. This viewpoint is also relevant for business analysts, business architects and business process modellers, all of whom capture business requirements and develop business processes, policies and collaborative arrangements between the stakeholders involved.

* + - 1. Information viewpoint

The information viewpoint is concerned with the representation of information in the system and is relevant for business (i.e. clinical and administrative) stakeholders and information modellers.

The major contribution in this viewpoint is expected from subject matter experts (i.e. clinicians), health informatics experts (i.e. clinical terminologists and informaticians) and information architects who document information objects and the appropriate clinical terminology concepts according to their preferred style of expression.

* + - 1. Computational viewpoint

The computational viewpoint is concerned with describing the functional decomposition of the system into computational objects which interact at their interfaces; this includes descriptions of services that objects offer and other objects consume, i.e. service contracts in general terms. These objects describe the key functionality of the system to be built, assuming that the necessary infrastructure support and services are specified elsewhere, using the engineering and technology viewpoint concepts described below.

This viewpoint is mainly relevant for solution architects and software developers, although a high-level computational description of the interaction between IT systems and users may also be used. This can be a refinement of the interactions defined in the enterprise viewpoint and can involve subject matter experts and business analysts.

* + - 1. Engineering viewpoint

The engineering viewpoint includes definitions of mechanisms and functions to support distributed interactions between computational objects as a series of templates (i.e. patterns) for computational interactions. These templates in turn are parameterized to support a range of different policies defined in the enterprise, information or computational specifications.

Examples of functions are repository (e.g. storage and information organization) functions, security (e.g. access control, authentication, security audit, integrity and confidentiality) functions, network services (e.g. naming services, time services and directory) functions, and type repository functions.

The engineering viewpoint is relevant for those who are providing infrastructure services and functions, such as system architects, network architects, security architects and middleware specialists.

* + - 1. Technology viewpoint

The technology viewpoint is concerned with the implementation stage and provides a link between specifications, expressed in terms of the other viewpoints in Figure 2 and the real implementation. This viewpoint is relevant as a guide for those who are implementing and testing systems for deployment in specific organizational contexts and need to make decisions about factors such as—

1. technologies available for the implementation (hardware, network products and infrastructure software);
2. standards to be used; and
3. resource constraints that need to be taken into account, e.g. existing legacy systems or services that need to be integrated with the new system.

Some of these decisions may reflect business or policy requirements of a particular organization, e.g. the use of open source products.

This viewpoint may also be needed to guide developers by providing additional implementation level details specifying how conformance testing of their products against specifications will be performed.

* + - 1. Viewpoint correspondences

The viewpoints described in Clauses 3.1.2.2 to 3.1.2.6 are a mechanism to support the separation of concerns of different stakeholders. The fact that the subject of the specification is a single target system implies the need to identify linkages between the concepts specified using different viewpoints.

For example, a business process (enterprise viewpoint) typically involves exchange of information objects (information viewpoint) and/or invocation of technical services (computational viewpoint). In order for a system specification to be complete, correspondences need to be identified between concepts from different viewpoints. In terms of stakeholders involved, these correspondences can be regarded as a set of agreements about the relationship between their views on the system.

Enterprise architects, who are concerned with an overall architecture of the system, are particularly interested in viewpoint correspondences to ensure overall consistency and integrity of the system.

* + 1. Design abstractions
       1. Overview

The rows of the matrix in Figure 2 provide additional levels of detail to categorize the viewpoints further according to the needs of different stakeholders. The rows categorize each of the viewpoint concerns in terms of conceptual, logical and implementable abstractions. This separation is in line with HL7 practices, in particular the recent requirements to support multiple interoperability paradigms (messages, services and documents), while also allowing for multiple target technologies for implementation, both information and behavioural types.

This separation is similar to the OMG’s CIM/PIM/PSM paradigm (June 2003) [Ref. 36], but more generic in that it supports a wider range of model-driven transformations, reflecting the specifics of the HL7 requirements.

* + - 1. Conceptual

The conceptual layer focuses on the subject matter expert’s view of a specific healthcare area. At this level, the details of the structure and processing of the system may not be specified. For example, a conceptual map could be used to represent key information objects in a shared EHR system. Depending on the subject matter expert’s skills, high-level UML class diagrams could also be used to depict key classes and their relationships, without mentioning their attributes and operations and therefore not considering the IT representation of information.

The conceptual layer is independent of the interoperability paradigm chosen (message, service or document).

* + - 1. Logical

The logical layer introduces more detail in terms of the behavioural and information descriptions of an IT solution. The logical layer will differ depending on the type of interoperability paradigm chosen, and will typically use UML or HL7 modelling notations.

Depending on the overall system requirements and the subject matter expert’s skills, some conceptual models can be categorized as logical models. One example is when all business services are supported by an underlying system, such as in a fully automated payment system.

Although the distinction between ‘conceptual’ and ‘logical’ may be blurred at times, the key point is that all the relevant models need to be provided in order to ensure a complete system specification.

* + - 1. Implementable

This layer is concerned with the specific choice of technology used to describe the components of the IT system—for example, a specific information type system, a specific vocabulary, specific technology to support interactions between components in the system (e.g. WSDL or REST), and specific access control security mechanisms.

* 1. Modelling concepts
     1. Overview

Figure 3 shows a set of modelling concepts (specification artefacts) that should be used as part of the e‑health specifications. The use of a common modelling language will facilitate conversation and, further downstream, the implementation of the e‑health specifications by various stakeholder groups, thus contributing to the delivery of sustainable and interoperable solutions. Each e-health specification will use a subset of the modelling concepts in Figure 3, depending on the nature of the system being specified.

NOTE: The implementable row shows examples of specific technologies that can be used to realize the key modelling concepts. The technologies selected to implement an e‑health interoperability solution should be based on one of the specific deployment scenarios identified in the technology viewpoint. The implementable artefacts are italicized in the figure to distinguish them from the related conceptual and logical concepts.



Figure 3 Framework structure populated with modelling concepts

* + 1. Enterprise viewpoint

The stakeholders concerned with enterprise aspects jointly capture and develop business requirements, and describe the business context through expressing collaborative arrangements between users—including their participation in the business processes and policies that apply to them.

Within an e‑health interoperability environment, the requirements documented as part of an enterprise viewpoint may reflect those arising from a combination of clinical, administrative, research and consumer-oriented domains.

* + - 1. Conceptual enterprise models
         1. Overview

The term ‘conceptual enterprise models’ collectively describes key specification artefacts used in the requirements elicitation, business requirements capture and business analysis stages in the development of an e‑health interoperability solution. These models describe key specification artefacts used in the requirements elicitation, business requirements capture and business analysis stages. These artefacts may be—

1. business scenarios;
2. business use cases; or
3. community models that define roles, business services, business policies and business processes.

The order in which these various artefacts may be developed may vary between organizations.

* + - * 1. Business scenarios

A business scenario describes a typical sequence of steps involved in performing different aspects of a business process. For example, in a care delivery situation, administrative as well as clinical aspects will be documented, such as the key steps involved when an individual registers for participation in a shared EHR system.

Depending on the purpose and particular needs of an e‑health interoperability solution, a business scenario may capture processes performed both manually and by supporting IT systems. A business scenario is used as a way of eliciting business requirements and can be a basis for developing more detailed business use cases, which is a more structured and formal way of expressing how actors in different roles interact with an IT system.

* + - * 1. Business use cases

A business use case is a structured and formal way of expressing how actors in different roles interact with an IT system. It captures high-level interactions between a stakeholder or stakeholders and a system, and provides a detailed description of the steps in a particular aspect of a business scenario.

For example, in the scenario in Clause 3.2.2.1.2, the typical steps for an individual registering for a shared EHR system consist of the following business use cases:

1. An individual accesses an EHR portal for the purpose of registration.
2. An individual supplies evidence of identity.
3. An individual defines specific access restrictions for his or her health records.

Typically, several business use cases involving different actors and their different interactions can be synthesized to express a collaborative arrangement between these stakeholders. These collaborative arrangements can form the basis of a community model, as described in Clause 3.2.2.1.4.

* + - * 1. Community models

According to RM-ODP [Ref. 41] a conceptual enterprise specification includes one or more communities, and one or more parties or enterprise objects which participate in these communities. A community model uses the RM-ODP concept of community to express key collaborative arrangements between parties associated with a proposed e‑health interoperability solution, by—

1. specifying the objective of collaboration;
2. defining key roles in the collaboration;
3. specifying behaviour in the community, either as business processes or as less structured interactions; and
4. documenting the set of policies that apply to the behaviour of roles.

A community model will be constructed by business analysts in discussions with subject matter experts and can then be passed on to business architects for further refinement.

A community specification addresses both the structural and behavioural aspects of a community. Structure is described in terms of community roles, reflecting responsibilities in the collaboration, including business relationships, such as reporting or delegation. Depending on the case, behaviour is described in terms of business interactions between community roles—specifying invoking or responding actions in an interaction, or in terms of a business process style of behaviour, stating data and control flow between objects fulfilling community roles. In the latter case, community roles are represented as partitions in the process model, and their process interactions are depicted through the control and data flow originating and terminating at specific roles.

In addition, community models specify policies as constraints on behaviour, such as permissions, obligations and authorization. Policies should then be shown as constraints on the roles and their interactions.

It is important to note that an overall enterprise specification of a system may contain a number of community contracts, modelling both hierarchical and peer-to-peer relationships between communities.

The following are the key modelling elements of the community model:

*Community roles* A community role defines a set of behaviours and responsibilities that can be taken up by a party participating in a community. The participation rules are defined by policies, defined by the community, or propagated into the community from its environment. Examples of the latter types of rules are regulative and legislative policies.

A community model should identify the relationships between community roles, for example those relationships arising from Community roles can be related, reflecting specific responsibilities defined in an organizational structures or cross-organizational arrangements.

*Business services* A business service specifies behaviour to be provided by some roles in the community (providers) offering value to other roles (consumers) [Ref. 32]. A business service can be described in terms of basic interactions between these roles and the policies that apply to providers and consumers. Examples of business services are provider lookup or pathology request.

NOTE: Business services will typically use one or more technical services, which are finer grained interactions, typically describing functionality of interfaces of the IT components supporting the business service. For example, a provider directory business service will use several technical services, such as provider lookup and provider update. Business services can also depend on certain non-IT services; for example, pathology tests will involve blood test analysis facilities, and this dependency needs to be captured in case certain manual interventions are required.

*Business process* Business processes describe the behaviour of the communities identified in the community models (see Clause ). BPMN diagrams or UML activity diagrams can be used, depending on the audience, the levels of abstraction required and the notation preference. The diagrams document the as-is and to-be processes within specific domain areas while depicting the position of IT systems as they support the business. These diagrams can be developed at various levels of abstraction, depending on the audience; this can include high-level processes for external engagement with stakeholders.

A business process can be encapsulated as a business service, and multiple business services (of lower level granularity) can participate in the same business process.

*Business policies* A policy is a constraint on behaviour that, in the enterprise viewpoint, applies to community roles, stating what they are required to do (obligations), what they are permitted to do (permissions), and their authority powers, for example in setting the policies in the community. A policy can describe constraints on how parties fulfil roles, as well as on the way they participate in interactions and business processes.

* + - * 1. Party

Party is used to model legal entities with lifecycles independent to those of a community. Some of these parties may be a generic model of parties—for example, a consumer or a health provider—while others may be concrete parties, such as Medicare Australia or the Royal Australian College of General Practitioners. Parties fulfil roles in a community, provided they satisfy constraints stated in the policies.

A party can fulfil multiple roles in one or more communities. This separation between party and community allows the expression of various business policies, such as separation of duty, and can provide a basis for expressing security policies, such as access control.

* + - 1. Logical enterprise models

Logical enterprise models provide a further level of modelling detail to conceptual enterprise models and are a step towards specifying computable models in the computational viewpoint.

Logical enterprise models typically describe detailed business process models showing interactions between parties, and between parties and IT systems. Interactions between parties and IT systems can lead to the specification of human computer interactions to be detailed in the computational viewpoint. The interactions can specify data and control flows between activities, including synchronization points, time-related constraints, business events and composition of individual activities into composite activities. Business processes can also specify the business services used and provided by the relevant parties, and how a business service offered by one party can be realized as a business process.

Logical enterprise models can also define a more detailed description of the policies identified in the conceptual enterprise models. This additional level of detail can capture the type of policy constraint, such as obligations, permissions, prohibitions, authorizations, consent, confidentiality, privacy, and conditions related to delegation and how they are passed from principals to agents. This level of detail is sufficient to express these constraints as business rules, which can be used to constrain the behaviour of components and parties in the computational viewpoint. This detail can provide a good basis for developing identity management and security architectures.

Logical enterprise models can also integrate policies, such as those identified in the previous paragraph, and process models. This has particular value when modelling interactions between parties in federated domains.

Effectively, logical enterprise models can be regarded as detailed community models derived from the high level community models in the conceptual enterprise models. They are expressed in sufficient level of detail to be specified using, for example, tools that implement UML and BPMN.

* + 1. Information viewpoint
       1. Key aspects

The information viewpoint focuses on the semantics of information, providing a shared understanding of the information managed by a system. It describes the information and the structure and content of the supporting data. Key aspects of the information viewpoint include—

1. information objects;
2. relationships between information objects;
3. constraints;
4. datatypes and their value domains; and
5. terminological resources.
   * + - 1. Information objects

An information objectrepresents information about some real-world entity, for example demographic information about an individual. An information object may be a composition of other information objects that are semantically related to any particular real-world entity. Information objects relevant to e‑health applications include objects representing administrative information or technical entities, such as identifiers or digital certificates, as well as information objects that represent clinical information.

Information objects need to be uniquely identifiable, to enable them to be correctly related to corresponding real world entities. Both names and identifiers may be used for identification of information objects but, in order to identify any given instance of an object and its related entity, an identifier is required that is unambiguous in the given context [Ref. 41].

Examples of identifiers include the HPI‑Os and HPI‑Is maintained by the Australian Healthcare Identifiers Service.

An example of a name in this context would be the term ‘prescribed medication’ to refer to an information object within a clinical document.

* + - * 1. Relationships between information objects

Relationships between information objects describe relationships between the real-world entities that they represent. Common types of relationships include associations, generalizations and dependencies.

Associations describe connections between different types of entities, such as the connections between a library and the books it holds, or between a clinical facility and its address.

A particular type of relationship, the composition or aggregation, describes how a more complex information object (e.g. a pathology report) may be composed of lower level objects (e.g. individual details, provider details, test results and digital signature).

Generalizations describe how information objects are grouped into hierarchies with common attributes, such as a surgeon being a type of medical practitioner (and therefore also being represented by information collected about a medical practitioner).

Dependencies indicate that one information object depends on another and that changes in the one have implications for the other (dependent) object. For example, an EHR entry is dependent on having a valid individual identifier. In most logical and implementable specifications, dependencies are specified as constraints, rather than being separately identified as dependencies.

* + - * 1. Constraints

Constraints apply to information objects or their relationships in particular contexts and represent rules or restrictions on allowed relationships or information values—for example, ‘An address shall have only one postcode and the postcode shall be valid for the country of residence’. More complex constraints may depend on dynamic values of the information objects.

* + - * 1. Datatypes and their value domains

Datatypes provide basic components and constraints used to detail information content in information specifications. These include datatypes formally defined as a ‘set of distinct values, characterized by properties of those values, and by operations on those values’(ISO/IEC 11404:2007, Clause 3.12; ISO 21090:2011, Clause 3.7). Examples include basic datatypes such as integers, strings and date-and-time, as well as aggregate datatypes such as the CD (concept descriptor) datatype for defining a clinical concept using an external clinical terminology. Some datatypes may also specify higher level data structures on which operations may be performed, such as an information record to be communicated between systems. Datatypes can be constrained to a value domain which is a specific set of permissible values—for example, the severity information object can be restricted to have a value of ‘mild’, ‘disabling’ or ‘life threatening’.

* + - * 1. Terminological resources

Terminological resources are classifications, terminologies and code sets that are used to represent concepts in a particular domain. Examples of widely accepted terminological resources that are managed and published via formal processes include the clinical terminologies SNOMED CT® and LOINC®, classifications such as ICD 10-AM and ICPC-2, and the various code sets maintained by the Australian Institute of Health and Welfare in the METeOR repository. Wherever possible, it is preferable for information objects to be specified in terms of shared terminological resources and datatypes, rather than custom-built, application-specific data specifications and value sets.

* + - 1. *Bringing the key aspects together*

These key aspects are brought together in information specifications that use information models to document the information objects within a particular domain, the relationships between them, any related constraints, and their relationship to concepts as defined by bindings to terminological resources. Examples include the production of sets of information specifications and information models for domains such as pathology reporting, medications, immunizations, discharge and referrals.

The use of formal information modelling approaches to document information models provides the basis for developing information models at conceptual, logical and implementable levels with the longer-term aim of supporting automated transformation between the levels to generate implementable specifications from conceptual requirements. Following a formal meta-information representation model, such as that recommended in ISO/IEC 11179, in documenting the different levels of information models will facilitate the consistent rendition (and transformation) of these information models.

* + - 1. Conceptual information models

The main criterion for an artefact to be part of the conceptual layer is its accessibility to subject matter experts, particularly to facilitate the expression of clinical concepts and value domains. The following information artefacts are among those identified in conceptual information models used to represent clinical information:

* Clinical concepts that represent real-world entities, such as adverse reaction, diagnosis and examination findings. These clinical concepts may be constrained by healthcare datatypes and value domains.
* A value domain, which stipulates a collection of allowable data values for the information objects defined in the concepts. These values can be sourced from various clinical and administrative terminological resources.
* People and organizations that participate in the clinical scenario. This includes those who are identified in the enterprise viewpoint.

Conceptual information models are commonly used in the analysis stage of the development process. Some examples of conceptual modelling accessible to clinicians and other subject matter experts are conceptual maps, high-level UML models (e.g. models depicting key classes without attributes) and archetypes (proposed by the openEHR or ISO/CEN 13606).

It is important to note that there is no need for an IT representation at the conceptual level.

* + - 1. Logical information models

Logical information models further refine and expand the conceptual models, with a more formal modelling notation. The objective is for the formal modelling technique to enable, through a well-known set of transformations, representations in one or more implementation technologies. For example, using UML as a representation notation would enable this, as would Semantic Web information models.

The aim is to use these artefacts as parts of the design stages in the development process.

The following are examples of such artefacts:

1. Logical clinical component—logical representations of the clinical concepts, defined in the conceptual information model.
2. Healthcare datatypes—modelling for frequently used information objects such as quantity, number, date, coded text and text. At the logical level, the semantics of each datatype are more important than the syntax (representation).
3. Structured templates—collections of logical clinical models which may be further constrained to meet domain-specific requirements, such as for a discharge summary or referral.
4. Terminological resources—classifications, terminologies and code sets. A clinical terminology, such as LOINC® or SNOMED CT®, consists of a set of uniquely identifiable terms, referring to clinical concepts (e.g. leg, calf, anaemia), an optional set of relationships between the terms (e.g. ‘part of’, ‘is a’) and an optional set of synonyms. The use of widely accepted clinical terminology allows clinical statements to be expressed in a semantically interoperable way.
   * + 1. Implementable information models

The main criterion for this layer is that it should contain descriptions of logical artefacts augmented with the details of specific implementation technologies used to describe both the information and behavioural semantics of these artefacts. The following are some examples, with their implementable mappings:

1. HL7 V3 clinical template—implementing logical clinical components.
2. CDA schemas—implementing the structured templates, e.g. referral templates.
3. HL7 V2 message schemas—implementing logical clinical components in HL7 V2 message segments.
4. SNOMED Release Format 2—implementing reference sets to produce subsets for a particular domain, such as medication.
5. OWL—implementing clinical ontologies.
   * 1. Computational viewpoint
        1. Overview

The computational viewpoint is concerned with describing the functional decomposition of a system into objects that interact at their interfaces. This viewpoint includes the description of services that objects offer through their interfaces and may be consumed by other objects.

* + - 1. Conceptual services models
         1. Overview

The design focus of the computational viewpoint is on the logical decomposition of the system, and is therefore mainly concerned with the logical service specifications at the logical abstraction level (see Clause 3.2.4.3). However, certain concepts can be described at the conceptual level, essentially providing more detail about the computational aspects than what is specified in the enterprise viewpoint. The main artefacts at this level are system use cases and specifications of high-level interactions between users and technical services (expressed in more detail through service contracts at the logical abstraction level).

* + - * 1. System use cases

System use cases describe the behaviour of the system in response to a trigger by a user. They are refined from business use cases. Several system use cases can be traced back to a business use case.

System use cases will ultimately be realizations of community model roles that represent IT systems, and can also be related to one or more information objects identified in the information viewpoints.

* + - * 1. Service interactions

Service interactions are high-level representations of functionality of those roles in a community that represent IT systems providing service as part of overall business interactions. Typically, the interactions will show the actions of roles such as providers, individuals and other healthcare roles, and the actions of service roles.

Interactions can be represented using UML behavioural diagrams, such as activity, sequence or state diagrams. The emphasis is on identifying capabilities to be provided to service users by the system, while also specifying actions of the system—which will be expressed in more detail as service contracts (see Clause 3.2.4.3.2).

A computational service can provide one or more capabilities. The conceptual models of a service focus on identifying on these capabilities from the computational viewpoint, including the respective pre and post conditions.

* + - 1. Logical service specifications

**3.2.4.3.1** *General*

The logical service specifications provide a formal, more detailed, platform-independent expression of system behaviour at the logical abstraction level expressed as service contracts and collaboration specifications.

* + - * 1. Service contract

A service contract describes the obligations of a system component, whereby one or more related interactions are grouped to form a service interface.

Typically a UML interface can be used to group such interactions and model a service. There are two approaches to service interface interactions:

1. The service contract is specified through the UML provided interface.
2. The service contract is described in terms of the provided interface and related required interface of the client component.
   * + - 1. Collaboration specification

A collaboration specification is a mirror of a certain fragment of behavioural specification stated in the enterprise specification, involving one or more community contracts. A collaboration specification can be represented through typical software architecture approaches, such as the following:

1. UML component models, showing dependencies between provided and required interfaces of the components defined in the service contract.
2. UML collaboration models, which can be used in more complex cases where a service contract involves multiple participants, and also supports the modelling and re-use of patterns.

NOTE: A more expressive approach is to use the ODP concept of compound binding, which allows the specification of orchestration aspects of interactions, but also allows possible transformations between messages exchanged. Either of these should be traceable back to the community model.

* + - 1. Implementable service specifications

The implementable cell under the computational viewpoint identifies details of the specific technology platform(s), and its model, where a model-driven approach is applied. This can then be used as the target for the platform-independent logical service specifications described in Clause 4.2.4.3 above. Examples are Web Service technologies, REST and .Net.

* + 1. Engineering viewpoint
       1. Overview

The engineering viewpoint includes the definition of mechanisms and functions to support distributed interactions between objects as a series of templates (i.e. patterns) for computational interactions. The following are examples of some useful functions for e-health applications:

1. Repository functions.
2. Security functions.
3. Type repository functions.

These belong to the logical level. There are no conceptual concepts in this viewpoint.

* + - 1. Logical specifications
         1. Repository functions

Repository functions cover concepts and rules for the following capabilities:

1. Storage function, defining the interface and object for storing data.
2. Information organization function, for managing a repository of information described by an information schema and including the function for—

modifying and updating the information schema:

querying the repository, using a query language; and

modifying and updating the repository.

* + - * 1. Security functions

Security functions cover the following capabilities:

1. Access control function.
2. Security audit function.
3. Authentication function.
4. Integrity function.
5. Confidentiality function.
6. Non-repudiation function.
7. Key management function.
   * + - 1. Type repository function

The type repository function manages a repository of type specifications and type relationships.

* + - * 1. Naming function

The ODP naming framework can be used to support naming service functionality, which is important for managing names of individuals, organizations, system components etc. It is a general naming framework for heterogeneous distributed systems, giving concepts and procedures that fully support general context-relative naming. These concepts can be applied in any ODP viewpoint. They can be applied to any function that uses naming and is subject to distribution and federation (see Ref. 12).

* + - 1. Implementable specifications

From the engineering viewpoint, the implementable specifications can capture specific functions and transparencies, such as the specific type of access control function, authentication function and other security functions, and specific type of repository model.

Some examples are—

1. PKI mechanisms;
2. attribute-based access control; and
3. XDS profile.
   * 1. Technology viewpoint

The technology viewpoint specifies implementation and deployment constraints that need to be taken into account when considering implementations of the specifications in a particular environment (in terms of other viewpoints). Therefore, the technology viewpoint details are mostly relevant to the implementable abstraction layer.

Artefacts that may be produced to support the technology viewpoint include—

1. lists of implementable standards, some of which may be more detailed artefacts specifically developed to support one or more of the information, computational or engineering viewpoints;
2. specifications identifying specific technologies to be used to deploy the e-health interoperability solution, e.g. hardware components such as computing nodes, communication links and routers, or infrastructure software such as operating systems or communication protocol drivers; and
3. resource constraints that need to be taken into account, e.g. existing legacy systems that may need to be integrated with the new system being specified.

This viewpoint also needs to identify the additional implementation level detail that vendors have to specify so that conformance testing can be performed against specifications.

In addition, this viewpoint provides for other implementation-related detail, such as configuration documentation, software implementation guides and processes for procuring or developing technology artefacts needed for implementation.

* + 1. Viewpoint correspondence

The ODP viewpoints provide separate but related abstractions of one system.

This specific type of relationship in ODP standards is referred to as a ‘correspondence specification’. The following are some examples of viewpoint correspondences:

1. The relationship between a business service and one or more technical services that implement it.
2. The relationship between artefact roles defined in the enterprise specifications, e.g. capturing data objects as part of process or interaction models, and information objects that specify information about these artefacts.
3. The relationship between a community contract and one or more service contracts that support computational interactions in the community.

The relationships between specific viewpoints modelling a particular system constitute a model in its own right. This approach to system specification facilitates better support when dealing with changes in models, which in turn yields many downstream benefits when the model-driven tools are adopted in a particular organization.

* 1. Conformity assessment for e-health interoperability
     1. Overview

Clause 3.3 elaborates on the key ideas behind conformity assessment, covering conformity assessment concepts, and their benefits for stakeholders of the E-health Interoperability Framework. The conformity assessment aspects of the E-health Interoperability Framework are based on the ISO RM-ODP standard, HL7 SAIF [Ref. 41] and AS ISO/IEC 17000.

Conformity assessment is an important element for ensuring interoperability outcomes. It complements the specification and design phases in e‑health specifications development which address concerns from different viewpoints using the concepts presented in Clause 3.2. The aim of conformity assessment is to provide assurance to all concerned that—

1. the implemented systems satisfy the e-health specifications;
2. new specifications are consistent with other relevant standards and specifications; and
3. health professionals meet competence expectations for using the systems.
   * 1. Specifications and implementations

Specifications provide requirements for interoperability of e-health implementations.. There can be many implementations fulfilling a specification. Specifications provide well-structured models of desired system behaviour, and may be satisfied by many solution designs and respective implementations using different technologies.

Standards and specifications provide guidance for interoperability, but it is through some form of measured adherence to these standards and specifications that the benefits will best be realized. Such adherence can be achieved through the use of conformity assessment.

Specifications are of significant value for a national e-health framework. In order to ensure consistency with other specifications, the specifications being developed should comply with openly available standards and defined requirements, whereby conformance and compliance to the specification can be assured.

* + 1. Definition and stakeholders

3.3.3.1   Definition

Conformity assessment is defined as ‘the demonstration that specified requirements … relating to a product …, process, system, person or body are fulfilled’ [AS ISO/IEC 17000].

When used in the context of software systems, the term ‘conformity’ covers both compliance and conformance. These concepts are defined as follows:

1. Conformance is the measurement (by software testing) of the adherence of an implementation to a specification or standard in accordance with common practice in the information technology industry [Ref. 41].
2. Compliance means—
3. the consistency of new specifications with other relevant standards and specifications in the context of e-health specifications according to RM-ODP [Ref. 41]; or
4. adherence to ‘the requirements of laws, industry and organizational standards and codes, principles of good governance and accepted community and ethical standards’ [AS 3806].

3.3.3.2   *Conformity assessment elements*

Figure 4 shows the various elements of conformity assessment for e-health interoperability. These include—

1. requirements for building e-health systems, which can originate from a number of sources (e.g. customers, standards, regulations);
2. conformity assessment activities (e.g. software measurement, product certification, and accreditation—see Clause 3.3.3.3 for further details;
3. roles involved in the software development community (e.g. vendors, jurisdiction);
4. conformity assessment products (e.g. assessment schemes, conformity requirements, test specifications, test tools and risk assessment (based on ISO/IEC 17007:2009(E) *Conformity assessment—Guidance for drafting normative documents suitable for use for conformity assessment*); and
5. roles involved in the specification community which use conformity assessment products.

NOTE: A special type of specification community is a standards specification community. It has its own set of policies and processes, reflecting a specific governance structure typical of many standardization organizations (and thus enabling its mandate in local, national and international arenas).

3.3.3.3   *Conformity assessment activities*

Conformity assessment activities can take the following forms in relation to e-health interoperability:

1. *Accreditation*—this is ‘third-party attestation … related to a conformity assessment body… conveying formal demonstration of its competence to carry out specific conformity assessment tasks’ [AS ISO/IEC 17000]. The accreditation function in Australia is well established through the National Association of Testing Authorities[[14]](#footnote-14) and the Joint Accreditation System of Australia and New Zealand;[[15]](#footnote-15)
2. *Product certification—*thisis an activity in which a third party gives written assurance that a product (including processes and services) fulfils specified requirements. Determination against the applicable specified requirements may include testing, measurement, inspection, design appraisal, assessment of services and auditing as examples of techniques used to check whether or not the product meets the specified requirements [Ref. 6].
3. *Software measurement*—this tests whether a software product fulfils specified requirements by performing conformance assessment, as described in Clause .



Figure 4 Conformity assessment activities and stakeholders

**3.3.3.4   Framework to support the conformity assessment program**

The conformity assessment program recognizes the work of the HL7 SAIF [Ref. 20] in developing the ECCF, which provides a robust underpinning to support comprehensive e‑health compliance and conformance assessment. The ECCF provides an organizational framework in which interrelated artefacts are verified for consistency, traceability and compatibility. These terms are defined as follows:

1. *Consistency* is a characterization of the logical coherence of the artefacts that are defined in the specification viewpoints.
2. *Traceability* defines relationships that link an attribute or other feature of a particular specification artefact with another artefact in another (or the same) cell of the matrix.

Changes in the specification at one level of abstraction may impact on the next level. For example, changes in a conceptual specification may require corresponding changes in the related logical and implementable specifications.

Furthermore, this effect applies to any artefact (or portion of an artefact) that is incorporated by reference, regardless of how many levels of indirection there may be. This necessitates strict and rigorous use of a common system both for versioning of specifications and for cross-referencing the content of specifications.

1. *Compatibility* is a relationship between two or more conformance points involving two or more specification instances. The relationship identifies whether two or more implementations claiming to be conformant to the specification instances can achieve interoperability.
   * 1. Compliance assessment

3.3.4.1   *Definition*

The concept of compliance has been defined in Clause 3.3.3.1. Compliance assessment provides assurance of an organization’s adherence to regulations, policies, standards and specifications. This assurance is obtained through one of two methods: compliance inspections or compliance audits.

3.3.4.2   Compliance inspections

Compliance inspections are typically performed by self-assessment or by a third-party inspection body, and may also be performed by a second party (i.e. a customer). The outcome of a compliance inspection is an inspection report describing the condition of the object of assessment at the time of the inspection. Compliance inspections may apply to a business process, a product design, a service, a product (e.g. design and development standards of software or hardware), or an installation (e.g. the operation of an e-health system in a healthcare provider organization).

3.3.4.3  Compliance audits

Compliance auditsapply specifically to audits of management systems. They can be performed only by an accredited third party, and the outcome of a successful compliance audit is a certificate. Assessment schemes used in audits of management systems include—

1. information security management systems schemes;
2. information technology service management systems schemes;
3. quality management systems schemes; and
4. healthcare sector schemes (schemes to ensure that healthcare organizations meet levels of safety and quality in the services they provide to patients).
   * 1. Conformance assessment
        1. Overview

Conformance has been defined in Clause 3.3.3. Conformance assessment provides assurance that a software implementation adheres to e-health specifications. It gives users of conforming implementations confidence that an implementation will behave as expected, perform functions in a known manner, have interfaces or formats that adhere to a specification, and interoperate with other conforming implementations.

Conformance assessment can be performed by testing and is performed by a first party (also known as self-assessment), a second party or a third party. Third-party conformance assessment provides increased rigour and reduces the risk of an organization using a non-conforming implementation.

Further work may be required to identify approaches for conformance assessment of some systems, such as those using service-oriented architecture.

* + - 1. Conformance points

A conformance statement is a statement that identifies conformance points of a specification, and the behaviour to be satisfied at these points. Conformance assessment uses conformance points to derive requirements and test criteria that are to be satisfied by an implementation in order to claim conformance.

‘Conformance relates an implementation to a standard. Any proposition that is true of the specification must be true in its implementation’ [Ref. 41, Part 2].

Conformance statements will only occur in standards that are intended to constrain some feature of a real implementation, so that there exists, in principle, the possibility of testing [Ref. 41]. Typically the conformance statement is a high-level description of what is required of implementations. It may refer to other parts of the standard. It may specify sets of functions, which may take the form of profiles, levels, or other structures. It may specify minimum requirements for certain functions and for implementation-dependent values. Additionally, it may specify the permissibility of extensions, options, and alternative approaches and how they are to be handled.

The objective of any conformance statement and its related conformance points is to provide clear and unambiguous statements, so that the reader knows what is required in order to claim conformance, and what is optional. To achieve this objective—

1. the distinction between normative and informative sections needs to be evident and, if necessary, the sections need to be labelled accordingly.
2. uniformity of structure, style, and terminology needs to be maintained within the specification.
3. identical wording needs to be used to express identical provisions, and analogous wording needs to be used to express analogous provisions.
   * + 1. Examples

The following are examples of conformance points extracted from the solution specification ATS5822—2010, which specifies the required behaviour of a Sender when delivering a message.

## EXAMPLE 1

In this example, the Sender of the message is required to ensure privacy and the adequate authentication of the Receiver.

The Sender shall initially validate the authenticity of Receiver encryption certificates. These certificates will usually be returned with a service directory record, or retrieved through certificate references returned with the service directory record.

[ATS 5822—2010, Clause 3.3.5(a)]

## EXAMPLE 2

This example is a non-mandatory requirement for the Sender to apply to the invocation of the Sealed Message Delivery interface by a Sender.

The Sender should use a service directory (TR 5823—2010) to initially locate the record identifying the service interface for a Receiver, noting that this service interface might be operated by a Receiver Intermediary. The Sender should cache this record at least until a final transport response is received for the associated message interaction.

[ATS 5822—2010, Clause 3.3.2(c)]

1. Using the framework
   1. Overview

The E‑health Interoperability Framework can be used directly to support the specification of e-health systems. This means using principles such as those referred to in Section 2 to guide design, using artefacts and concepts from each of the viewpoints described in Clause 3.2 to specify design and using conformity assurance to support testing.

Organizations may wish to extend the Framework to reflect their needs, and it can be further instantiated to develop compliant solution frameworks reflecting particular localized contexts of use, as described in Clauses and .

1. Appendix B gives an example of one instantiation.

The Framework can also be used to develop relevant implementation and supporting material, such as the establishment of a service taxonomy.

* 1. Compliant solution frameworks

An organization may wish to define its own e‑health architecture framework that is compliant with the E‑health Interoperability Framework (described in Section 3. Such a framework could—

1. refine or extend the Framework with architecture decisions specific to that organization;
2. be aligned with the organization’s architecture methodology; and
3. be integrated with any architecture governance framework.
   1. Layering of solution specifications

Usingthe E‑health Interoperability Framework as a guide to the production of separate but consistent specifications of a system design at conceptual, logical and implementable levels and from different viewpoints is the basis for achieving interoperable e‑health solutions that can evolve over time to match changes in both business requirements and implementation technologies. Fundamentally, this process entails achieving the following:

1. *Conceptual solution specifications* that provide the domain experts' view of the problem and, to some extent, the potential solution space, typically expressed in terms of the language familiar to domain experts, e.g. clinicians and administrative personnel, as well as end-users.
2. *Logical solution specifications* that provide the specifications of interactions and services at the level of completeness required for mapping to various technology choices, but are independent of any such choices. In other words, a logical specification can be refined into several possible implementable specifications.
3. *Implementable specifications* that provide sufficient additional detail to enable the requirements expressed in the logical specifications to be consistently realised using specific technology choices, e.g. implementation of an e‑prescribing system using secure message delivery infrastructure.

NOTE: The implementable specifications may in turn be used as a basis for selection, adaptation or development of specific software products, and the conformity assurance activities will need to ensure consistency between implementable specifications and real implementations.

* 1. Solution implementations

Having a common architecture and system building approach based on the E‑health Interoperability Framework supports systematic conformity assessment practices at the implementation stage while both promoting interoperability and allowing innovation.

Specific implementations can be produced using a combination of conceptual, logical and implementable specifications.

* 1. Supporting material

The E‑health Interoperability Framework can be used to categorize service types, such as business services, technical services and infrastructure services.

1. Appendix C provides some initial ideas about the development of such a service taxonomy.
2. Recommended document content
   1. Overview

This section uses the E‑health Interoperability Framework, as presented in Section 3, to characterize e‑health document components, each of which may be used on its own as an e‑health specification or be grouped together with other components as part of a larger e‑health specification document.

Each e‑health document component is characterized in terms of the content of its sections. The characteristics of the document components are shown in Clauses 5.2 to 5.12. These clauses are structured in tabular form for maximum clarity.

The mapping of these e‑health document components to the E‑health Interoperability Framework matrix is shown in Figure 5.

These standard e-health document components provide a set of building blocks which individual e-health organizations may use to define standards-compliant document components that reflect their own needs.



Figure 5: Mapping of document components to the Framework structure

* 1. Environmental scan

|  |  |
| --- | --- |
| Perspective | Conceptual |
| Viewpoint | Enterprise |
| Content | The purpose of the environmental scan document is to describe the findings of a systematic analysis of the domain environment within which the adoption, implementation and maintenance of the solution to be developed will occur, covering—   * key stakeholders and end users; * existing business problems; * barriers to change; * existing e‑health standards; * the current and foreseeable future state of the domain environment within Australia; and * opportunities for e‑health capabilities which are implementable at a national level.   The environmental scan typically comprises—   * executive overview; * introduction; * domain description, current and future state; * [analysis](file://\\syd-fs2\SA\Standards%20Development\Communications%20IT%20and%20e-Commerce%20Services\IT-014%20Health%20Informatics\SubCom\IT-014-09\Projects\AppData\AppData\Local\Microsoft\Windows\Temporary%20Internet%20Files\AppData\Local\Microsoft\Windows\AppData\Local\Microsoft\BenSkinner\AppData\Local\Microsoft\Windows\Temporary%20Internet%20Files\Content.Outlook\D0XK5IJI\PD-PDLC04-EnvironmentScan_v0%204%201.doc#_Toc272477985), [stakeholders](file://\\syd-fs2\SA\Standards%20Development\Communications%20IT%20and%20e-Commerce%20Services\IT-014%20Health%20Informatics\SubCom\IT-014-09\Projects\AppData\AppData\Local\Microsoft\Windows\Temporary%20Internet%20Files\AppData\Local\Microsoft\Windows\AppData\Local\Microsoft\BenSkinner\AppData\Local\Microsoft\Windows\Temporary%20Internet%20Files\Content.Outlook\D0XK5IJI\PD-PDLC04-EnvironmentScan_v0%204%201.doc#_Toc272477986), [existing prevailing specifications](file://\\syd-fs2\SA\Standards%20Development\Communications%20IT%20and%20e-Commerce%20Services\IT-014%20Health%20Informatics\SubCom\IT-014-09\Projects\AppData\AppData\Local\Microsoft\Windows\Temporary%20Internet%20Files\AppData\Local\Microsoft\Windows\AppData\Local\Microsoft\BenSkinner\AppData\Local\Microsoft\Windows\Temporary%20Internet%20Files\Content.Outlook\D0XK5IJI\PD-PDLC04-EnvironmentScan_v0%204%201.doc#_Toc272477994), [regulation and policy](file://\\syd-fs2\SA\Standards%20Development\Communications%20IT%20and%20e-Commerce%20Services\IT-014%20Health%20Informatics\SubCom\IT-014-09\Projects\AppData\AppData\Local\Microsoft\Windows\Temporary%20Internet%20Files\AppData\Local\Microsoft\Windows\AppData\Local\Microsoft\BenSkinner\AppData\Local\Microsoft\Windows\Temporary%20Internet%20Files\Content.Outlook\D0XK5IJI\PD-PDLC04-EnvironmentScan_v0%204%201.doc#_Toc272477995); and * [summary and recommendations](file://\\syd-fs2\SA\Standards%20Development\Communications%20IT%20and%20e-Commerce%20Services\IT-014%20Health%20Informatics\SubCom\IT-014-09\Projects\AppData\AppData\Local\Microsoft\Windows\Temporary%20Internet%20Files\AppData\Local\Microsoft\Windows\AppData\Local\Microsoft\BenSkinner\AppData\Local\Microsoft\Windows\Temporary%20Internet%20Files\Content.Outlook\D0XK5IJI\PD-PDLC04-EnvironmentScan_v0%204%201.doc#_Toc272477996) |
| Comments | This is a narrative document component; although it may use concepts from the E‑health Interoperability Framework, e.g. interactions, roles or services, these are not expected to be modelled using a formal notation such as UML. |

* 1. Business scenarios and use cases

|  |  |
| --- | --- |
| Perspective | Conceptual |
| Viewpoint | Enterprise |
| Content | The purpose of the business scenarios is described in Clause .  Each business scenario within this component typically comprises—   * a description of the persona (a fictitious end user) to whom a scenario relates; * an identifier for the business scenario; * a narrative description of the user’s interactions with a system; * a structured breakdown (high-level steps) of the user’s interactions with a system; and * the use case(s) to which each step in the business scenario relates (where applicable).   The purpose of business use cases is described in Clause .  Each business use case within this component typically comprises—   * an identifier for the use case; * the name of the use case; * the goal of the use case; * the actors (roles) participating in the use case; * assumptions; * pre-conditions; * triggers; * the basic flow of events; * any alternate flows of events; * post-conditions; * the business requirement(s) to which the use case relates; and * the step(s) of the business scenario (or scenarios) to which the use case relates. |
| Comments | There may be a need for a separate document component to cover system use cases. This could be located in the conceptual computational cell in Figure 5. Both can be represented with UML use case notation, and system use cases can be further represented as UML sequence diagrams. |

* 1. Business requirements

|  |  |
| --- | --- |
| Perspective | Conceptual |
| Viewpoint | Enterprise |
| Content | The purpose of the business requirements document is to describe the business needs of stakeholders in terms of what the project intends to accomplish for end users.  This document has four major goals:   * To serve as a mechanism for feedback between project stakeholders and the project team. * To decompose the business needs into component parts. * To define the focus for the definition of detailed requirements. * To serve as a product validation check.   Business requirements are documented in narrative form and structured in terms of—   * description of community roles; and * listing of business requirements (narrative) by community roles. |

* 1. Concept of operations

|  |  |
| --- | --- |
| Perspective | Conceptual |
| Viewpoint | Enterprise |
| Content | The purpose of the concept of operations is to describe the high-level business processes and supporting architecture proposed for the implementation of an e‑health solution. The document identifies the ongoing benefits for stakeholders of the solution, and describes how the solution works, the possible implementations, and key policy and privacy considerations.  The concept of operations is typically a standalone specification in narrative form, comprising the following sections:   * Business problem statement. * Capabilities of the future state. * Benefits—a description of the benefits that could potentially be achieved. * Priorities and rationale. * Scope. * Conceptual architecture. * Roles and participants. * Services—this includes the function(s) to be performed. * Business processes * Service resilience—describes how key functions would be performed if one or more of the services are temporarily unavailable. * Phased implementation. * Governance and policy—issues to be resolved in relation to the conceptual architecture. * Privacy**—**issues relevant to the conceptual architecture. * Security controls. * Adoption model. * Change management. |
| Example | *Concept of Operations for Electronic Transfer of Prescription* 1.1 [Ref. 27] |
| Comments | The concept of operations is based on IEEE 1362-1998 (superseded). It is typically created at the early stages of a project in parallel with the business scenario and use cases. Although it may use concepts from the framework, e.g. interactions, roles and services, these are not expected to be modelled using a formal notation such as UML.  The Concept of Operations is not intended to be a living document. It reflects the thinking at the early stage of a project, and will be superseded by detailed analysis and design documentation. |

* 1. Enterprise context model

|  |  |
| --- | --- |
| Perspective | Conceptual |
| Viewpoint | Enterprise |
| Content | The purpose of the enterprise context model is to capture the positioning of an e‑health system within its business context. A formal representation is given in terms of one or more community models, and the enterprise objects/parties that play roles in the community, as described in Clause .  In particular, each community should be described in terms of the following concepts:   * Community objective. * Roles in the community. * Parties (and, more broadly, enterprise objects) that fulfil roles in the community. * Enterprise policies that apply to the roles. * Business services that are provided by the community, or within the community. * Business processes that describe behaviour (events, interactions, artefacts exchanged etc.) between community roles—   + within community (orchestration style of behaviour); or   + across communities (choreography style of interactions). |
| Example | *High-Level System Architecture, PCEHR System*, Version 1.35 [Ref. 31] |
| Comments | Modelling notations can be used to model behavioural aspects of a community, taking into account the approach detailed in ISO/ITU-T RM-ODP standards.  A community may include both human and system roles and can be used to scope the business problem, identify systems to be produced and define business requirements for the roles. Models can specify as-is and to-be situations and describe transition paths. |

* 1. Logical information specification

|  |  |
| --- | --- |
| Perspective | Logical |
| Viewpoint | Information |
| Content | The purpose of the logical information specification is to describe key information objects and their relationships in the form of an information model.  In general, a logical information specification consists of the following elements:   * Key information objects. * Information models, comprising information objects and their relationships. * State diagrams of information objects, showing their lifecycle. * Structured templates. * Abstract datatypes and structures. * Terminology bindings where applicable. |
| Example | *Event Summary Structured Content Specification*, Version 1.1**.** [Ref. 30] |
| Comments | Depending on the system, a logical information specification can describe clinical information, non-clinical information or both.  A clinical document specifies clinical information and documents from the logical perspective, covering elements described in Clause .  NOTE: NEHTA’s approach to publishing clinical information at the logical level is through the use of Structured Content Specifications.  Non-clinical specifications capture information that is outside of the realm of clinical informatics and/or terminology, and are used to describe information objects such as health identifiers, or information models of repositories. |

* 1. Logical service specification

|  |  |
| --- | --- |
| Perspective | Logical |
| Viewpoint | Computational |
| Content | The purpose of the logical service specification is to identify the logical system roles required to realize the roles identified in the enterprise context model. This specification identifies the behaviour of the roles and the required logical services to support the community.  A logical service specification typically consists of the following elements:   * *System roles*   This element should provide a succinct overview of the complete responsibilities of all of the roles. Each system role section will encapsulate—   * + one or more service interfaces;   + constraints applicable to roles interactions; and   + constraints to the systems fulfilling the roles. * *Service interface,* consisting of—   + service operation;   + pre-condition;   + post-condition;   + inputs, outputs and faults; and   + exception conditions * *Service interface correspondences* *to information objects*, including their state transitions (specified in the logical information model), in particular through inputs, outputs and faults. |
| Example | *PCEHR Document Exchange Service Logical Service Specification*, Version 1.2 [Ref. 34] |
| Comments | This component provides a platform independent specification of the systems and services to be implemented within the community.  This specification is further refined in the technical service specification—see Clause 5.10. |

* 1. Technical information specification

|  |  |
| --- | --- |
| Perspective | Implementable |
| Viewpoint | Information |
| Content | The purpose of the technical information specification is to provide further, implementable level of detail on the information artefacts specified in the logical information specification. This additional detail reflects the specifics of the platform and standard(s) chosen.  The following are the elements of a technical information specification:   * Background about the logical information specification. * List of platform and standards used (technology viewpoint). * Datatypes used (e.g. as in ISO 21090:2011). * Schemas for documents used (e.g. HL7 CDA). * Schemas for messages used (e.g. V2). * Specific terminologies/vocabularies used (e.g. SNOMED CT®). * Mappings to the logical information objects. * Australian extensions (if needed). |
| Example | National E-Health Transition Authority, *e-Referral CDA Implementation Guide,* Version 2.2 [Ref. 28] |
| Comments | One logical information specification may be instantiated using one or more technical implementation specifications. |

* 1. Technical service specification

|  |  |
| --- | --- |
| Perspective | Implementable |
| Viewpoint | Computational |
| Content | The purpose of the technical service specification is to specify the behaviour required from a set of interworking systems. This behaviour is specified in terms of a catalogue of related services that are provided and consumed by the interworking systems. Services are specified in terms of interface contracts.  Also identified is the explicit set of requirements to be satisfied by parties fulfilling identified roles within the service. These requirements are identified as conformance points.  This document component provides further, more detailed information on how to realize the capabilities specified in the logical service specification. It includes a set of mappings of the logical service specification on common behaviours and also specific statements for each role specification. The following are the elements of a technical service specification:   * List of platform and standards used (technology viewpoint) * Specific security functions and associated conformance points—   + identification and authentication (e.g. PKI);   + authorization (e.g. role based access control—RBAC)   + data integrity (digital signatures);   + data confidentiality;   + data provenance; and   + audit. * Transport protocol (e.g. SOAP, HL7 V2 message …) and associated conformance points. * Technical interfaces (instantiation of logical interfaces) and associated conformance points—   + system interfaces provided;   + system interface invoked;   + events accepted;   + events notified;   + event specific behaviour; and   + interface signatures. * References to related technical information specifications. |
| Example | *Electronic Transfer of Prescription - Technical Service Specification* V1.1 [Ref. 29]. |
| Comments | The technical service specification provides a platform-specific realization of the definition provided within a preceding logical service specification. While the logical and technical service specifications may be used together, the technical service specification needs to be able to stand alone. |

* 1. Conformance profile

|  |  |
| --- | --- |
| Perspective | Logical and implementable |
| Viewpoint | Technical |
| Content | The purpose of the conformance profile is to capture the key points of conformity within the logical service specification, technical service specification, logical information specification and technical information specification. The conformance profile precisely describes how an implementer is expected to demonstrate conformance against the specification.  The conformance profile includes—   * statements about how clinical information systems can conform to the specifications; and * a description of the conformance levels applicable to implemented solutions. |
| Example | *PCEHR Conformance Profile for Consumer Entered Information, Clinical Documents,* Version 1.3, 17 May 2012  <http://www.nehta.gov.au/implementation-resources/clinical-documents/EP-1094-2011/NEHTA-1263-2012> |
| Comments | Conformance profiles may need to consider both point-to-point and point-to-share communication models. |

* 1. Implementation and support material

|  |  |
| --- | --- |
| Perspective | Implementable |
| Viewpoint | Technology |
| Content | The implementation and support material describes how to adopt, implement and operate the solution and its component parts in a specific customer or implementation context.  It consists of the information required to successfully adopt or implement the solution specified. Examples of implementation and support material are as follows:   * Context. * Reference to related specifications. * Constraints, including information security and dependability constraints, platform specific constraints and rules. * Assumptions. * Prerequisites. * Release information. * Implementation-specific requirements. * Specific implementation guides. * Business process guides. * Privacy and consent considerations. * Legislative requirements. * System sizing, capacity and scalability requirements. * Testing material. * Compliance, conformance and accreditation criteria. * Service management guides and documentation, including guides to service level agreements. * Maturity planning and assessment guides. |
| Comments | The composition of implementation material may vary greatly depending on the associated solution. The important consideration is that sufficient information is provided to support all parties impacted by implementation of the specification, whether they are service providers or consumers. |

1. Interoperability capability
   1. Approach
      1. Introduction

Interoperability capability can be expressed using concepts from the well-established and recently updated CMMI® (Capability Maturity Model® Integration) Framework architecture [Ref. 42] This is tailored for interoperability in the form of capability *levels* for interoperability, work *products* for interoperability, *process areas* for interoperability, and specific *goals* for interoperability (the terms in italics in Clauses 6.1.1 to 6.1.5 are CMMI concepts).

* 1. In CMMI, capability levels are also expressed in terms of reaching generic goals, which apply across all CMMI applications; i.e, generic goals 1, 2, and 3 are related to capability levels 1, 2 and 3 respectively.
  2. This Handbook uses a simpler form of CMMI representation, called a ‘continuous CMMI representation’, in which improvements are stated in terms of capability levels rather than maturity levels; the latter can be useful for benchmarking, but their development requires significant standardization and is typically done at later stages of improvement processes, as required.
     1. Capability levels for interoperability

Capability *levels* for interoperability reflect how well organizations perform processes related to the delivery of interoperability outcomes (termed ‘work products’ in CMMI [Ref. 42]). These levels are the following:

* A *performed* (capability level 1) process accomplishes the work needed to produce interoperability work products and, in doing so, satisfies specific interoperability goals.
* A *managed* (capability level 2) process—
  + is planned and executed in accordance with policy;
  + involves relevant stakeholders; and
  + is monitored, controlled and reviewed.

NOTE: Managed processes help to ensure that realization of the specific interoperability outcomes is retained during times of stress.

* A *defined* (capability level 3) process is a managed process which is tailored from an organization’s set of standard processes according to the organization’s tailoring guidelines, and contributes process experience to the organization’s process assets. Defined processes help ensure more consistency of interoperability practices across (particularly large) organizations, and they are described more rigorously than those at level 2.
  + 1. Work products for interoperability

Work *products* for interoperability may be—

* specifications and standards (produced by standards organizations, and other organizations such as national authorities);
* e‑health solutions (produced by vendors and system integrators), often based on standards and specifications; or
* e‑health services and processes (which involve participation by health providers, users and vendors to provide the required solutions).
  + 1. Process areas for interoperability

A *process area* for interoperability is a cluster of related *practices* that, when implemented collectively, satisfy a set of specific interoperability goals and that are considered important for making improvement in that area. The degree to which these interoperability goals is realized indicates how well a particular interoperability process area is supported.

For example, service orientation is an important process area for interoperability, and service interface, service interface, service agreement, service identification and service composition.

Another example of an interoperability process area is common semantics, with a number of interoperability-specific goals such as information semantics, service semantics, process semantics and policy semantics.

* + 1. Specific goals for interoperability

A specific *goal* for interoperability is a required model component that describes unique characteristics required to satisfy the process area—for example, service interface is a specific goal for the service orientation process area.

* 1. Governance—overarching concern

The establishment of policies, processes, measurement and other activities is an important part of the CMMI methodology; hence the use of the CMMI Framework [Ref. 42] to support improvements in interoperability—that is, managed and defined capability levels—can be regarded as a mechanism for governing interoperability.

Governance establishes rules that control decision-making. When applied to interoperability, governance defines organization-specific policies, standards and roles, in order to achieve a jointly negotiated shared purpose between multiple organizations—as described in the SAIF Governance Framework [Ref. 20]. This shared purpose is to be realized through a technical solution that is specified using concepts from other viewpoints. Governance is therefore an overarching concern in the enterprise viewpoint, but is also of relevance for other viewpoints.

It is important to note that governance concerns include both the governance of organizational resources and the governance of technology components. For example, the governance that occurs at a computational interface via constructs such as pre-conditions, post conditions, contracts, roles and accountabilities is, in fact, a technical realization of an agreement between two or more parties to achieve a shared purpose. Such an agreement would need to define clearly the responsibilities, expectations and response to non-performance, which are the basic content of a contract [Ref. 20].

The establishment of interoperability governance thus ensures a controlled approach to performing required processes to support delivery of interoperable outcomes. These outcomes can be either—

1. specifications produced by standards development organizations or other appropriate organizations; or
2. interworking between diverse healthcare organizations and/or diverse systems using underlying e‑health services and infrastructure.

The essential governance relationship between process and outcomes is developed further in—

1. Clause , which identifies a set of process areas‑ and the specific interoperability principles that are supported by each process area; and
2. Clause , which identifies specific goals, practices and work products that may be considered when assessing the achievements of interoperability outcomes for each of the process areas.

The process areas, goals and outcomes in Clauses 6.3 and 6.4 represent a model that may be extended or partly replaced in any particular application of the E-Health Interoperability Framework; however, as with any approach to quality management, the specific aspects being assessed in any particular application need to be maintained and documented in a controlled fashion.

* 1. Interoperability process areas

The interoperability principles described in Section 2 are useful for helping to identify key process areas for interoperability. See Clause 2.3.

* + 1. Governance

The governance process area contains organizational practices for establishing and evolving policies, processes and standards that support the identification of shared purposes and the expression of these purposes in well-formed specifications that enable interoperable e‑health solutions to be effectively achieved and maintained. This process area encompasses the following principles:

1. Governance of change.
2. Conformance and compliance.

NOTE: Governance is both an overarching concern (as described in Clause 6.2) and a specific interoperability process area.

* + 1. Technical interoperability

The technical interoperability process area contains practices for realizing e‑health solutions that support health information exchange using different technologies.

1. See for approaches that have influenced the elements in this process area.

This process area reflects the following principles:

1. Universal participation.
2. Enabling interoperability.
3. Observance of standards.
4. Agreement on common semantics.
5. Conformance and compliance.
   * 1. Policy compliance

The policy compliance process area contains practices for realizing e‑health solutions that comply with jurisdictional/organizational policies. This process area reflects the following principles:

1. Policy compliance.
2. Resolution of policy conflicts.
3. Separation of business rules.
   * 1. Common semantics

The common semantics process area contains practices for supporting shared understanding of technical and organizational concerns.

This process area reflects the following principle: ‘Agreement on common semantics’.

* + 1. Conformity assurance

The conformity assurance process area contains practices for checking the conformance of implementations with specifications. This process area reflects the following principles:

(a) Conformance and compliance.

(b) Observance of standards.

* + 1. Service orientation

The service orientation process area contains practices for supporting the building of service oriented solutions. This process area reflects the following principle:

Support services-based approach.

* + 1. Collaboration and stakeholders

The collaboration and stakeholders process area contains practices that support collaboration among users and vendors across the continuum of technology design, including the embedding of products into clinical workflow and ongoing product optimization [Ref. 24]. This process reflects the following principles:

1. Stakeholder engagement.
2. Universal participation.
   1. Specific interoperability goals

Table 1 lists specific interoperability goals for each of the interoperability process areas. The table also lists related specific practices to support the achievement of the interoperability goals, and provides examples of work products. Some of the work products may be the recommended document components identified in Section .

NOTE: The process areas and specific goals are numbered for ease of referencing. The numbers do not reflect relative maturity between specific goals or practices; this is consistent with the CMMI approach.

TABLE 1

INTEROPERABILITY PROCESS AREAS AND GOALS

| Process Area (PA) | Specific Goals (SG) | Specific Practices (SP) | Indicative Work Products (WP) |
| --- | --- | --- | --- |
| 1. Governance | * 1. An established governance/management system | * + 1. Establishment of organizational policies     2. Implementation of processes     3. Measurement of process effectiveness | * + - 1. Managed policies and processes |
|  | * 1. Effective governance of change | * + 1. Systematic approach to change     2. Management of change within defined timeframes | * + - 1. Effective adoption of changes |
|  | * 1. An established and maintained risk management system | * + 1. Systematic approach to risk assessment and review of clinical and business requirements | * + - 1. Business requirements and specifications that include controls aligned with identified business risks. |
| 1. Technical interoperability | * 1. Machine transportable data (transport of unstructured data, such as fax, scanned pictures and documents, Walker level 2; see Clause D2) | * + 1. Using basic communication infrastructure | * + - 1. Regular process for communication of information, such as scanned laboratory reports. |
|  | * 1. Basic electronic data interchange (exchange of structured yet non-standardized messages or documents, Walker level 3, syntactic; see Clause D2; see also Appendix E. | * + 1. Establishment of basic communication protocol     2. Using the communication protocol | * + - 1. Exchange of basic electronic communications, e.g. PDF attached to email, using organization’s customized document layout, or email containing free text. |
|  | * 1. Semantic interoperability (exchange of information using standardized data structures, Walker level 4, semantic)   NOTE: Semantic interoperability requires common semantics (PA4). | * + 1. Establishment of communication protocol and information semantics     2. Using the communication protocol and information semantics | * + - 1. Exchange of messages containing standardized content (e.g. exchange of HL7 CDA documents) |
| 1. Policy compliance | * 1. Compliance with e‑health legislation and regulations | * + 1. Review of relevant regulations/legislations     2. Identification of applicable policies | * + - 1. Demonstrated compliance in all organizational systems       2. Authentication and access controls |
| 1. Common semantics | * 1. Information semantics | * + 1. Establishment of a shared meaning of healthcare information (e.g. SNOMED CT)     2. Establishment of rules for describing dependencies across clinical concepts     3. Specification of e-health systems using information semantics (e.g. ISO/IEC 11179) | * + - 1. Reference information models (HL7 RIM, openEHR RM)       2. Clinical information ontology       3. Standard data structures (e.g. HL7 V2, CDA) |
|  | * 1. Service semantics | * + 1. Identification and adoption of key elements of service meta-model     2. Specification of systems in terms of the service meta-model     3. Establishment of rules for describing dependencies across services and service concepts | * + - 1. Agreed service meta-model (e.g. in UML)       2. Service ontology (e.g. OWL)       3. Service catalogue |
|  | * 1. Process semantics | * + 1. Identification and adoption of key elements of a process meta-model (e.g. BPMN2)     2. Specification of systems in terms of the process meta-model | * + - 1. Agreed process meta-model |
|  | * 1. Policy semantics | * + 1. Identification and adoption of key elements of a policy meta-model (e.g. ODP Enterprise Language)     2. Specification of policy constraints in terms of this policy meta-model | * + - 1. Agreed policy concepts |
| 1. Conformity assurance | * 1. Requirements management | * + 1. Eliciting requirements     2. Obtaining commitment to requirements     3. Managing requirements changes     4. Maintaining traceability of requirements     5. Ensuring alignment between project work and requirements | * + - 1. Documented requirements and traceability |
|  | * 1. Risk management | * + 1. Clinical safety assessment     2. Security assessment     3. Privacy assessment | * + - 1. Clinical safety case report       2. Security threat and risk assessment       3. Privacy impact assessment |
|  | * 1. Verification and validation | * + 1. Product verification process     2. Product validation process | * + - 1. Product verification report       2. Product validation report |
|  | * 1. Declaration of conformance | * + 1. Demonstration of validated product | * + - 1. Third party testing by test labs completed |
| 1. Service orientation (based on SG4.2) | * 1. Service interface | * + 1. Adhering to service interface template | * + - 1. Service interface specification |
|  | * 1. Service agreement | * + 1. Standardized service negotiation | * + - 1. Service level agreement |
|  | * 1. Service identification | * + 1. SP1: Implementation of service catalogue | * + - 1. Service catalogue |
|  | * 1. Service composition | * + 1. Service orchestration | * + - 1. Workflow support, e.g. collaborative care service pattern |
| 1. Collaboration and stakeholders | * 1. Engage with stakeholders | * + 1. Engagement and consultation in deriving e‑health specifications | * + - 1. Consensus reflected in e‑health specifications e.g. for policy authorities, SMEs |
|  | * 1. Universal participation | * + 1. Participation in development of and contribution to e‑health specifications | * + - 1. Interoperable e‑health systems       2. Vendors, consumers, health industry participants |



Background information

(Informative)

* 1. National E‑Health Strategy

The E‑health Interoperability Framework is specifically aimed at satisfying a number of national e‑health expectations, as identified in the *National E‑Health Strategy* [Ref. 18]. These expectations include—

1. national alignment and coordination of e-health solutions;
2. support for the co-existence of national and local solutions;
3. a balance between needs for regulation and competition/innovation; and
4. the adoption of the best architecture and software engineering practices.

These expectations are manifest in the following broad set of interoperability requirements:

1. Explicit representation of health-specific legislative, regulatory and organizational policies as part of the design, implementation and operation of e‑health systems. This is required to support building cross-jurisdictional and cross-organizational systems of national significance.
2. Precise modelling of the clinical information and terminology concepts, striking a balance between the needs of clinical and administrative domain experts and the needs of rigorous information modelling.
3. Support for a nationally agreed approach to conformity assessment.
4. Sound clinical practice.
5. The facilitation of sustainable e‑health interoperability by leveraging relevant national and international standards while supporting changes that reflect new business practices or new technologies.
   1. Standards-based approach

The E‑health Interoperability Framework in SA HB 137—2013 is based on accepted national and international standards to ensure its wide use, support and evolution; having this base will increase the likelihood of its success as a reference framework supporting sustainable and interoperable e‑health outcomes at the national level. These outcomes should be reflected in a set of agreements for standard-based interoperability, covering both technical and business (i.e. health ) aspects.

The Framework should support the implementation of many specific e‑health architectures, which, if compliant with the Framework, will contribute towards increasing the maturity of national e‑health solutions in relation to their interoperability. This will ultimately deliver better quality, safer and more reliable healthcare outcomes while promoting competitive and innovative e‑health solutions—solutions that support clinical delivery and adopt the latest clinical research results.

In brief, a standards-based framework supports a shared understanding of, and a common language for describing, policy, clinical and IT issues among the different stakeholders involved in the specification, build, implementation and evolution of e‑health systems. These characteristics are in line with the *Australian Government Architecture Reference Models* [Ref. 16] and the Australian Government Interoperability Framework [Ref. 17].

* 1. Support for diverse stakeholders

The E‑health Interoperability Framework needs to support many types of stakeholders, and should therefore accommodate the separation of concerns typically achieved by adopting architecture viewpoints. The different types of stakeholders include national regulatory and policy authorities, health services (private and public), business decision makers, clinicians, enterprise and solution architects, and standards architects and analysts, as well as users and developers. Each of these stakeholder groups has its own concerns and languages, and the framework should identify key concepts of relevance to these stakeholder groups.

* 1. Interoperability and semantics
     1. Overview

This Framework should facilitate traceability from requirements to specification to implementation and deployment. As a result, the Framework requires well-defined expressions of concepts and their meaning (ISO 17115:2007—that is, their semantics)—for both specification and implementation artefacts. This would then support the system, and software developers (e.g. software vendors, system integration houses and jurisdictional IT system departments) will be able to leverage the increasing capability of tools that link architecture development, software design and development processes. This precision is also important for those involved in procuring e‑health systems who need clear requirements specifications

* + 1. Business semantics

One aspect of these semantics is the ability to describe succinctly the business (i.e. health-specific) environment in which IT systems are to be deployed—in particular the roles, business services, business processes and business policies. A common grammar to express these business concerns is needed to facilitate business interoperability aspects among end users and designers across the e‑health sector. This in turn will facilitate architecting and designing fit-for-purpose e‑health systems.

* + 1. Data semantics

Another aspect of these semantics is the semantics of information objects that are used or referenced by business processes, business policies and business interactions. These information objects need to express clinical, administrative and research aspects of health systems.

The interoperability framework should provide guidance on the use of information models—structures that describe the information to collect—and the terminology—the values that are stored in the data structure. Often it is possible to combine data elements into a single statement from the terminology, and resolving these issues is currently the subject of ‘terminology binding’ research efforts.

* + 1. Technical architecture semantics

The third aspect concerns distributed system components and interactions and their explicit service definitions, infrastructure mechanisms and technology choices. Again, a common grammar should be defined to express these contracts, infrastructures and technologies.

* 1. Formalization approach

The concepts identified in the framework should be sufficiently precise to support the use of software tools to develop, test and deploy e‑health solutions, and should leverage the concepts of the established industry and e‑health approaches. These include—

1. UML2.3 semantics;
2. clinical archetypes, such as those defined in openEHR;
3. the associated UML profiles, such as UML profile for ODP (ISO/IEC 19793), OMG Business Motivation Model [Ref. 35], and SoaML;
4. the UML representation of HL7 datatypes or clinical archetypes; and
5. other languages natively, such as HL7 RIM concepts and clinical archetypes.

The concepts from the framework can then be formalized in a meta-model, which can be implemented using existing commercial UML-based tools, with appropriate plug-ins for the profiles in item (c) above, or the use of a more fundamental and UML-independent way of expressing the meta-model—for example, the use of the Meta-Object-Facility (MOF) based expression of meta-models, which would provide better integration with the HL7 style of expression, being conceptually different to that of UML representation.

In addition, a further level of formalization can be achieved through the adoption of approaches that are based on formal logics—for example, ontologies represented in the OWL language.

This approach provides strong support for the development or implementation of e-health systems/solutions, by supporting—

1. traceability from requirements, through design and architecture, to implementation and testing;
2. the establishment of an explicit conformance approach to support vendor implementations against conformance points in the specifications;
3. sustained interoperability through facilitating an incremental approach to supporting additional capabilities, based on clinical requirements or availability of new technologies; and
4. tool-based documentation and manipulation, as well as implementation of downstream components and systems. For example, many tools have been developed to support UML-based development and, where necessary, model transformation. The latter is particularly important in an environment that has multiple standards for the artefacts, such as message types, datatypes and terminologies.
   1. Sustainable e‑health interoperability

An interoperability framework should explicitly support two facets of interoperability:

1. Interoperability of different solutions at a certain point in time, i.e. ensuring the coexistence of different solutions.
2. Interoperability over different points in time, i.e. ensuring the evolutionary aspect of interoperability.

The latter will typically reflect organizations’ increasing levels of interoperability maturity, in terms of both organizational processes and the technologies available.

The need to support the above two facets of interoperability is driven by the need to deal with change, and this change can cover a number of facets, such as changing requirements, semantics, knowledge, technology and business processes.

In order to promote sustainable e‑health interoperability, it is also desirable to describe the underlying e‑health support in a technology-independent way, to ensure support for new technologies as they emerge. For example, supporting different interaction options for communication between e‑health participants, ranging from document-oriented to message-oriented to service-oriented and even event-driven options for a particular e‑health solution, such as a discharge or referral solution.

Governance in an interoperable environment is critical for success.



Example—use of an interoperability framework

(Informative)

An example of the use of an e-health interoperability framework by an organization is illustrated in Figure 6, which shows the solution framework for the National E-Health Transition Authority (NEHTA).



Figure 6: E-health frameworks landscape

NEHTA’s solution framework was developed to deliver consistent NEHTA solution specifications. The solutions specifications include templates for conceptual, logical and implementable specifications, which were used to structure and specify different solution components of the Personally Controlled Electronic Health Record (architecture).

NEHTA’s internal solution framework—

1. refines the framework (described in Section ) with a number of architecture decisions adopted by NEHTA, in particular in terms of the modelling concepts used in the enterprise and information viewpoints and support for conformance points;
2. is centred around the use of a software tool for modelling requirements, architecture and implementation artefacts—based on UML. This tool allows traceability from requirements to architecture artefacts and is used in part to support the generation of document specifications; and
3. is supported by the specification development methodology adopted by NEHTA and aligns with the governance framework for specification delivery within NEHTA.

The E-health Interoperability Framework can also be used to facilitate the description of integration architectures between national e-health systems and jurisdictional or local systems (such as those used in GP practices). This is indicated by the red and blue boxes in Figure 6.



Developing a service taxonomy

(Informative)

This Handbook provides a structured way of representing concepts and their relationships as parts of an e-health specification. It also allows for the inclusion of the specification of constituent information artefacts, business processes, business policies and services. Not only can the Framework be used for building e-health specifications, it can also provide a classification scheme for design and implementation artefacts, leading to more sophisticated ontologies.

An earlier framework, the *Interoperability Framework* 2.0 [Ref. 32], identified a number of interoperability patterns such as common business processes, common policies of relevance for e-health, and common interaction patterns. These contribute towards a common knowledge base of value for e-health designers and developers, and it is anticipated that these patterns will continue to grow in quantity and quality to reflect community involvement.

There is an increasing demand to develop a classification of services in e-health, and the E-health Interoperability Framework provides a way of classifying services according to viewpoints and design abstractions. This approach is also aligned with the approach being adopted as part of the HSSP Service Functional Model guide [Ref. 21].

Such a classification of services could include both—

1. high level business services, such as e-discharge, e-referrals, care plans and electronic transfer of prescriptions; and
2. technical services, such as getting information from repositories of different kinds, storing information there, controlling access, etc.

This classification should reflect the specific Australian e-health environment and could take into account contributions from other, similar efforts, such as that of the US National Cancer Institute[[16]](#footnote-16)

The classification rules and approach could thus reflect the E-health Interoperability Framework, and the classification itself is anticipated to be developed as part of the collaborative efforts of relevant stakeholders in the Australian e-health community—for example, healthcare organizations, vendors and standards organizations.



Technical interoperability capability and existing approaches

(Informative)

* 1. General

Two early approaches to analysing and defining technical capability are offered in Walker [Ref. 45] and the *Extending the* *Levels of Conceptual Interoperability Model* (LCIM) [Ref. 43]. These approaches informed the structure of the technical capability level in Table 1.

* 1. Walker’s taxonomy

Walker’s four-level taxonomy [Ref. 45] developed for the health sector categorizes the exchange of healthcare information and interoperability in terms of level of sophistication and standardization. The four levels are the following:

* *Level 1* Non-electronic data—no use of IT to share information (e.g. mail, telephone).
* *Level 2* Machine-transportable data—transmission of non-standardized information via basic IT; information within the document cannot be electronically manipulated (e.g. fax or personal computer-based exchange of scanned documents, pictures, or PDF files).
* *Level 3* Machine-organizable data—transmission of structured messages containing non-standardized data; requires interfaces that can translate incoming data from the sending organization’s vocabulary into the receiving organization’s vocabulary. This usually results in imperfect translations because of vocabularies’ incompatible levels of detail (e.g. email of free text, or PC-based exchange of files in incompatible/proprietary file formats, HL7 messages).
* *Level 4* Machine-interpretable data—transmission of structured messages containing standardized and coded data; idealized state in which all systems exchange information using the same formats and vocabularies (e.g. automated exchange of coded results from an external lab into a provider’s electronic medical record, or automated exchange of a patient’s ‘problem list’).
  1. Levels of Conceptual Interoperability Model

The LCIM [Ref. 43], developed for the purpose of modelling and simulation, has seven levels of interoperability:

1. *No interoperability—*no interoperability exists.
2. *Technical interoperability*—a communication protocol exists for the exchange of information.
3. *Syntactic interoperability*—a shared syntax or data format exists for the exchange of information.
4. *Semantic interoperability*—a shared understanding exists of the content of the information, based on a common information exchange reference model.
5. *Pragmatic interoperability*—a shared understanding exists of the context in which the information was used or produced.
6. *Dynamic interoperability*—systems are able to comprehend state changes and the effect of these changes.
7. *Conceptual interoperability*—a fully specified but implementation independent model exists.

CDA CONFORMANCE LEVELS

(Informative)

The levels of Clinical Document Architecture (CDA) conformance are defined in Table 2. These levels specify the encoding, structure and semantics of clinical documents exchanged between health software systems

Table 2 is sourced from the *Common Conformance Profile for Clinical Documents*, Version 1.3, Table 3.4 [Ref. 26].

TABLE 2

LEVELS OF CLINICAL DOCUMENT ARCHITECTURE (CDA) CONFORMANCE

|  |  |
| --- | --- |
| CDA Level | Minimum conformance requirements |
| **1A** | With level 1A conformance a clinical document **shall** consist of:   1. A CDA body in XML format; and 2. A CDA body that only includes attachment references in the narrative block. |
| **1B** | With level 1B conformance a clinical document **shall** consist of:   1. A CDA body in XML format; and 2. A CDA body that includes at least one section which contains a narrative block. |
| **2** | With level 2 conformance a clinical document **shall** consist of:   1. A CDA header in XML format; 2. A CDA body in XML format; 3. A CDA body that contains mandatory sections; and 4. Mandatory sections, each containing a section label and a narrative block. |
| **3A** | With level 3A conformance a clinical document **shall** consist of:   1. A CDA header in XML format; 2. A CDA body in XML format; 3. A CDA body that contains mandatory sections; 4. Mandatory sections, each containing a section label, a narrative block and encoded content for mandatory elements. |
| **3B** | With level 3B conformance a clinical document **shall** consist of:   1. A CDA header in XML format; and 2. A CDA body in XML format; and 3. A CDA body that contains mandatory sections; 4. Mandatory sections, each containing a section label, a narrative block and encoded content for mandatory elements; and 5. Specified terminologies. |

1. The verb **shall** when appearing in a conformance requirement indicates a mandatory requirement..

Bibliography

1. AS 3806—2006 Compliance programs. Available at <http://infostore.saiglobal.com/store/details.aspx?ProductID=304437>

1. AS ISO/IEC 17000—2005 Conformity assessment—Vocabulary and general principles. Available at <https://infostore.saiglobal.com/store/Details.aspx?ProductID=344787>

3. ATS 5822—2010 E-health secure message delivery Available at <http://infostore.saiglobal.com/store/Details.aspx?ProductID=1391035>

4. SA HB 138—2013 E-health Architecture Principles.

5. SA TR 5823—2010 Endpoint location service. Available at <http://infostore.saiglobal.com/store/details.aspx?ProductID=1391037>

6. ISO 1087-1:2000, Terminology work—Vocabulary—Part 1: Theory and application. Available at <http://www.iso.org/iso/catalogue_detail.htm?csnumber=20057>

7. ISO 11179 Information technology—Metadata registries. Available at <http://metadata-standards.org/11179/>

8. ISO 13606-1:2008, Health informatics—Electronic health record communication, Part 1: Reference model. Available at <http://www.iso.org/iso/home/store/catalogue_tc/catalogue_detail.htm?csnumber=40784>

9. ISO 17115:2007, Health informatics - Vocabulary for terminological systems. Available at <http://www.iso.org/iso/home/store/catalogue_tc/catalogue_detail.htm?csnumber=32881>

10. ISO 21090:2011 Health informatics—Harmonized data types for information interchange. Available at <http://www.iso.org/iso/home/store/catalogue_tc/catalogue_detail.htm?csnumber=35646>

11. ISO/IEC 11404:2007, Information technology—General-Purpose Datatypes (GPD). Available at <http://www.iso.org/iso/home/store/catalogue_tc/catalogue_detail.htm?csnumber=39479>

12. ISO/IEC 14771, Information technology—Open distributed processing—Naming framework, ITU-T Recommendation X.910. Available at <http://www.itu.int/rec/T-REC-X.910-199809-I/en> or <http://www.iso.org/iso/home/store/catalogue_tc/catalogue_detail.htm?csnumber=25507>

13. ISO/IEC 17007:2009 Conformity assessment— Guidance for drafting normative documents suitable for use for conformity assessment, 2009. Available at <http://www.iso.org/iso/home/store/catalogue_tc/catalogue_detail.htm?csnumber=42635>

14. ISO/IEC GUIDE 67:2004, Conformity Assessment—Fundamentals of product certification. Available at <http://www.iso.org/iso/home/store/catalogue_tc/catalogue_detail.htm?csnumber=30258>

15. ISO/TR 16056-1:2004, Health informatics—Interoperability of telehealth systems and networks—Part 1: Introduction and definitions. Available at <http://www.iso.org/iso/home/store/catalogue_tc/catalogue_detail.htm?csnumber=37351>

16. Australian Government, *Australian Government Architecture Reference Models*, Version 3.0, August 2011 [viewed 8 January 2013]. Available at <http://agimo.gov.au/files/2012/04/AGA_RM_v3_0.pdf>

17. Australian Government, *Australian Government Interoperability Framework* [viewed 17 October 2012]. Available

<http://www.finance.gov.au/publications/australian-government-information-interoperability-framework/docs/Information_Interoperability_Framework.pdf>

18. Australian Health Ministers’ Conference, *National E-Health Strategy*, December 2008 [viewed 17 October 2012]. Available at http://www.health.gov.au/internet/main/publishing.nsf/content/national+Ehealth+strategy

19. GridWise Architecture Council with Harbor Research, Inc., *Financial Benefits of Interoperability: How Interoperability in the Electric Power Industry Will Benefit Stakeholders Financially*, September 2009 [viewed 17 October 2012]. Available at [www.gridwiseac.org/pdfs/financial\_interoperability.pdf](http://www.gridwiseac.org/pdfs/financial_interoperability.pdf)

20. Health Level Seven Inc, *Service-Aware Interoperability Framework*—Canonical Definition, Normative Draft Standard for Trial Use Ballot 1, January 2012. Available at <http://www.hl7.org/documentcenter/public/standards/dstu/SAIF_CANON_DSTU_R2_2012MAY.pdf>

21. Health Level Seven/OMG Healthcare Services Specification Project [viewed 17 October 2012]. Available at [hssp.wikispaces.com/](http://hssp.wikispaces.com/)

22. Health Level Seven EHR Interoperability Work Group, *Coming to Terms: Scoping Interoperability for Health Care*, 7 February 2007 [viewed 8 January 2013]. Available at <http://www.hln.com/assets/pdf/Coming-to-Terms-February-2007.pdf>

23. IEEE Standards Association, IEEE Std 1362-1998 IEEE, Guide for Information Technology—System Definition—Concept of Operations (ConOps) Document [viewed 17 October 2012]. Available at [standards.ieee.org/reading/ieee/std\_public/description/se/1362-1998\_desc.html](http://standards.ieee.org/reading/ieee/std_public/description/se/1362-1998_desc.html)

24. Institute of Medicine of the National Academies*, Health IT and Patient Safety: Building Safer Systems for Better Care*, 2012, The National Academies Press, Washington DC.

25. ITU-T Rec. X.906 | ISO/IEC 19793:2008 Information technology—Open distributed processing—Use of UML for ODP system specifications. Available at <http://www.itu.int/rec/T-REC-X.906-200711-I/en> or <http://www.iso.org/iso/home/store/catalogue_tc/catalogue_detail.htm?csnumber=52089>

26. National E-Health Transition Authority, *Common Conformance Profile for Clinical Documents*, Version 1.3, 12 May 2012 [viewed 14 May 2013] Available at <http://nehta.gov.au/implementation-resources/clinical-documents/EP-1094-2011/NEHTA-1263-2012>

27. National E-Health Transition Authority, *Concept of Operations for Electronic Transfer of Prescription 1.1*, 17 December 2010 [viewed 18 March 2013]. Available at <http://nehta.gov.au/implementation-resources/clinical-documents/EP-0938-2010/NEHTA-0942-2010>

28. National E-Health Transition Authority, *e-Referral CDA Implementation Guide,* Version 2.2, 7 March 2012 [viewed 18 March 2013]. Available at <http://nehta.gov.au/implementation-resources/clinical-documents/EP-0936-2012/NEHTA-0967-2012>

29. National E-Health Transition Authority, *Electronic Transfer of Prescription - Technical Service Specification* V1.1, 17 December 2010 [viewed 14 May 2013].Available at <http://nehta.gov.au/implementation-resources/clinical-documents/EP-0938-2010/NEHTA-0987-2010>

30. National E-Health Transition Authority, *Event Summary Structured Content Specification*, Version 1.1, 30 November 2011 [viewed 14 May 2013]. Available at http://www.nehta.gov.au/implementation-resources/clinical-documents/EP-0939-2012/NEHTA-0995-2011

31. National E-Health Transition Authority, *High-Level System Architecture, PCEHR System*, Version 1.35, 11 November 2011 [viewed 14 May 2013]. Available at https://www.nehta.gov.au/implementation-resources/ehealth-foundations/EP-1001-2011/NEHTA-1002-2011

32. National E-Health Transition Authority, *Interoperability Framework 2.0*, August 2007 [viewed 14 May 2013]. Available at https://www.nehta.gov.au/implementation-resources/ehealth-foundations/EP-1144-2007/NEHTA-1146-2007

33. National E-Health Transition Authority, *PCEHR Conformance Profile for Consumer Entered Information, Clinical Documents*, Version 1.01, 21 December 2011 [viewed 14 May 2013]. Available at http://www.nehta.gov.au/implementation-resources/clinical-documents/EP-1047-2011/NEHTA-1046-2011

34. National E-Health Transition Authority, PCEHR Document Exchange Service Logical Service Specification, Version 1.2, 6 September 2012 [viewed 18 March 2013]. Available at [https://vendors.nehta.gov.au/sections/myLearning/index.cfm?action=activitySearch&sortBy=title&filterByDescription=document%20exchange&filterByCategoryId=0&filterByActivityTypeId=0#](https://vendors.nehta.gov.au/sections/myLearning/index.cfm?action=activitySearch&sortBy=title&filterByDescription=document%20exchange&filterByCategoryId=0&filterByActivityTypeId=0)

35. OMG, Business Motivation Model (BMM) [viewed 17 October 2012]. Available at [www.omg.org/spec/BMM/](http://www.omg.org/spec/BMM/)

36. OMG MDA Guide Version 1.0.1, omg/2003-06-01, 12 June 2003 [viewed 17 October 2012]. Available at [www.enterprise-architecture.info/Images/MDA/MDA Guide v1-0-1.pdf](http://www.enterprise-architecture.info/Images/MDA/MDA%20Guide%20v1-0-1.pdf)

37. OMG, OMG’s MetaObject Facility [viewed 17 October 2012]. Available at [www.omg.org/mof/](http://www.omg.org/mof/)

38. OMG, Service Oriented Architecture Modeling Language (SoaML) [viewed 17 October 2012]. Available at [www.omg.org/spec/SoaML/](http://www.omg.org/spec/SoaML/)

39. OMG, UML 2.3 [viewed 17 October 2012]. Available at [www.omg.org/spec/UML/2.3/](http://www.omg.org/spec/UML/2.3/)

40. openEHR [viewed 18 March 2013]. Available <http://www.openehr.org/home>

41. RM-ODP, Reference Model of Open Distributed Processing (RM-ODP, ITU-T Rec. X.901-X.904 | ISO/IEC 10746) [viewed 17 October 2012]. Available at <http://www.rm-odp.net/>

42. Software Engineering Institute, Carnegie Mellon, CMMI Version 1.3, 2010 [viewed 17 October 2012]. Available at [[http://www.sei.cmu.edu/reports/07tn009.pd](http://www.sei.cmu.edu/reports/07tn009.pdf)f](http://www.sei.cmu.edu/cmmi/solutions/info-center.cfm)

43. Turnitsa, C.D., 2005, ‘Extending the Levels of Conceptual Interoperability Model’, Paper presented at IEEE Summer Computer Simulation Conference, IEEE CS Press.

44. W3C, OWL 2 Web Ontology Language Document Overview [viewed 17 October 2012]. Available at [www.w3.org/TR/owl2-overview/](http://www.w3.org/TR/owl2-overview/)

45. Walker J, et al. 2005, *The Value of Health Care Information Exchange and Interoperability*, 19 January 2005, Center for Information Technology Leadership, Partners HealthCare System, Boston, MA, USA. Email: [Jwalker@citl.org](mailto:Jwalker@citl.org)

1. Reference Model of Open Distributed Processing (RM-ODP, ITU-T Rec. X.901-X.904, ISO/IEC 10746)[viewed 17 October 2012]. Available at <http://www.rm-odp.net/> [↑](#footnote-ref-1)
2. HL7 Services-Aware Interoperability Framework—Canonical Definition, Normative Draft Standard for Trial Use Ballot 1, January 2012. Available at <http://wiki.hl7.org/index.php?title=Product_SAIF> [↑](#footnote-ref-2)
3. NEHTA, Interoperability Framework, version 2.0, August 2007 [viewed 17 October 2012]. Available at <http://nehta.gov.au/implementation-resources/ehealth-foundations/EP-1144-2007> [↑](#footnote-ref-3)
4. Australian Health Ministers’ Conference, National E-Health Strategy, December 2008 [viewed 13 March 2013]. Available at <http://www.health.qld.gov.au/ehealth/docs/nat_ehlth_strat_1208.pdf> [↑](#footnote-ref-4)
5. National Research Council. Health IT and Patient Safety: Building Safer Systems for Better Care. Washington, DC: The National Academies Press, Washington DC 2012. Available athttp://www.nap.edu/openbook.php?record\_id=13269&page=1Press, 2012, Washington DC. [↑](#footnote-ref-5)
6. Health Level Seven, EHR Interoperability Work Group, Coming to Terms: Scoping Interoperability for Health Care, February 7, 2007. Available at <http://www.hln.com/assets/pdf/Coming-to-Terms-February-2007.pdf> [↑](#footnote-ref-6)
7. GridWise Architecture Council and Harbor Research, Inc.*, Financial Benefits of Interoperability: How Interoperability in the Electric Power Industry Will Benefit Stakeholders Financially*, September 2009 [viewed 17 October 2012]. Available at www.gridwiseac.org/pdfs/financial\_interoperability.pdf [↑](#footnote-ref-7)
8. J Walker et al., ‘The Value of Health Care Information Exchange and Interoperability’, HealthAffairs, 19 January 2005. Available athttp://www.ncbi.nlm.nih.gov/pubmed/15659453 [↑](#footnote-ref-8)
9. Health Level Seven, *Service-Aware Interoperability Framework*—Canonical Definition, Normative Draft Standard for Trial Use Ballot 1, January 2012. Available at <http://www.hl7.org/documentcenter/public/standards/dstu/SAIF_CANON_DSTU_R2_2012MAY.pdf> [↑](#footnote-ref-9)
10. HL7/OMG Healthcare Services Specification Project [ viewed 17 October 2012]. Available at hssp.wikispaces.com/ [↑](#footnote-ref-10)
11. *Australian Government Architecture Reference Models*, Version 2.0, December 2009. Available at <http://agimo.gov.au/policy-guides-procurement/australian-government-architecture-aga/aga-rm/> [↑](#footnote-ref-11)
12. http://loinc.org/ [↑](#footnote-ref-12)
13. SNOMED CT® is a registered trademark of the International Health Terminology Standards Development Organisation. [↑](#footnote-ref-13)
14. The National Association of Testing Authorities, commonly known as NATA, is the authority responsible for the accreditation of test laboratories, inspection bodies, calibration services, producers of certified reference materials and proficiency testing scheme providers. [↑](#footnote-ref-14)
15. The Joint Accreditation System of Australia and New Zealand, commonly known as JAS-ANZ, is the government-appointed accreditation body responsible for providing accreditation of conformity assessment bodies in the fields of certification and inspection. [↑](#footnote-ref-15)
16. <http://cabig.cancer.gov/perspectives/biomedicine/health20/soa/> [↑](#footnote-ref-16)