Unique_Ballot_id_R1_INFORM_2013MAY

Healthcare SOA Ontology, Release 1

May 2013

HL7 For Comment Only Ballot

Sponsored by:
SOA WG
Healthcare Services Specification Program

Active Contributors

Contributor	Organization
Zoran Milosevic	Deontik Pty Ltd, Australia
Sasha Bojicic	Infoway, Canada
Michael Rossman	Kaiser Permanente, USA
Max Walker	DH Victoria, Australia

Abstract

This paper provides an initial set of ideas for describing services in eHealth taking into account concerns of different stakeholders involved in the definition, implementation and management of services. The emphasis of the paper is on identifying key concepts needed for describing eHealth services from business (i.e. healthcare) and technical perspectives (i.e. an ehHealth service semantics) - consistent with HL7 SAIF-CD. These concepts serve as a means for establishing a common language for service design and implementation for both groups of stakeholders. Such a language is a foundation for incremental development of multiple service taxonomies and ontologies in eHealth - reflecting needs of different applications and stakeholders. The primary aim of this ballot document is to get early feedback from HL7 community about the proposed eHealth service semantics, the agreement for which is needed, before the development of eHealth service taxonomies and other knowledge management applications regarding eHealth services.

1. Introduction 4
2. Background 5
Taxonomy related influences 5
HL7 EHR-S Functional Model 6
eHealth Interoeprability Framework 6
Service related conceptual models 6
HL7 Service Aware Interoperability Framework 6
Reference Model for Open Distributed Processing 7
OASIS Reference Architecture Framework 8
3. Use cases 8
Consumer oriented 8
Use case 1: eReferrals 9
Provider oriented 15
Use case 2: Add new service – use of organisation's service catalogue 15
Use case 3: Adding new service capability – community environment 16
4. Service semantics 19
Foundational concepts 19
Service 19
Object 20
Service description 21
Service contract 21
Policy 21
Enterprise viewpoint 22
Business service 22
Business Service description 23
Business Service contract 23
Computational viewpoint 24
Computational service 24
Computational service description 25
Computational service contract 25
Computational policy 25
Business and computational service – distinction and correspondence 26
5. Example: eReferral 28
Enterprise viewpoint 28
eReferral business services 28
Computational viewpoint 33
Computational service and its interface 34
Requestor role 34
Oferrer role 34
6. Examples of classifications 36
7. Next steps 38
References 39
Appendix 39

1. Introduction

The OMG Ontology Definition Metamodel [OMG ODM] provides an excellent reference framework for ontologies and we use it where possible to frame this SOA healthcare ontology, beginning with the ontology definition next.

> An ontology defines the common terms and concepts (meaning) used to describe and represent an area of knowledge. An ontology can range in expressivity from: a Taxonomy (knowledge with minimal hierarchy or a parent/child structure), to a Thesaurus (words and synonyms), to a Conceptual Model (with more complex knowledge), to a Logical Theory (with very rich, complex, consistent, and meaningful knowledge)

There are many kinds of application for ontologies as also elaborated in section 7 of the OMG Ontology Definition Metamodel. Ontologies can be used at design time only or at both design and run time; they can involve types (schemas) only or involve both types and instances; their structure can be imposed from outside their domain or can emerge from the activities of interoperating parties.

The purpose of the SOA Healthcare Ontology described in this document is to express knowledge about ehealth services while following the tenets of the SOA approaches [SoaML, SOA RAF]. Several uses cases have been considered (as summarised in section 3) which highlight the need to support the following capabilities:

1. Semantically grounded run-time discovery of eHealth services; this is to support healthcare providers and others involved in healthcare delivery to describe and find suitable ehealth services in the context of care coordination and handover
2. Design time analysis and construction of eHealth services; this is to support service architects, developers and service portfolio managers to look for the existing services in a service catalogue and develop new services while reusing the elements of the existing ones, as required
3. Design and run-time update of ontology definitions in ontology repositories – both standards and localised ontologies; this is to be performed by ontology developers and authoritative bodies
4. Extension of the core ontology specified in this document by other standards bodies or organisations through definition of new concepts for eHealth services. This would include the development of new classifications or taxonomies;

The update capability identified at (3) above corresponds to the *ontology lifecycle* usage scenario identified in the ODM, with the principal objectives of conceptual knowledge analysis, capture, representation, and maintenance.

The first three ontology capabilities above are aligned with two business application scenarios identified in [ODM], namely *run-time interoperation* and *application*

generation. The following requirements from the ODM business application scenarios are relevant for this healthcare ontology:

- the ability to represent situational concepts, such as player/actor – role – action – object – state,
- the necessity for multiple representations and/or views of the same concepts and relations, and
- separation of concerns, such as separating the vocabularies and semantics relevant to particular interfaces, protocols, processes, and services from the semantics of the domain.

Considering the requirements above and the use cases specified in section 3 of this document, the scope of this ontology is limited to the taxonomy and conceptual model aspects of the ODM ontology definition.

In addition to the scope and requirements captured above, the following development principles are applied:

- Use existing HL7 concepts related to the specification of eHealth messages, documents and services
- Use other existing reference architecture concepts, in particular the HL7 Service Aware Interoperability Framework, RM-ODP [ODP], Oasis SOA Reference Architecture Framework {SOA RAF], SoaML [SoaML], ISO HISA [HISA] and ContSys [Contsys].

Section 2 of this paper lists several standards initiatives of relevant to the taxonomy and conceptual aspects of ontology.

The use cases considered are described in section 3.

The service conceptual model, described using a meta-model according to the OMG MOF principles is explained in section 4.

Section 5 provides an example of service classifications, taking into account service concepts identified in section 4.

Section 6 outlines next steps for this work.

2. Background

This specification was influenced by the following standards or initiatives:

Taxonomy related influences

The following is a list of some key initiatives that have influenced the taxonomy applications of the SOA eHealth services ontology.

HL7 EHR-S Functional Model

The HL7 EHR System Functional Model provides a reference list of functions that may be present in an Electronic Health Record System (EHR-S). The function list is described from a user perspective, with the intent to enable consistent expression of system functionality. This EHR-S Model, through the creation of Functional Profiles, enables a standardized description and common understanding of functions sought or available in a given setting (e.g. intensive care, cardiology, office practice in one country or primary care in another country)

eHealth Interoeprability Framework

eHealth Interoperability Framework is an Australian adoption and adaption of HL7 SAIF. The framework can be used for building e-health specifications and also provides a classification scheme for the design and implementation artefacts that may be incorporated in more sophisticated ontologies. Such a classification can include:

- high level business services, such as e-discharge, e-referrals, care plans, electronic transfer of prescriptions; as well as
- technical (computational) services, such as retrieving information from repositories, storing information, access control, and so on.

The classification rules and approach should reflect the concepts in the E-health Interoperability Framework and is anticipated to be developed as part of collaborative efforts of relevant stakeholders in the Australian e-health community, including healthcare organizations, vendors and standards organizations.

Service related conceptual models

The following are key influences that were considered when developing service meta-model as a way of expressing a conceptual model for the service ontology.

HL7 Service Aware Interoperability Framework

Service Aware interoperability framework - canonical definition (SAIF-CD) defines a set of canonical frameworks consisting of languages that could then be instantiated in organization-specific Implementation Guides (IGs).

Each modeling language defines a set of concept and relationships that can be further refined into more specific models reflecting specific areas of concern.

SAIF-CD defines the following languages:

The language of the Governance Framework (GF), which enables an enterprise implementing SAIF to define explicit, organization-specific policies, standards, and roles

The language of the Behavioral Framework (BF), which defines constructs to specify the dynamic semantics of interactions in a shared purpose interoperability scenario. The BF

focuses on the semantics of contracts, operations, and processes that collectively define shared purpose scenarios at a technical level.

The language of Information Framework (IF), which defines the static/informational semantics relevant to interoperability scenarios, including information and terminology models, metadata, vocabulary bindings, value sets, executable models, etc. that collectively specify the static semantics of interactions. This includes the language to describe patterns of structured and unstructured data, documents, messages and services, quality measures and transformations.

The language of Enterprise Consistency and Conformity Framework (ECCF), which describes the various relationships – e.g. conformance, compliance, consistency, traceability, compatibility, etc. – between the artifacts that collectively define a given specification, including the relationship between a specification and the derived implementations of the specification, and other specifications that use one or more of the artifacts as part of their artifact collection.

Reference Model for Open Distributed Processing

The Reference Model for Open Distributed Processing (ODP) is an ISO/ITU-T standard created to give a solid basis for describing and building widely distributed systems and applications in a systematic way. It stresses the need to build such systems with evolution in mind. It identifies the concerns of major stakeholders and then expresses the design as a series of linked viewpoints. Each viewpoint defines a set of concepts and their relationships, as a conceptual ontology. These viewpoints are:

The enterprise viewpoint, concerned with defining the purpose and scope of the systems, key stakeholders involved in community, and the interactions and policies that apply to them.

The information viewpoint describes the shared information that is manipulated by the system, in order to provide a common understanding to all parties. In this viewpoint the focus is on the information itself, without considering further platform-specific or implementation details, such as how the data is represented, implemented or distributed.

The computational viewpoint describes a set of concepts and relationships that allows system architects to express their designs using a set of basic elements, which are common to most software architectures and languages. It describes the functional decomposition of an ODP system as a configuration of computational objects, the interactions that occur between those objects at their interfaces, and the environment contracts for them, specifying non-functional constraints.

The engineering viewpoint describes a set of concepts and their relationships needed to identify and specify the supporting mechanisms for distributed interactions between objects. The focus is on specifying how distribution works — how objects are distributed over nodes, and how the structures of the nodes, and of the channels linking the nodes, are going to be modelled. It also defines common functions needed to support the

required distribution transparencies. Clearly, this viewpoint is used by system designers who are particularly concerned with the infrastructure of systems.

The technology viewpoint provides concepts for specifying the hardware and software products from which the system is built, to test that such an implementation complies with the specification as prescribed by the rest of the viewpoints and to specify the plans and processes for the selection, acquisition and evolution of the system parts (hardware and software products) during its lifetime.

This ontology primarily focuses on the enterprise and computational viewpoints because they are concerned with behavioural aspects of systems and thus the interactions and organizational aspects of relevance for using and developing services.

OASIS Reference Architecture Framework

The OASIS Reference Architecture Foundation for Service Oriented Architecture (SOA-RAF) extends the concepts and relationships defined in the OASIS Reference Model for Service Oriented Architecture. The focus of the SOA-RAF is to integrating business with the information technology needed to support it. These issues are always present but are more important when business integration involves crossing ownership boundaries.

The SOA-RAF follows the recommended practice of describing architecture in terms of models, views, and viewpoints, as prescribed in the ANSI/IEEE 1471-2000 (now ISO/IEC 42010-2007) Standard. It has three main views:

- the Participation in a SOA Ecosystem view which focuses on the way that participants are part of a Service Oriented Architecture ecosystem;
- the Realization of a SOA Ecosystem view addresses the requirements for constructing a SOA-based system in a SOA ecosystem;
- the Ownership in a SOA Ecosystem view focuses on what is meant to own a SOA-based system.

The SOA Reference Architecture Framework addresses holistically the concept of service and, together with the RM-ODP approach and SAIF, serves as a basis for addressing many of the requirements identified above.

3. Use cases

Two broad categories of use cases were considered as a way of identifying requirements for the SOA healthcare ontology, i.e. consumer-oriented and provider oriented use cases as described next.

Consumer oriented

A consumer use case related to the processes associated with electronic support for referrals was considered as a simple use case example, and is described next.

Use case 1: eReferrals

The eReferral use case describes typical roles, interactions and documents exchanged when a referring provider needs to refer a patient to another provider to handover care. These form the characterising elements of an Referral service when considered holistically, i.e. from the perspective of both the healthcare professionals as consumers of eReferral service, i.e. referring (referrer) and referred-to clinicians, and software vendors as providers of this service. This use case is based on the material from Canada Infoway [Infoway] and Nehta eHealth blueprint [NEHTA].

In order to implement an eReferral service the following functionality is identified:

- Support for healthcare provider participation registration
- Lookup or search for other providers e.g. specialists, with the aim to initiate a transfer of care
- Transmission of an eReferral from the referring provider to the referred-to provider
- Completion of the process by the referred-to provider accepting or rejecting the referral and completing the processing of eReferral request(s).

This functionality is described using several use cases below.

The following picture depicts high level processes of the eReferral workflow, followed by each of the sub-processes.

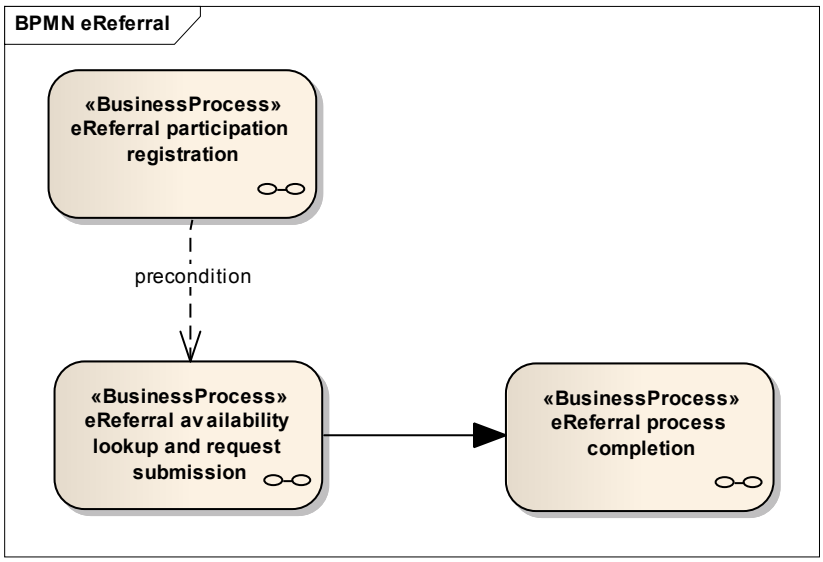

Figure 1 eReferral service components

Registration for eReferral service delivery (Admin Process)

This workflow describes eReferral registration process conducted by a provider who intends to participate in an eReferral community (e.g to become "eReferral enabled"). This applies to any provider but the use case below uses referring provider to illustrate the interactions.

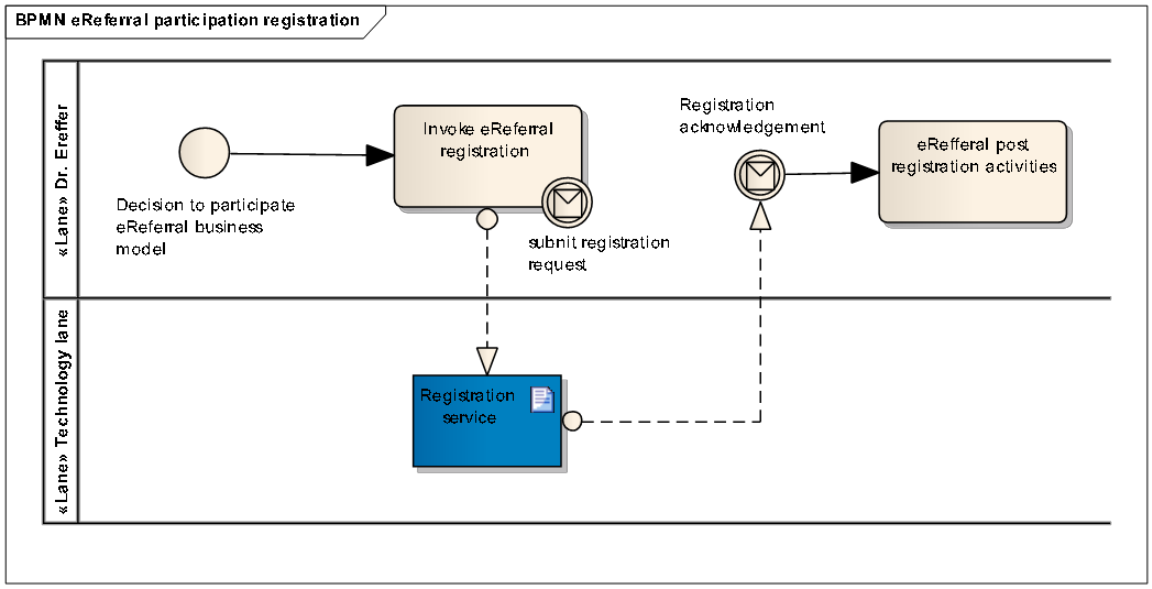

Figure 2 eReferral participation registration

1. Decision to participate eReferal model

Dr Toreff decides to register for and participate in eReferral health care service delivery model. This will allow **Dr Toreff** to extend his practice and accept eReferral(s) through an alternate health care delivery channel

2. Invoke eReferral registration task

Dr Toreff

- *Authenticate* as user on the local system (e.g. EMR)
- *Select* link to eReferral registration service exposed by EHR via local system
- *Provide* information required for registration
- *Submit* eReferral registration request

3. Registration submission

A precondition to this step is that the provider identity has been already established, by a suitably qualified credentialing authority , such as a medical registration board. Further, the provider identity should be already established, e.g. through a set of set of Interdependent Registries which the provider has used to register its Health Service, Service Location and Organization.

The system does following:

•The service *updates* provider's profile and flag the Provider as "*eReferral enabled*"

•The service *subscribes* Dr. Toreff for eReferral delivery queue (facility)

•The system *sends* acknowledgment to Dr. Toreff, indicating successful registration for eReferral delivery model.

(In some cases the system verifies the provider's credential by validating that credential with the credentialing authority).

4. Post registration activities

•**Dr Toreff** *configures* required system features for eReferral delivery (accepts subscription to eReferral request)

eReferral look up and request submission

This workflow describes process where a provider decides to refer a patient to another provider who is *eReferral enabled*

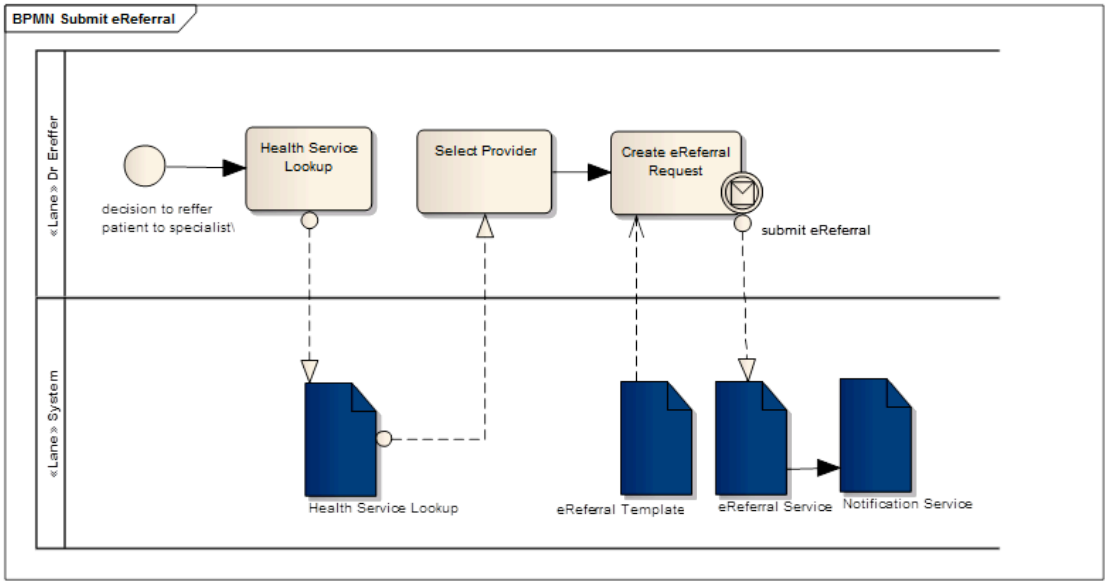

Figure 3 eReferral search/lookup

1. Patient symptoms indicate need for specialist referral

Dr Ereffer sees patient John Doe. John complains about shortness of breath, pain in the left arm and occasional sharp chest pain. Based on symptoms and vital parameters reading (blood pressure beyond threshold), Dr Ereffer decides to refer John to Internist for further exams.

2. eReferral Lookup

Dr Ereffer:

•*Select* Health Service lookup (or search) function

•*Provide* search criteria: eReferral service type (internist), location / area for service delivery and approximate date

3. Select provider

Dr Ereffer:

•*Review* list of available providers selected based on lookup criteria

•*Decide* that Dr Toreff is the most appropriate selection for the patient (proximity to patient's home, availability)

•*Select* Dr Toreff for eReferral request

4. Create eReferral Request

Dr Ereffer:

•*Pull* the standardized eReferral template from the local or central repository

System

•Automatically *preset* patient's and provider's demographic data.

•*Provide* required data / content for eReferral request (we need to figure what those data / attributes / content are

•*Submit* the request

System

•*Format* eReferral request data in form of CDA document

•*Place* referral in the queue for Dr Ereffer

•*Trigger* Notification service to inform **Dr Toreff** and **Ereffer** that new eRefferal was submitted.

eReferral completion

This workflow describes process where Provider process eReferral request.

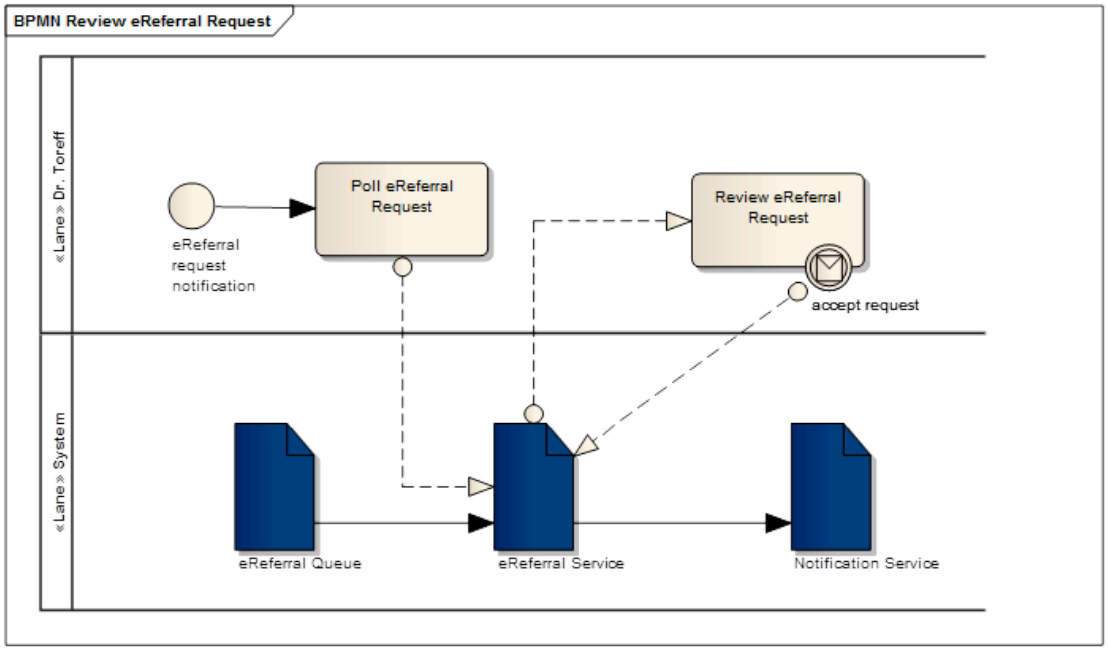

Figure 4 eReferral review and completion

1. Request notification received

Dr Toreff *receives* new eReferral request notification.

2. Pull referral request
Dr Toreff *polls* referral queue.

3. Review request
Dr Toreff (One or more request may be in the queue.)

- *Select* and *review* each request from the eReferral queue.
- *Decide* that patient John Doe has symptoms that qualify him for further investigation.
- *Decide* to *accept* eReferral request.
- *Submit* acceptance acknowledgement.

4. Accept/Reject eReferral

System
•*Update* status of eReferral record (either accepted or rejected).
•*Trigger* Notification service to inform **Dr Toreff** about eReferral request acceptance.

Provider oriented

Two use cases were considered from this perspective, reflecting a specific organisation's concerns and reflecting community concerns, as described next.

Use case 2: Add new service – use of organisation's service catalogue

This use case presents the process of new service development in a large organisation. It shows how the existence of a structured/formal approach to service description, as adopted through a service ontology approach, can facilitate the analysis of existing services and promote reuse of service components.

The use case is centred around the organisation's service catalogue, that holds information about eHealth services in an organisation. The role of ontology is to maintain description of service types by means of service descriptors, which are essentially meta-attributes established for the purpose of searching for a specific service type value.

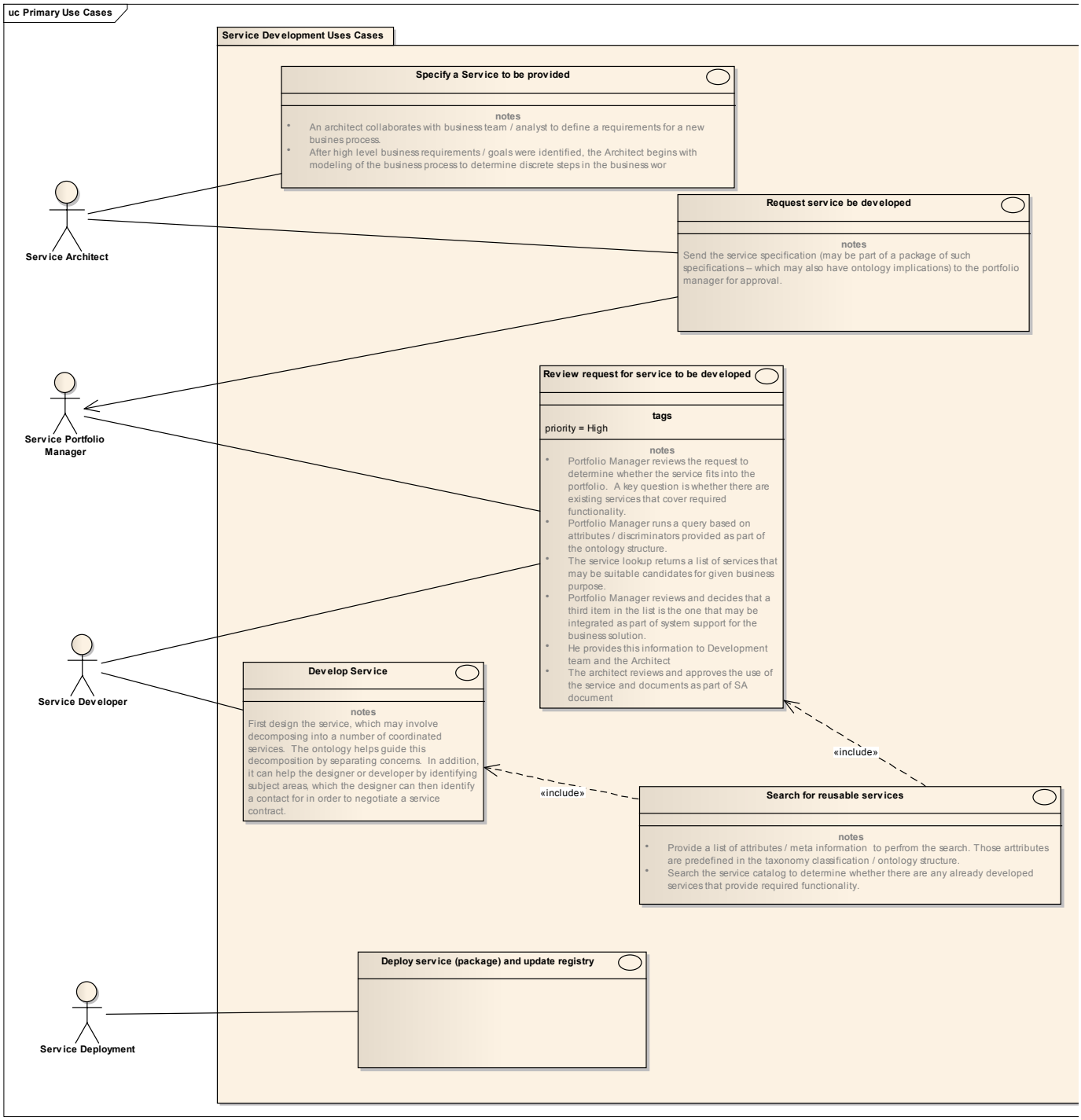

Figure 5: Add new service - large organisation

Use case 3: Adding new service capability – community environment

This use case illustrates the process of the development of an eHealth service and its deployment in a cross-organisational setting such as the one in Australian environment.

The use case makes use of an End Point Locator service (ELS) developed by NEHTA and being standardised in Australia. ELS is used by an organisation to identify end points through which it can participate in service interactions – essentially specify its capability to handle technical services.

The following is a description of this use case.

ABC Home Care is a leading organisation delivering services to the aged and disabled in their homes. In order to improve service delivery they have developed a dependency assessment that forms the basis of their care planning. ABC participate in a number of shared care arrangements and would like to ensure that their dependency assessment can be shared with the client's GP and other organisations that deliver care into the client's home.

ABC approach their software vendor SoftwareXL to develop the Dependency Assessment. Software XL suggest that ABC create a collaboration of service providers to jointly design the the assessment.

ABC perform a non secure lookup on the services directory to find a list of organisations that deliver aged and disability care in the home with the purpose of setting up a collaboration. ABC also contact their government funding body and industry association to see if they would participate in the development.

The assessment is designed by the collaboration and specified by SoftwareXL.

The specification submitted to the national standards authority for review. Following standards ratification, the assessment specification is published for other vendors to incorporate in their software.

The specification includes a template, data definitions for the fields, and the business rules for processing the data items. It also includes the information required to populate the ELS that describe the dependency assessment.

Software XL develops the dependency assessment and incorporates it into Release 2 of the software

When Release 2 is installed at ABC Home Care's site, the software issues an electronic notification to the ELS to register the dependency assessment as a capability of ABC Home Care and adds a copy of ABC Home Care's public certificate to the ELS entry.

Another organisation, Home Helpers has also implemented the dependency assessment. They wish to refer to ABC Home Care and attach the client's dependency assessment to the referral.

The Home Helpers co-ordinator creates the referral and attaches the dependency assessment. The Home Helpers application locates the ELS entry for ABC Home Care,

and retrieves the capabilities, public certificate and electronic address for ABC Home Care. The Home Helpers application confirms that ABC Home Care is capable of processing the Dependency Assessment, and thus encrypts the assessment with ABC's public key, and sends the referral with the assessment attached.

This scenario is depicted in the process model below.

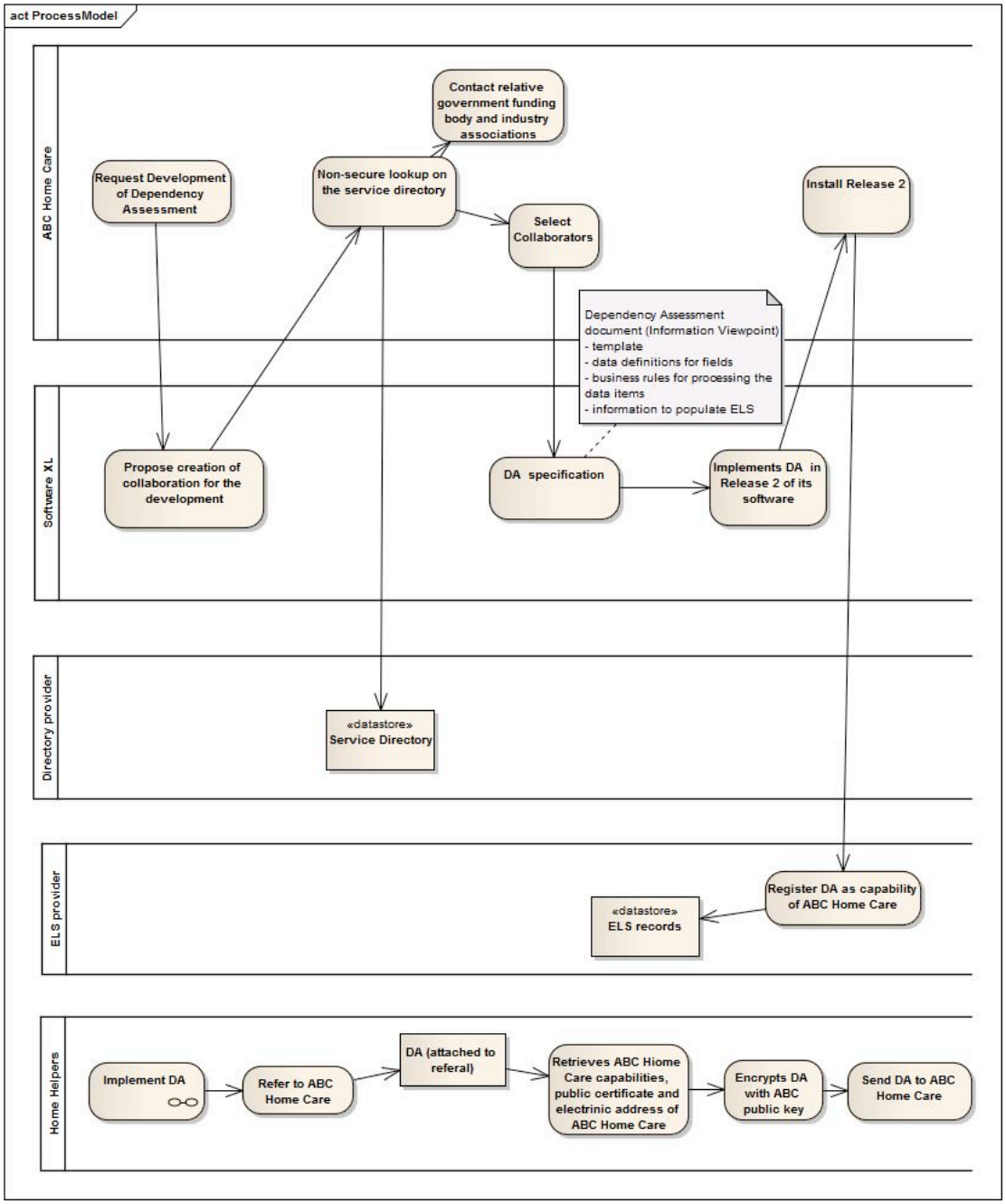

Figure 6 Add new service - community

4. Service semantics

This section identifies key concepts needed to precisely define a service, with particular emphasis on identifying different service properties that reflect concerns of respective stakeholders, e.g. business users, information modellers, solution architects and developers.

Each modelling concept is described in its own section and should be considered in the context of the service meta-models (see included figures).
In the rest of the document, several concepts from RM-ODP were used to describe service, i.e. entity, object, action, interaction, viewpoint and policy and these are definitions are provided in Appendix A.

The description begins with core service modelling concepts as defined in RM-ODP standards [RM-ODP], followed by their refinements from the enterprise and computational viewpoints, as introduced in HL7 SAIF [SAIF DSTU]. The purpose of these core concepts is to define the foundational concepts for service, while the refinements in the enterprise and computational viewpoint are used to reflect both the business context and the service logical implementations respectively. The enterprise and computational concepts are linked because they refer to the same system, albeit focusing on different characteristics of the system.

Note that there are implications for RM-ODP information, engineering and technology specifications [ODP], as introduced in the RM-ODP sub-section in section 2 above, but these are not elaborated in this document.

Foundational concepts

Foundational concepts are abstract concepts serving as a common base for the enterprise and computational views of service, which in turn may be used for modelling and building systems.

There are a number of definitions of service [SOA RAF], [SoaML] but many of them could be conceptualised through the following definition from RM-ODP [ODP]:

Service

> A behaviour triggered by an interaction, that adds value for the service users by creating, modifying or consuming information; the changes become visible in the service provider's environment.
>
> Note that the provider's environment includes the service user and the last sentence suggest that provision of a service involves some kind of commitment by the provider to stand by its actions; they are seen by at least some things outside it, so the action is subject to audit.
>
> The provision of a service involves a collaboration between its provider and its user. This collaboration may involve a complex series of interactions [RM-ODP].

Thus, a frequently used term 'service offer' is necessary but not sufficient condition for a service to be instantiates. What is also required a participation of a service user in the collaboration.

A service can be composed out of other services and it can also have relationship with other services. A special kind of relationship is 'is a' relationship typically used to construct relationship between elements in taxonomy, e.g. an orthodontist service is a dental service.

Note that this definition refers to the concept of service as an instance of a real-world thing that involves interactions between service users and service providers. It implies that there is an implicit or explicit agreement between them (i.e. a service contract). There may be also a service description as a way of supporting the expression of what service provider offers to the environment to facilitate the understandings of service users' understanding of what value service will deliver to them. In RM-ODP, service description can be expressed using the concept of template, defined 'the specification of the common features of a collection of <X>s in sufficient detail that an <X> can be instantiated using it', e.g concrete classes in object oriented languages such as Java.

Service user and service provider

Service user and *service provider* are service roles, i.e. a service user triggers the behaviour made available by a service provider (often referred to as a 'service offer').

More precisely, it is an actor object fulfilling the service user role that triggers service behaviour and it is an actor object that fulfils the service provider role that offers service behaviour (referred to a capability in some SOA approaches). both these actor objects thus need to posses capability to act in order to participate in service behaviour, the concept of which is used in several SOA approaches. For example, in soaML a capability is defined as 'the ability to act and produce an outcome that achieves a result. It can specify a general capability of a participant as well as the specific ability to provide a service' [SoaML].

Object

An *object* is an entity that has its own identity, independent of its participation in service roles.

Note that this distinction between service roles and objects fulfilling roles is in line with the SoaML definition of service, where the concept of participant has a similar modelling purpose to object.

A service is specified through a service description, defined below.

Service description

An artifact, that defines or references the information needed to use, deploy, manage and otherwise control a service.

This information includes not only the information and behavior models associated with a service that define interaction via the service interface, but also includes information needed to decide whether the service is appropriate for the current requirements of the service consumer. Thus, the service description should also include information such as service reachability, service functionality, and the policies associated with a service [SOA-RAF]. Note that these various components of service description are expressed in different viewpoints of service.

A service description is typically created by a service provider and published in a directory. A service user can inspect a service description and either accept it 'as is' or negotiate with the service provider to refine or customize the service description for that service users needs. When both the service provider and service user accept the refined service description, this constitutes a basis for a service contract, described next.

Service contract

An agreement that defines the behaviour of service user and service provider roles involved in the provision and consumption of a service, as well as information models and policies that constrain the behaviour of service users and providers.

The distinction between service, service description and service contract is influenced by the SOA RAF specification and is in line with the RM-ODP concepts of service, (service) type and (service) contract.

Policy

In general, a policy is defined as '*a constraint on a system specification foreseen at design time, but whose detail is determined subsequent to the original design, and capable of being modified from time to time in order to manage the system in changing circumstances*' [ODP].

Policies can apply in any viewpoint. For example enterprise policies define business, legal and social constraints at the time, while IT policies can be stated in the information, computational, engineering and technology specifications, as per RM-ODP and HL7 SAIF.

Note that some policies are defined by the service contract while others are defined in the environment external to the service.

These foundational concepts related to services are depicted in the meta-model shown in the figure below.

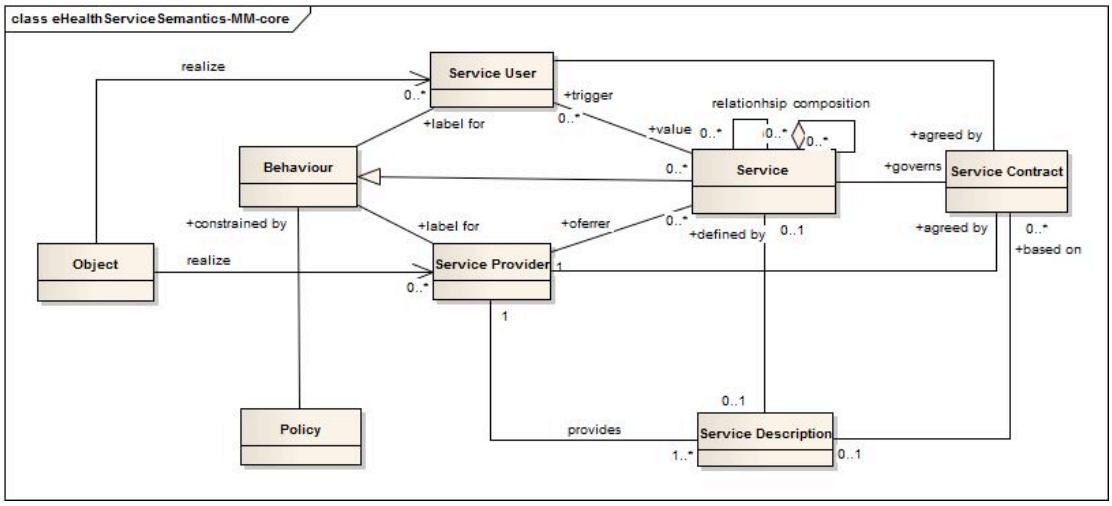

Figure 7 Service concepts – foundations (core)

These concepts are further refined in the enterprise and computational viewpoints [HL7 SAIF], [RM-ODP], as discussed below.

Enterprise viewpoint

In the enterprise viewpoint, the foundation concepts are further refined to reflect concerns from the stakeholders concerned with organisational and policy aspects. For example, the core concept of service is refined into business service, the concept of object is refined into enterprise object, and policy into enterprise policy, each of which will be described below.

Business service

A *business service* is an elaboration of the concept of service described above, including the following additional concepts and refinements:

- Service roles are referred to as *consumer* and *provider*, to reflect typical usage in a business environment. Service roles may be expressed as parties, while others are *active enterprise objects*.
- *Party* is an enterprise object modelling a natural person or any other entity considered to have some of the rights, powers and duties of a natural person [RM-ODP]
- *Active enterprise object* is an enterprise object that can be involved in some behaviour, for example a clinical information system [ODP EL].
- The description and specification of *business service behaviour* captures collaboration between consumer and provider which can typically be implemented through a *business process*. A business process is defined as 'A collection of steps taking place in a prescribed manner and leading to an objective' [RM-ODP]. Note that for the purpose of eHealth service modelling,

the specific details of particular process language are omitted, but could be added if use cases requires so.

- Identification of *information artifacts* representing, for example, documents exchanged or consumable resources that are essential in delivering services; note that some of these may not require IT support but nonetheless would need to be identified in an enterprise viewpoint of a service model
- A business service needs to be compliant with *enterprise policies* that apply to the business service. Typically, deontic policies such as obligations, permissions, prohibitions and authorisations would state constraints on the use or management of business services [EL]. These can in turn provide a foundation for describing more complex policies such as responsibility, accountability, consent, privacy and duty of care, each of which can be expressed as a combination of obligations and permissions. Note that privacy and consent policies that might apply are further discussed in the Privacy, Access and Security Services project [PASS].
- In order to be able to measure value that a business service delivers to service users, the enterprise concept of *evaluation* is introduced for this purpose, defined as 'an action that asses the value of something' [ODP EL]
- A business service can be invoked from a business process but can also be realised through business process, such as in case of eReferral service presented in next section.

Business Service description

Business service description extends a service description using the business service concepts as above. The description can also specify metrics or other data needed for service monitoring and assessment, including significant business events (e.g. payment) and non-functional service properties (e.g. service level agreements).

The business service description can be advertised by service providers through service offers in a publicly accessible directory. Consumers can inspect these descriptions and accept the conditions or negotiate further tailoring for their requirements. The availability of a service taxonomy is expected to facilitate lookup and search of business service descriptions.

Business Service contract

Once an agreement is reached on the business service description between a service provider and consumer, a business service contract formalises the agreement. In many industries the term Service Level Agreement is used to describe the operational and service quality constraints associated with a service contract.

Note that a business service contract will need to include specific values related to the non-functional properties such as Quality of Service (QoS) requirements, price and so on.

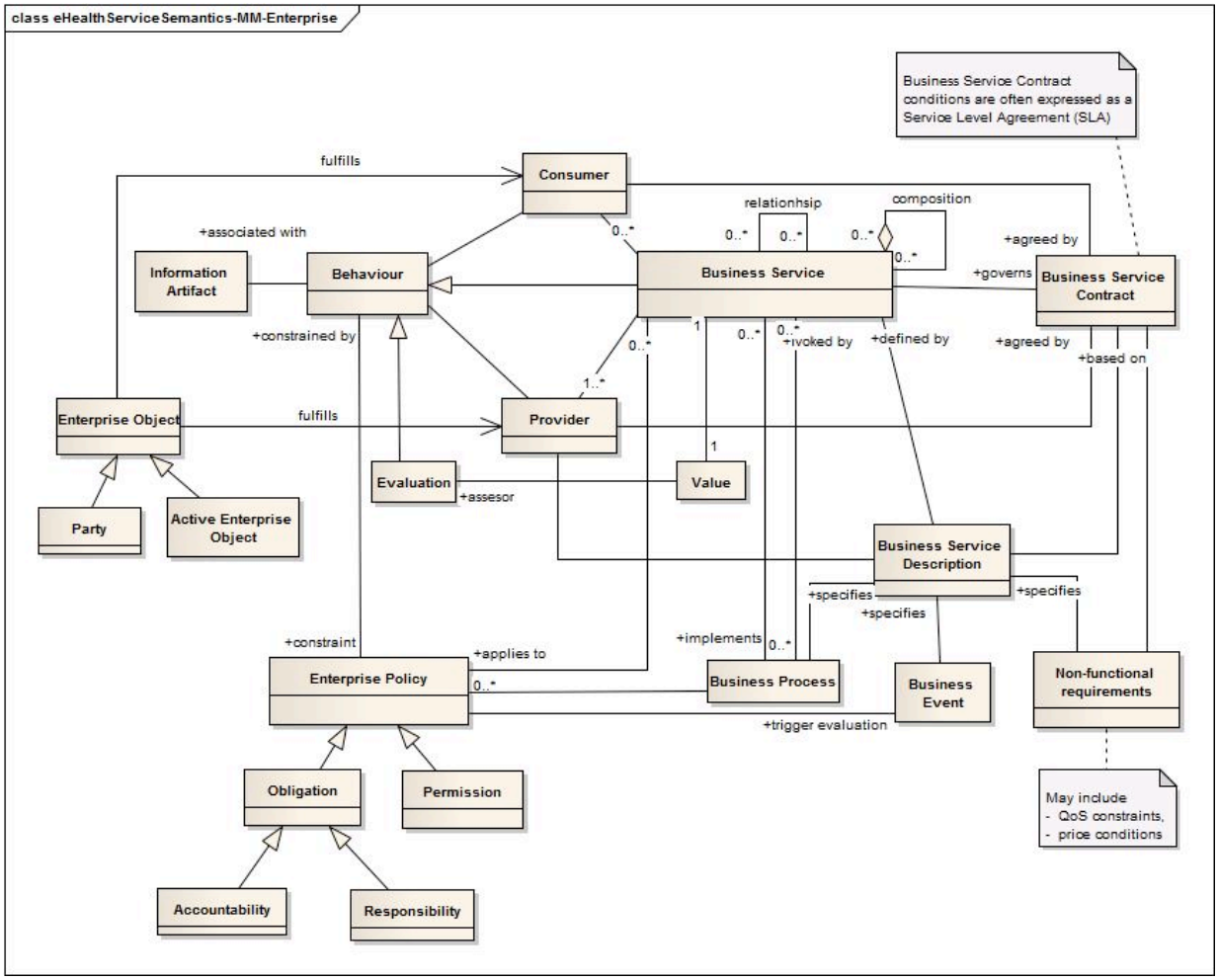

Figure 8 Service concepts - enterprise viewpoint

Computational viewpoint

Computational service

A computational service is a refinement of the core concepts of service, adding specific concepts for computational interactions, as listed below:

- Service roles are referred to as *requestor* and *offerer*; these terms are deliberately chosen to be distinct from the service role in the enterprise viewpoint.
- A *computational interface* provides access to a computational service both for the computational object fulfilling offerer role and for the computational object fulfilling requester role (as in the UML's provider and required interfaces);
- An *environment contract*, which states non-functional properties of a service offered, such as availability, delay and so on.

Note that the distinction between computational and business services is discussed at the end of this section in the context of viewpoint correspondences.

Computational service description

This modelling element captures properties of computational service that are offered by computational objects, in effect defining computational service types. These offers can be looked up through a computational service directory or searched by the requestor service role and once the computational service description is agreed (as advertised or through further tailoring), it forms the basis of a service contract.

Computational service contract

A *computational service contract* models an agreement for service provision. It varies depending on service type and other service variables such as those defined in an environment contract stating constraints on the service offered.

In simplest cases, this can be a description of a service interface, in terms of its signature, or it can be a computational binding [RM-ODP] specified between the Requestor and Offerer service roles. Note that this definition is in alignment with the SoaML service contract type semantics.

Computational policy

A *computational policy* is a computable expression of enterprise policies or requirements,– which form constraints on interactions between computational objects implementing service requestor and offerer roles.

A computational policy can also be derived from the environment contract portion of the computational service contract.

Note that in IT systems the term *business rule* is often used to describe a declarative specification of a computed decision or result. While in many cases, a computational policy could be realised through a business rule, the use of business rules has a broader scope that could include the description of service functional behaviour.

The modelling concepts relating to the computational viewpoint of service are depicted in the figure below.

The figure also provides two computational objects that implement computational policies through the standards approach of using Policy Enforcement Point (PEP) and Policy decision Points (PDPs). A Policy Enforcement Point object observes and intercepts computational interactions that are subject to policies in the service contracts of a computational service. For each interaction, it checks whether any policy is violated by querying the Policy Decision Point. It then allows or blocks the interaction based on the policy decision.

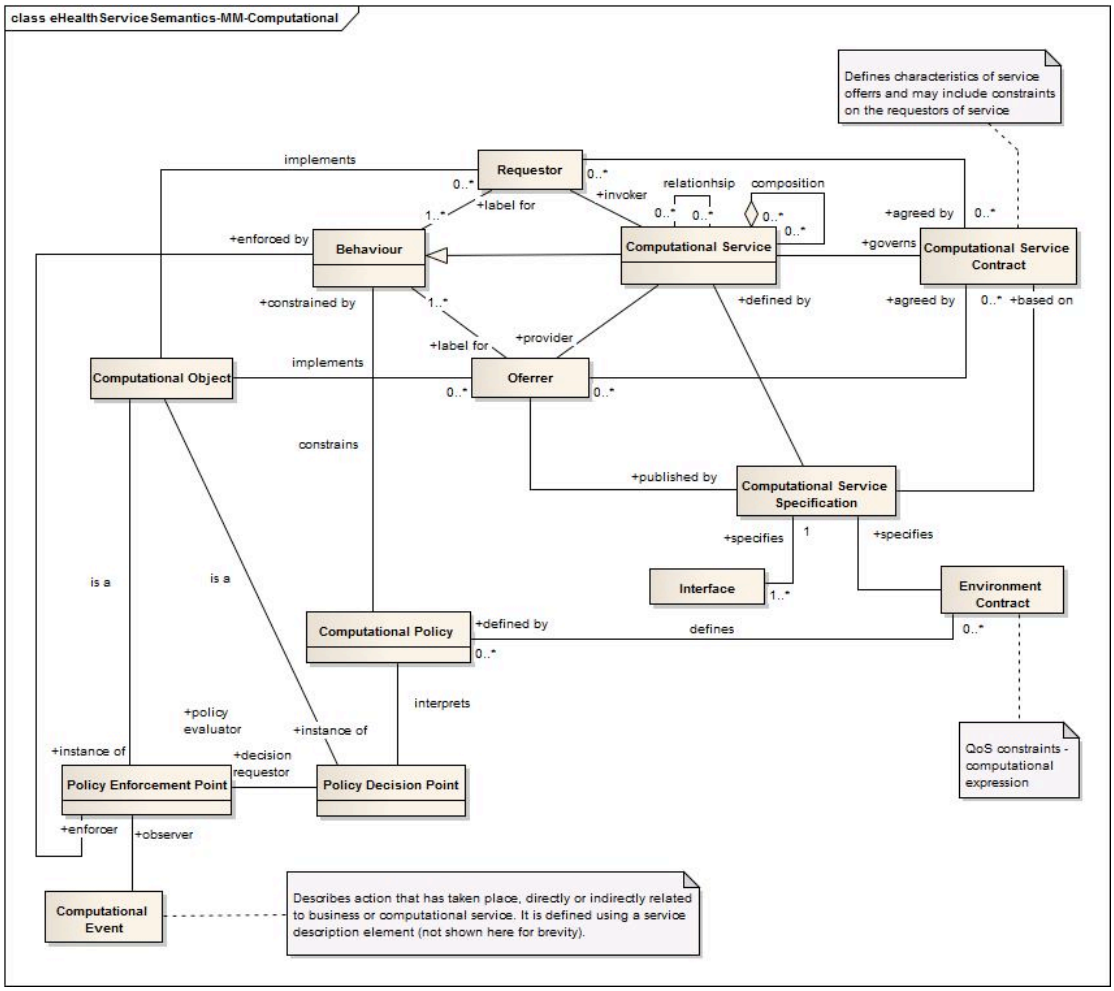

Figure 9 Service concepts - computational viewpoint

Business and computational service – distinction and correspondence

In both the enterprise and computation specification of service contract, the concept of behaviour is used. The enterprise behaviour describes business-level activity and derives from both explicit actions (e.g. a business process) and enterprise policy constraints. Computational behaviour is concerned with the computational interactions while respecting computational policies. In this respect, business service and computational service are specific refinements of behaviour governed by the corresponding service contracts.

Similarly, the concepts of service user and service provider roles are further specialised. In enterprise viewpoint these roles refer to the organisational context while in the computational viewpoint they refer to the communication patterns involved. In other words, in the enterprise viewpoint these are organisational roles involved in business service, while in computational viewpoint they are roles in interaction. Note that RM-

ODP distinguishes between one way interactions (as in message passing), two way interactions (as in traditional RPC style of interaction) and stream type of interactions (e.g. video) but these are beyond scope of this paper.

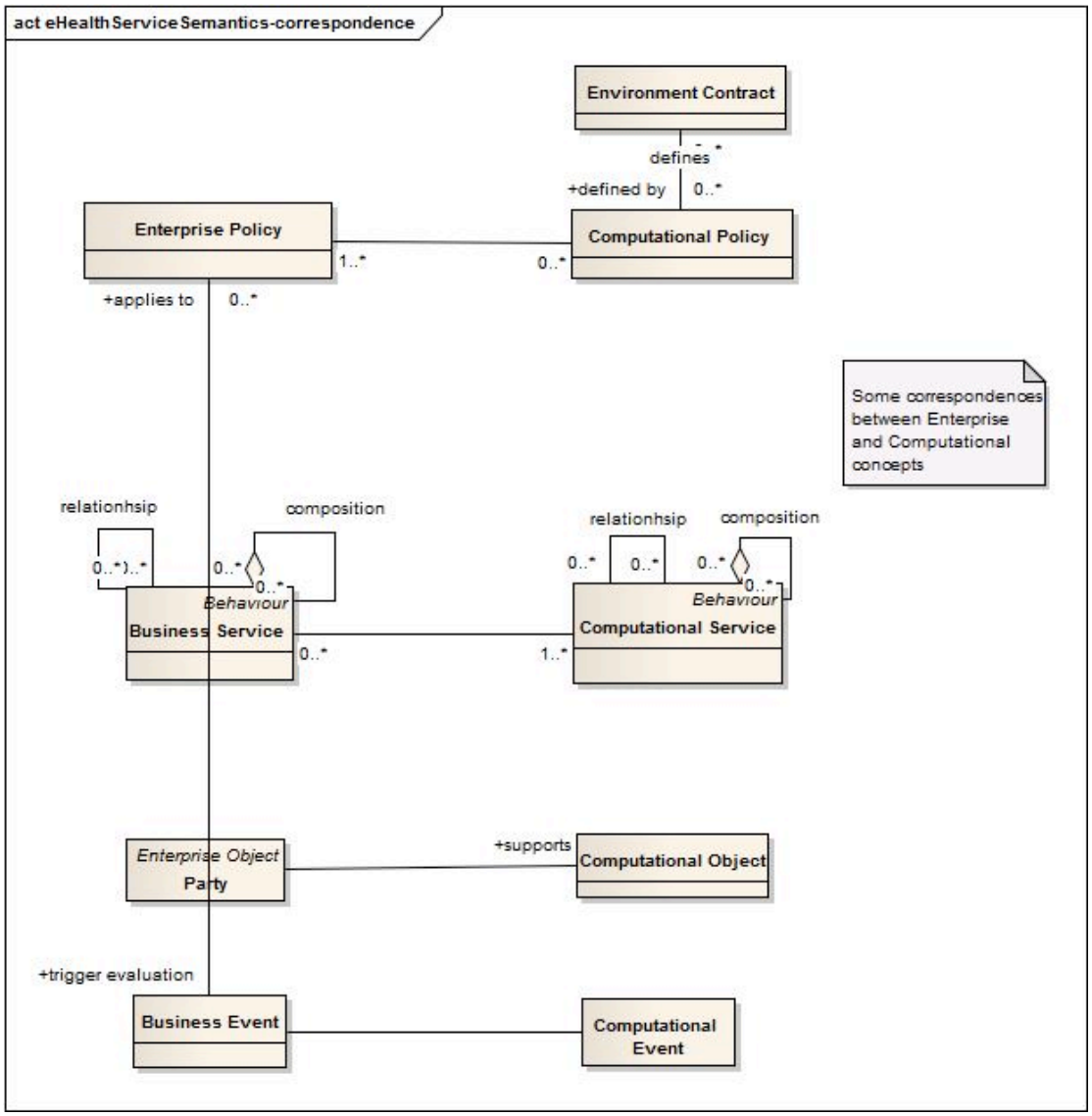

Figure 10: Some correspondences between concepts from different viewpoints

It is important to make the distinction between the two different views on service when specifying organisational or technical aspects of a system.

Consider an example of an immunisation reporting business service, in which a health provider is obliged to regularly deliver an immunisation report to a state immunisation registry. In the enterprise viewpoint, a health provider organisation has the provider role while the immunisation registry has the consumer role, since it obtains business value from the reports through collecting nation-wide data on immunisation. Conversely in the

computational viewpoint, the business service is realised through the immunisation organisation exposing a web service interface (i.e. an offerer) to the health provider organisations that are invoking this service (i.e. requestors).

While the distinction should be made between enterprise and computational viewpoints on service, there is obviously a correspondence between business and computational service as the above example highlights. A computational service can support one or more business services while a business service can be implemented by zero or more computational services (if zero, this means that there is no computational automation of the business service).

Similarly a business event defined in a business service contract can be often linked to one or more computational events, which are often linked to infrastructure related actions, such as arrival of message, evaluation of a security policy and so on.

5. Example: eReferral

The example of an eReferral is used to illustrate the service concepts introduced above. The example should not be treated as a complete eReferral specification; rather it is a simplified example of a fragment of typical activities and processes related to referrals, with the aim of showing how the concepts defined in preceding sections of this document can apply to an eHealth scenario.

Enterprise viewpoint

eReferral business services

Referral is defined as a 'demand for care where a healthcare professional asks a healthcare provider to accept a clinical process mandate', for example, a referral from an orthopaedic surgeon to a rehabilitation service where the surgeon does not plan any further healthcare activities [ContSys].

Informally, an eReferral is a then a process of handing over health care of a subject of care (referred to as a patient) from a referring to a referred-to healthcare providers supported by a capability of an IT system. This capability is offered by technology provider(s) and is structured in terms of several business services, each of which provides a different value to a different type of consumer, as described below.

Value

Electronic referral (eReferral) capability provides *value* to:

- sending and receiving healthcare providers, in terms of streamlining their processes and providing better visibility of private and public healthcare options to offer to the patients, as available in a service directory
- patients, in allowing for improved care as the electronic referral allow faster, more reliable and consistent information exchanges between health providers, as well as improved consistency and security of information.

Parties

The following are the primary parties involved:
- a GP or specialist fulfilling referrer provider role (of a Consumer type)
- specialist, allied health provider or aged care assessment teams, fulfilling referred-to role (also of a Consumer type)
- patient (individual) – which, also has a consumer role, although depending on eReferral implementation, it can have varying degrees of participation in the service;
- IT service providers, who offer eReferral technology services

In order to better describe the concepts of parties and their relationship to roles, we use the following diagram that informally depicts these concepts, as part of a broader RM-ODP enterprise modelling concept referred to as *community*. Community is used to model collaborations, within one organisation or across organisations, each of which provides a boundary of the enterprise policies that apply to the parties fulfilling the roles in the community. At this stage of ontology specification, the concept of community is not formally introduced but could be added in case some services, in particular patient care coordination, may require it.

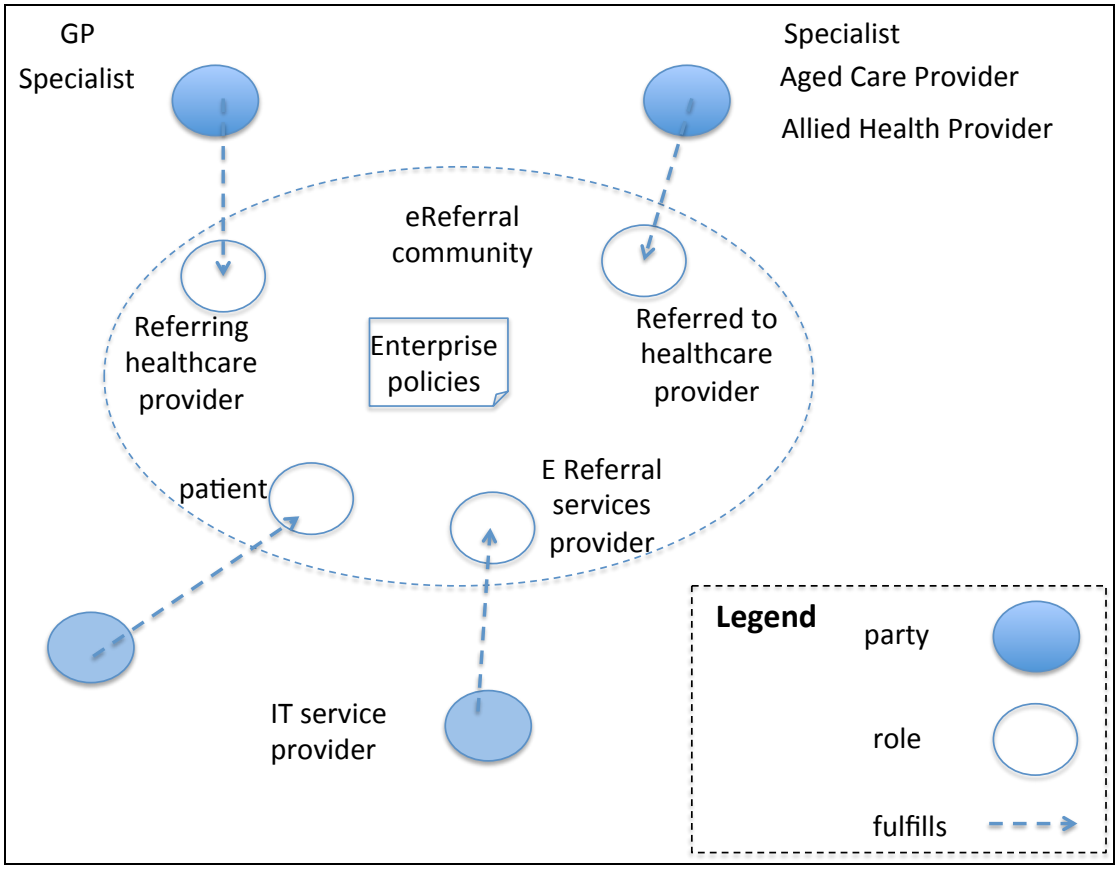

Figure 11 Parties and Roles in eReferral

Consumer roles

There are three types of consumer roles of eReferral,:
- referring healthcare provider
- referred-to healthcare provider
- patient

as identified in the use case above.

Their actions include:
- registering for eReferral service (by any accredited healthcare provider which can subsequently use the functionality of the eReferral business service),
- search for referred-to providers (by a referring healthcare provider)
- creating eReferral document artifacts (by a referring healthcare provider)
- booking an appointment (by patient)
- submitting eReferral documents/messages (by referring healthcare provider)
- sending back report (by referred-to provider to referring provider)

Note that the first two actions would require support of a healthcare provider/ services directory while the last two actions would require support of some kind of patient-controlled storage.

Note that a referring and referred-to providers will also include an administrator role whose purpose is to facilitate the establishment, use and management of the eReferral business service within the consumer organisation. For brevity, this is not shown in this example.

Provider roles

The main role involved in providing eReferral business services will be typically play by one or more IT service provider supporting eRefarral functionality.

Two kinds of provider actions can be distinguished.

One set is performed by an administrator role within the eReferral provider(s) to which a particular set of policies apply. These include business actions such as:

- defining a business service description
- advertising the service,
- making offers available and
- the formation of business service contract;

Second set of actions performed by the IT system (modelled as an active enterprise object) that implement the functionality of the eReferral business services and with which consumers of business services interact; from the consumer's point of view these actions can be regarded as business services in their own right, namely:

- register for eReferral service ,
- lookup of referred-to providers
- create eReferral document artifacts
- submit eReferal documents/messages
- sending back reports (by referred-to provider)
- booking an appointment (by patient)

In fact, the IT system would consists of a number of components, each with a distinguished role and acting on behalf of a specific party. There are potentially two parties with IT-only roles, namely the directory provider (register/search for referral target) and the exchange provider (storage). There are also IT systems used by the referrer and referred-to healthcare providers. There is potentially an IT system or storage device provided by the patient, if no "public" exchange provider is used.

Enterprise Policies

The following are examples of enterprise policies:
- patient is permitted to use the eReferral information artefact and to make an appointment with a referred-to provider other than the one nominated in eReferral
- referred-to provider has permissions to request additional information from the referring provider;
- the eReferral service provider (which may be fulfilled by one or more parties representing IT service providers) is obliged to respect obligation policies stated in its Service Level Agreements (i.e. a type of a Business Service Contract) with both the referring and referred to consumer roles, including various QoS constraints, price conditions and so on.

eReferal Information Artifact

The eReferral Information Artifact models an electronic representation of an eReferral document or message. It can be based on a standard eReferal template or a localised template. One possible implementation option may be a CDA referral template.

eReferral Business Process

An eReferral business process will implement an eReferral business service implementing functionality specified in the use cases in section 3.

Figure below depicts a simplified version of the eReferral registration business service (shown in light blue). This can be also regarded as a pre-condition for participation in eReferal service.

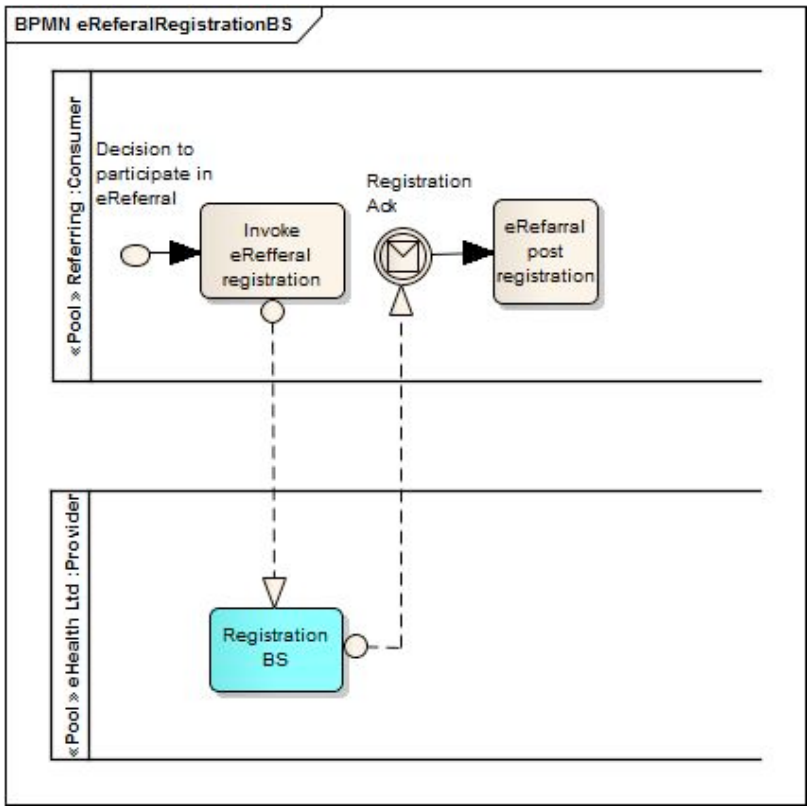

Figure 12 eReferral Registration Business Service

The remainder of the overall eReferral process shown in the diagram below, is implemented as a business process, which includes several business services. The diagram depicts activities of consumer roles and also business services provided by a provider role (shown in blue).

Note that the Appointment Scheduling business service is likely to be defined externally as this is not a service specific only to eReferral, so this service could have been identified as part of eReferral package during design time, or in a more advanced situation, it could be identified at run-time and invoked then, e.g. using the End Point Locator service (see below).

The diagram also depicts examples of enterprise policies that provide constraints on the actions of object fulfilling roles in the business process and they are included within comments as part of the appropriate BPMN pools.

Finally, the diagram illustrates the use of the End Point Locator (ELS) service, used as a means for locating end points of two computational service instances that implement business services of Appointment Scheduling or Submit Report business services. Although ELS is a computational viewpoint concern it is shown in this process to indicate the correspondences between the concepts.

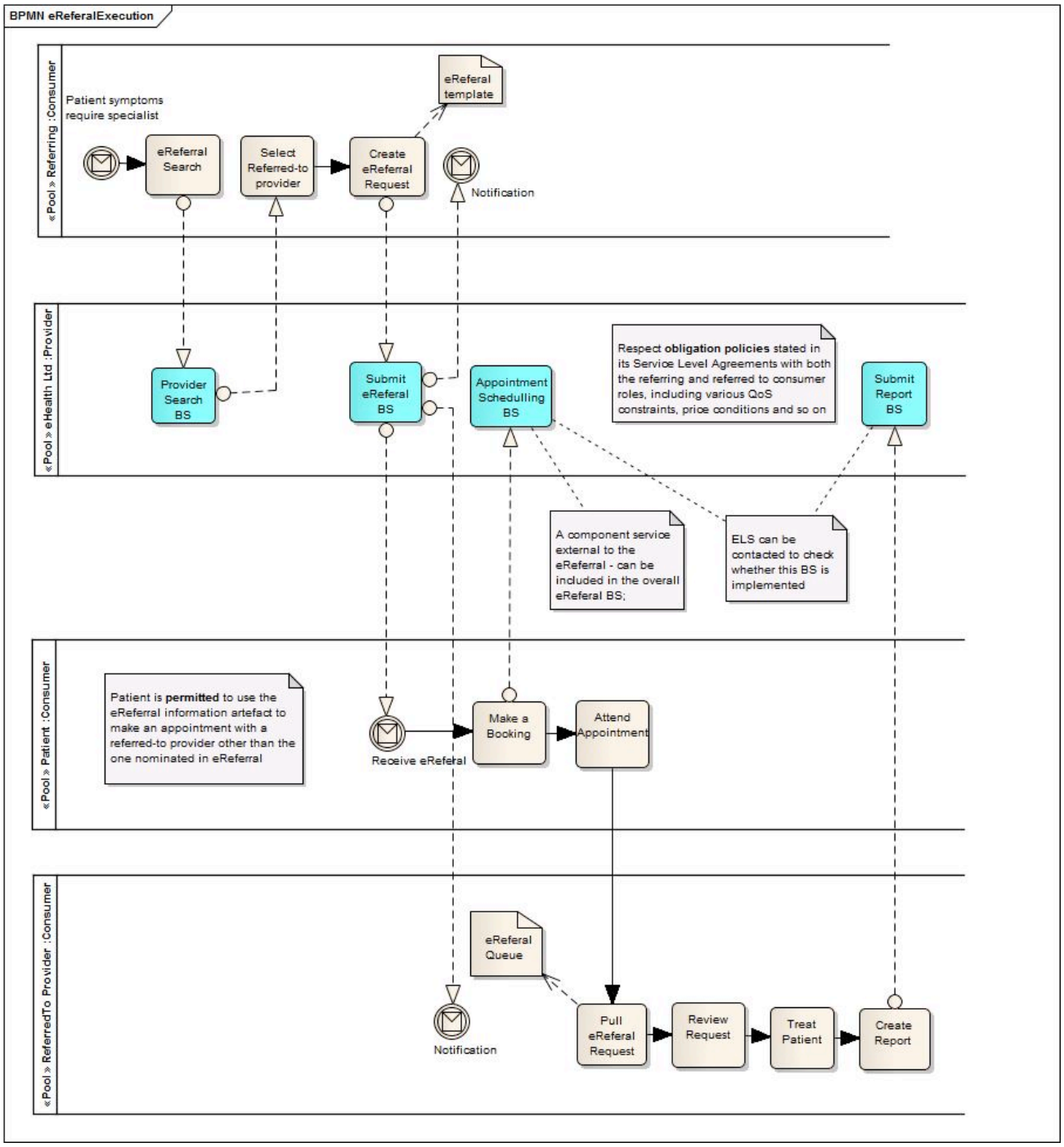

Figure 13 eReferral business services - overall

Computational viewpoint

Each business service identified above can be implemented using one or more computational services.

The example below illustrates how one such business service can be implemented, using one or more computational services. Note that a full eReferral specification will describe the implementation of each of the business service components it contains, but this level of detail is beyond the scope of this example.

Computational service and its interface

The *Provider Search* business service defined in the eReferal business process can be implemented using two computational services, whose interfaces are specified according to the HL7/OMG ServDir specification [ServDir] namely:

- *Search*
- *Retrieve Detail*

Note that a UML *provided* and *required* interfaces used in this specification are in effect an offerrer and requestor roles to the computational service – in a similar manner as is done in SoaML specification.

Requestor role

The *Searching Application* of the ServDir specification is a requestor role for each of the computational service above.

In this case the Searching Application is implemented by a computational object that realizes the eReferral search activity of the referring provider, but the internals of this computational object are not further elaborated.

Oferrer role

The ServDir Core component of the ServDir specification plays an Offerrer role for each of the computational services above.

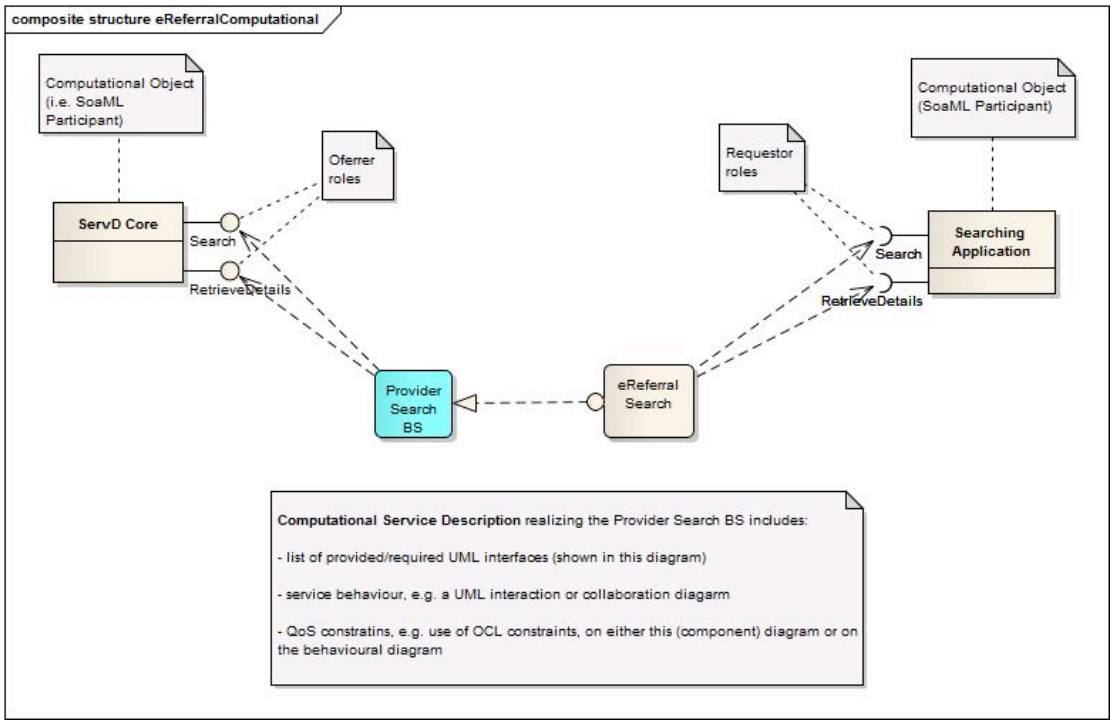

Figure 14 ServDir Core - use of computational concepts

The diagram below describes behaviour of ServD Core computational service.

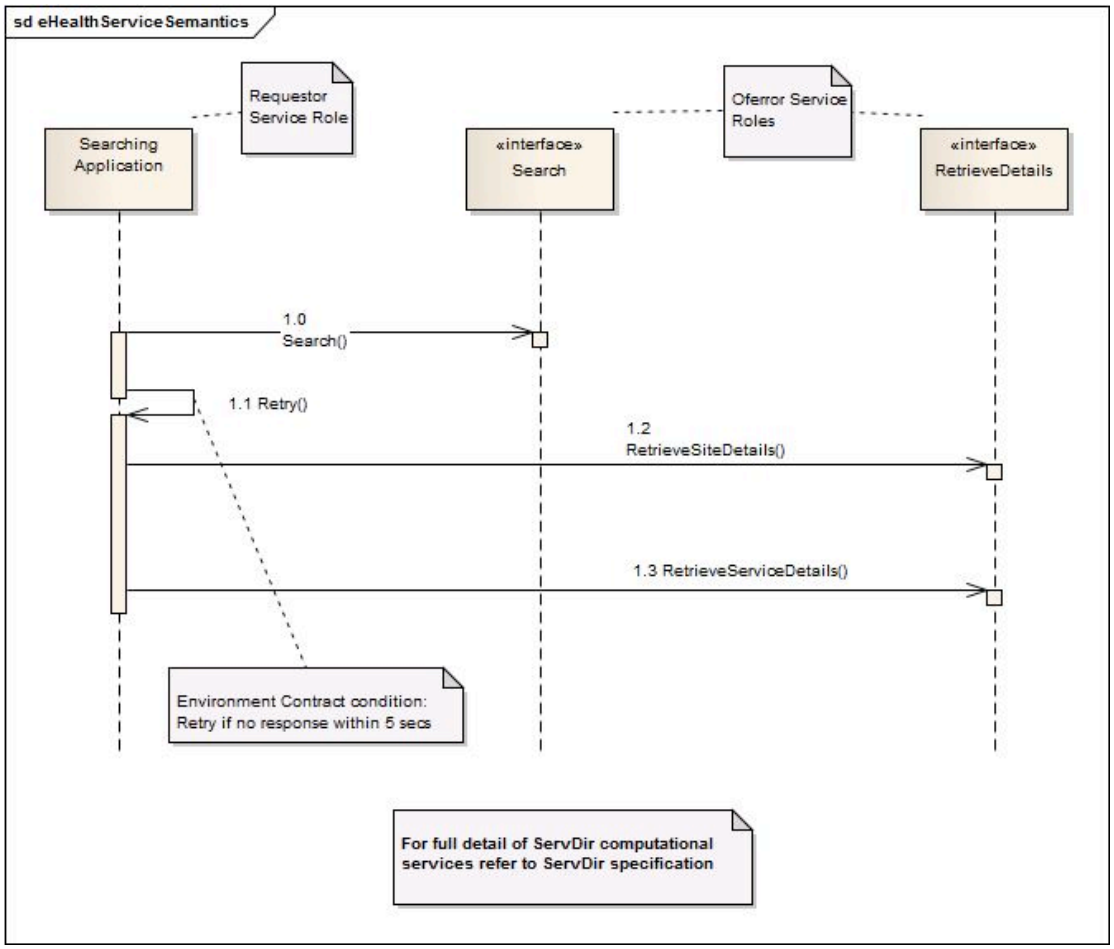

Figure 15; ServDir core - behavior

6. Examples of classifications

The service concepts described in previous sections provide many different properties of a service, while taking into account different stakeholders views. Many of these properties can be used to develop related classification of services. One can thus consider these properties as attributes or discriminators (in case of the availability of multiple choices of instance of a variable) for searching types or instances of services according to the selection criteria expressed therein.

Recall that an ontology defines a set of concepts and their relationships used to describe and represent an area of knowledge. A knowledge representation using an ontology based approach may be as simple as defining the taxonomy or classification schema for selected domain of concepts.

Defining a taxonomy (as one application of the ontology) and applying it to describe the hierarchy of services provides a consistent, logical and comprehensive methodology to

develop a framework for managing service inventory and streamline publishing, discovery, selecting and utilizing existing service assets in specific business context.

This section provides an example of how one coarse gain property, namely business service, can be used as a basis for one such classification.

The example is based on the *HL7 EHRS Functional Model* [EHR FM], representing a holistic definition of the systemic capabilities which are required to support various health care processes and workflows. It is fair to assume that individual capabilities which address need for specific domain can be logically grouped and encapsulated in form of *Business Services*. Depending on the purpose in the given context or business domain to which it applies *Business Service* are associated with one or more *HL7 EHRS FM* categories / subcategories.

Current EHRS FM R2 specification groups existing functional requirements into following high level categories:

Care Provision	Contains those functions that are required to provide direct care to a specific patient and enable hands-on delivery of healthcare.
Care Provision Support	Contains those functions required to support the provision of care to a specific patient to enable hands-on delivery of healthcare.
Population Health Support	Contains those functions required of the EHR to support the prevention and control of disease among a group of people (as opposed to the direct care of a single patient), usually with something(s) in common.
Administration Support	Contains those functions required in the EHR-S to support the management of the clinical practice and to assist with the administrative and financial operations.
Record Infrastructure	Contains those functions common to EHR System record management, particularly those functions foundational to managing record lifecycle (origination, attestation, amendment, access/use, archive…) and record lifespan (persistence, indelibility, continuity, audit, encryption)
Trust Infrastructure	Contains those functions common to an EHR System infrastructure, particularly those functions foundational to system operations, security, efficiency and data integrity, assurance, safeguards for privacy and confidentiality, and interoperability with other systems

Note that concept of function above is similar to the concept of behaviour in the service meta-model. The concept of service adds additional semantics to the concept of behaviour (and thus function), involving service user, service provider and policies as constraints on behaviour.

Each of EHRS FM high level categories branches into more granular and purpose specific categories. An example for selected branches in that taxonomy is illustrated in

the classification diagram (below). Each category may contain either more sub-categories or collections of services associated with that category.

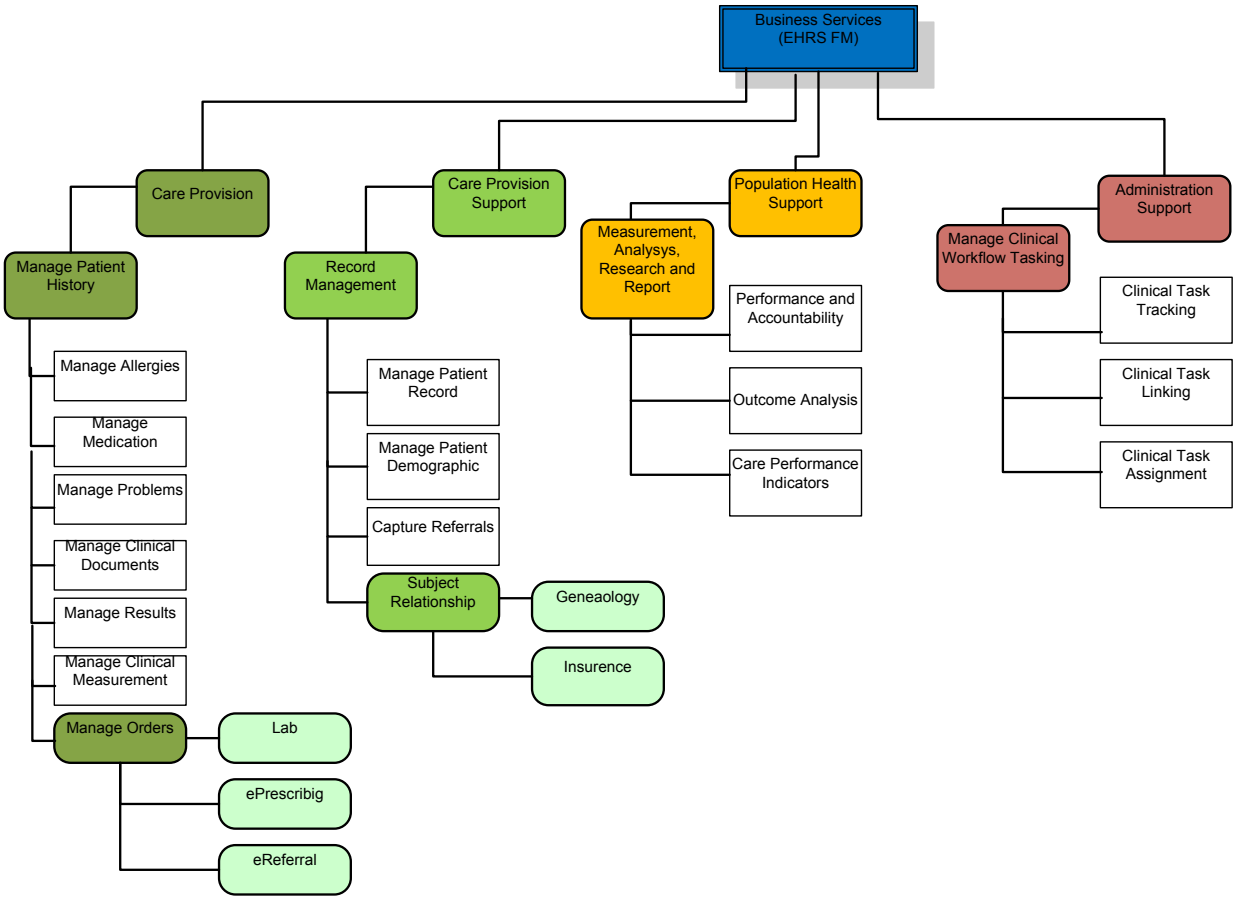

Figure 16: Example of business services classification - EHR-S Functional Model

While depth and richness of taxonomy depends on scope, objective and complexity of the domain that it describes, it is important that classification appropriately serves common interoperability and usability requirements for the particular domain.

7. Next steps

It is planned that the approach to ontology in this paper be further tested using further examples of eHealth services. For example it would be useful to:

- provide further detail to eReferral service in terms the implementation of other business services identified
- consider specific variants of this service in which a referred-to provider can be an allied health provider or aged care provider and develop a simple eReferral taxonomy reflected this.
- consider some other services such as a patient care coordination service and

- identify possible common components from eReferral service that could be exploited.

Finally, once the conceptual model is agreed and stabilised it would be beneficial to consider the use of formal ontologies such as OWL as a way of implementing this conceptual model. This would provide additional features such as run-time search for specific properties using a discriminator approach and perhaps supporting inferencing.

References

[ODM] Ontology Definition Metamodel: http://www.omg.org/spec/ODM/1.0

[ODP] ISO/IEC IS 10746-2, Information Technology — Open Distributed Processing — Reference Model: Foundations, 2009. Also published as ITU- T Recommendation X.902.

[ODP-EL] ODP Enterprise Language, DIS for ITU T Recommendation X.911| ISO/IEC 15414 Amd 1

[RAF] Reference Architecture Foundation for Service Oriented Architecture Version 1.0 Committee Specification Draft 04 / Public Review Draft 03

[SoaML] Service oriented architecture Modeling Language (SoaML) Specification, http://www.omg.org/spec/SoaML/1.0.1

[EHR FM] Electronic Health Record-System Functional Model, Release 2.0, http://wiki.hl7.org/index.php?title=May_2012_Ballot_Package

[ODP EL] DIS for ITU-T recommendation X.911 | ISO/IEC 15414 Amd 1, 2012-11-9, Output from the November 2012 meeting in Paris

[ServDir] Services Directory, OMG Document Number Health/12-08-1

[PASS] http://hssp-security.wikispaces.com/PASS+Architecture+Overview

[NEHTA] http://www.nehta.gov.au/connecting-australia/ehealth-architecture

[Infoway] https://www.infoway-inforoute.com/

[HISA] http://en.wikipedia.org/wiki/Health_Informatics_Service_Architecture

[Contsys] ISO/DIS 13940 System of concepts to support of continuity of care

Appendix

Viewpoint (on a system): a form of an abstraction achieved using a selected set of architectural concepts and structuring rules, in order to focus on particular concerns within a system.

RM-ODP defines the following viewpoints: enterprise, information, computational, engineering and technology.

Entity - any concrete or abstract thing of interest.

Object - A model an entity. An object is characterised by its behaviour and dually by its state. An object is distinct from any other object. An object is encapsulated, i.e. any change in its state can only occur as a result of an internal action or as a result of an interaction with its environment.

Depending on the viewpoint, the emphasis may be placed on behaviour or on state. When the emphasis is placed on behaviour, an object is informally said to perform functions and offer services (an object which makes a function available is said to offer a service). For modelling purposes, these functions and services are specified in terms of the behaviour of the object and of its interfaces. An object can perform more than one function. A function can be performed by the cooperation of several objects.

Note that object is refined if each of the ODP viewpoints, so we have enterprise object, information object, computational object, engineering object and technology object.

Action – Something which happens. Every action of interest for modelling purposes is associated with at least one object. The set of actions associated with an object is partitioned into internal actions and interactions. An internal action always takes place without the participation of the environment of the object. An interaction takes place with the participation of the environment of the object.

Note that "Action" means "action occurrence" not "action type". That is to say, different actions within a specification may be of the same type but still distinguishable in a series of observations. Depending on context, a specification may express that an action has occurred, is occurring or may occur.

Policy - A constraint on a system specification foreseen at design time, but whose detail is determined subsequent to the original design, and capable of being modified from time to time in order to manage the system in changing circumstances.

NOTES
1 Policies can be applied in any viewpoint; examples are an enterprise delegation policy, a computational persistence policy or an engineering scheduling or quality support policy.
2 The expectation of change is fundamental to the concept of policy, and a rule that does not envisage change is not a policy.
3 Policies may be expressed in terms of obligations, permissions or prohibitions, but this is not necessary for simple policies.

