

Product Line Scoping for Healthcare Information Systems Using the ISO/IEC 26550 Reference Model

Juan Carlos Herrera¹, Francisca Losavio², Oscar Ordaz^{2,3}
jchr1982@gmail.com, francislosavio@gmail.com, oscarordaz55@gmail.com

¹PFG Informática para la Gestión Social, Universidad Bolivariana de Venezuela, Caracas, Venezuela

²Escuela de Computación, Laboratorio MoST, Universidad Central de Venezuela, Caracas, Venezuela

³Escuela de Matemáticas, Universidad Central de Venezuela, Caracas, Venezuela

Abstract: The main goal of this work is to present a process for Software Product Lines (SPL) scoping focused on early consideration of software product quality. Product Line Scoping is the first phase of SPL Engineering (SPLE), in the Domain Engineering (DE) lifecycle, where the SPL long-term feasibility must be determined. The PLScoP process proposed here for SPL scoping, is an adaptation of the general PL Scoping phase defined in the new ISO/IEC 26550 standard describing a reference model or framework for SPLE, and it concerns three main stages, Portfolio Scoping, Domain Scoping and Asset Scoping. In this work, the complete PLScoP is outlined, but only the Domain Scoping phase will be detailed and applied. General guidelines on “What to do”, as the majority of standards offer, are defined in ISO/IEC 26550. Our PLScoP complements this framework by presenting the “How to do”, offering precise techniques and artefacts to be applied and constructed, and by considering early and systematically quality issues; this will allow reduction of the development effort in the subsequent DE phases, Domain Requirements Engineering and Domain Design, where the major effort is concentrated SPL development workload. The Domain Scoping step of PLScoP will be applied to the Healthcare Information Systems domain.

Keywords: Software Product Lines; Product Line Scoping, PLScoP; Domain Scoping; ISO/IEC 26550; Software Quality; ISO/IEC 25010; Healthcare Information Systems.

I. INTRODUCTION

Software Product Lines (SPL), or simply Product Lines (PL), is an approach that provides a way of massive personalization of individual solutions from a repository of reusable software assets, in a particular domain; it is inspired in the Fordism technique used to increase production while lowering costs in early 20th century automotive industry. The term domain is often used in reference to a particular knowledge area; an application domain denotes any aspect where computing can be applied [1]; a *domain* is defined by Bérard as the minimal set of properties describing precisely a family of problems in which a computational application or system is involved for their solution [2], and it is the definition adopted in this context. The problem of software development based on reusing components or a core of software elements, favouring efficient and reliable development is not new [3][4][5][6] and it is a complex problem not yet completely solved in academic or industrial practices; moreover, there is still a huge gap between research results and their industrial application [7][8]. The SPL development, also called SPL Engineering (SPLE) [9], aims to promote maximal reuse exploiting common elements in similar products of the SPL family in a particular domain. The main idea, but not an easy task, is to capture the essential common

elements and possible variable issues to construct an evolutionary SPL, since it must manage changes and last over time to provide an economic payback. Instead of describing a single system, the SPL model describes a set or family of software systems, products or applications. Complex domain analysis must be achieved in order to specify and delimit the family that can be developed from the core of assets, with their commonality and variability [3][4][10].

This work is framed within the Domain Engineering (DE), first lifecycle of SPLE, where the major development effort is concentrated in constructing the SPL Reference Architecture (RA) and Core Asset Repository (see Figure 1). The main goal of this paper is to present and apply a PL Scoping Process (PLScoP), with precise techniques and artefacts, adapting the PL Scoping phase or initial DE phase, defined in the Reference Model of the new standard ISO/IEC 26550 [7][8][11]. Notice, in general that standards or general frameworks specify the “What to do”, however the details of the “How to do” have always to be properly defined; even if in [11] a list of available techniques and methods are provided, how to combine or adapt them to the SPL context to achieve a particular activity is not specified. Our work complements the SPL standard framework, offering explicit techniques to be applied to

perform PL Scoping activities. In particular, for the study of domain existing products in the Portfolio Scoping step, a bottom-up process [12] can be considered, providing as output an initial candidate architecture; moreover, the ISO/IEC 25010 quality model [13] is used to specify quality properties related to functional (FR) and non functional requirements (NFR), and BPMN ¹[14] is proposed to specify the domain model. PLScOP emphasizes software product quality assurance at early stages of SPLE; quality properties and their traceability w.r.t. FR and NFR requirements has in general been poorly considered in SPL development, playing however a major role in the SPL variability and evolutionary capacity. In this work, the Domain Scoping phase will be applied to the Healthcare Information Systems domain to obtain a first draft of a domain model.

Usual approaches focus more on products' features directly perceived by users [15], which is not the case of quality properties that appear as "implicit functionality" often later on during the SPL development; nevertheless, quality requirements are responsible of most of the SPL variability at the moment of deriving concrete SPL products during the Application Engineering (AE) or second SPLE lifecycle [16][17]. Figure 1 shows the ISO/IEC 26550 SPL Reference Model [11]. If quality properties are not considered early in the DE lifecycle of the SPL development, the global quality of the products derived from RA cannot be guaranteed, compromising organizational goals and the whole SPL ROI (Return of Inversion).

Besides this introduction and the conclusion, the structure of this paper is the following: some related works are discussed in Section 2; Section 3 describes the SPL Domain Engineering guidelines of the ISO/IEC 26550 standard; Section 4 presents PLScOP, as an adaption of the PI Scoping phase [11]. Finally, Section 5 is dedicated to the application of the Domain Scoping step of PLScOP to a case study in the Healthcare Information Systems domain.

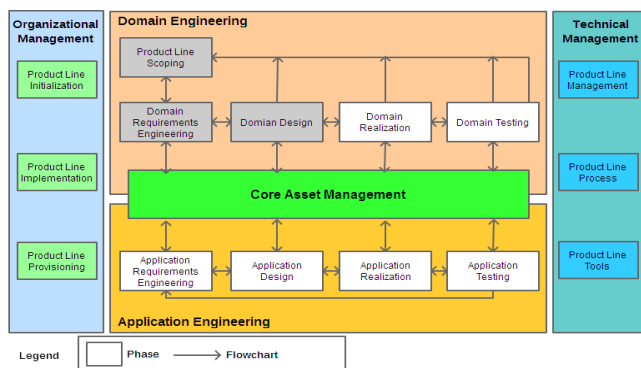


Figure 1: ISO/IEC 26550 Reference Model for Software and Systems Product Lines

II. RELATED WORKS

In what follows, some related works relevant to the subject of SPL scoping, context of this research, are discussed:

An SDR (Systematic Documental Review) [18] is presented in [19] to identify best practices, challenges and limitations of the

SPL scoping phase; this study points out that scoping is important and even essential for the achievement of the product line. Its main goal is to determine the feasibility of the SPL, identifying crucial aspects such as products that will conform the SPL, risks, potential reuse and costs to implement main assets. An important question related with our work is the following: *How is the SPL development affected by the scoping phase?* It is clear that the artefacts produced during this phase are input to the DRE phase and should reduce the effort also in subsequent phases. The study claims that the majority of the approaches reviewed do not have a clear understanding of the relation between scoping and DRE phases. In this sense, our proposition to use Bjørner's domain modelling [20] in PL Scoping will fill this gap.

A characterization of the benefits and weakness perceived in the Domain Scoping and DRE phases using the agile method is presented in [21]. The observed variables were the stakeholders' motivation, effort, communication and collaboration, iterative and adaptability aspects of the process, requirements and technological volatility. A question arisen from this study is: *How the effort to perform SPL scoping is characterized by the stakeholders?* The effort is measured in man-hours, and the answer was "great", due to the huge amount of domain documentation (often incomplete and inconsistent) that has to be analysed to capture enough domain knowledge to provide acceptable asset scoping and product portfolio, and the lack of domain experts. Activities identified for the scoping phase were: *Pre-scoping, Domain scoping, Product scoping, and Assets scoping.* Bottlenecks found were: the absence of domain and product experts to capture products' main functionalities, and the clear identification of features and their granularity. Moreover, variables stakeholders' communication and collaboration, iterative and adaptability aspects of the process, were also found to affect the effort. Our research do not use an agile method, nevertheless we claim to reduce the scoping effort by introducing the Bjørner's business process-centric technique of domain modelling [20], modified by introducing the quality intrinsic descriptor, to specify quality properties related to business processes functions at this very early stage.

A *framework* is described in [22] for SPL developments including the SPL scoping phase, to specify what the SPL can or cannot do, by defining those behaviours or aspects that will be incorporated or eliminated from the SPL. Scoping defines the long-term feasibility of the SPL; it starts with a broad document, which is being refined in the measure that more domain knowledge is captured. The goal is to establish a limit for the SPL to achieve business and market goals. Our research aims to establish a "structure" for this document (input/output artefacts and their structure, techniques used, etc.), since a precise definition was not found in the literature; it will be used as a valid input to reduce the effort in the DRE and DD phases. Hence the scoping document will result an important asset.

The proposed framework suggests the following activities for SPL scoping:

- Workshops and interviews with the stakeholders
- Examine existing products (bottom-up approach [23])

¹ Business process Modeling Notation

- Context diagrams
- Develop a matrix of attributes/products
- Develop SPL scenarios

where a context diagram represents relevant entities related with the SPL w.r.t the users of the products; an attributes/products matrix is used to define the variability of the SPL; scenarios are used to identify interactions that are common to all products and those that are variants.

The Bjørner's domain modelling [20] used in our Domain Scoping step is compliant with most of the above requirements, excepting for - Examine existing products, which is part of the Portfolio Scoping step, where a bottom-up approach will be used [12][23]; however, this point will not be treated here, being the object of an on-going work.

In [24], a specific method based on PuLSE, Pulse-Eco, is presented. This process uses a DE bottom-up development strategy; we are actually focusing on a DE top-down approach [23] combined with a bottom-up approach to study existing products [12], which is recommended in SPL scoping by the new standard [11]; this study however, can be performed on the basis of an "agile" bottom-up process [25].

In [26], three domain engineering approaches were compared, two of them related to SPL, namely, Pohl et al. [9] and the ISO/IEC standard [11], the other one, the classic Bjørner's DE approach [20] strongly based on business processes modelling. From this comparison, [20] was found suitable to integrate the Domain Scoping step of the ISO/IEC 26550 SPL Reference Model [11], and quality was included as a new facet; however, since quality is involved in all facets used in [20] to specify the domain model from different stakeholders' viewpoints; in our present work, quality is specified as an intrinsic facet descriptor and the whole approach is illustrated with a complete case study.

III. SPL DOMAIN ENGINEERING GUIDELINES WITH THE NEW STANDARD ISO/IEC 26550

The complete DE lifecycle will be briefly described in what follows, according to the ISO/IEC 26550 guidelines [11]; however in this work, only the Domain Scoping step of the PL Scoping phase will be treated in details.

According to the SEI² definition [22][27][28], PL Scoping is itself a core asset. In SPL development, scoping is a fundamental activity that will determine the long-term viability of the SPL. Like scoping in general, PL Scoping determines what's "in" and what's "out" of the SPL. The scope definition identifies those entities with which products in the SPL will interact (that is, the product line context), and it also establishes the commonality and sets limits on the variability of the products in the SPL. The scope definition usually begins as a broad, general draft document that is refined as more knowledge is captured and more analysis is performed. For example, for an SPL for a Web-based system, browsers would definitely be "in". Aircraft flight simulators instead, would definitely be "out"; the PL scope may not come into sharp focus all at once. The goal of the scope definition is to draw the

boundary between "in" and "out" in such a way that the SPL satisfies its business and market goals.

Five phases are considered in ISO/IEC 26550; notice that phases and sub-phases can be performed in the order established by the company or organization requiring the SPL [11].

A. Product Line Scoping (PL Scoping)

It defines SPL boundaries for DE, envisioning major common and variable features to all products within the SPL. Economical aspects are analysed and the commercialization of the SPL product family is planned; PL Scoping is responsible of the whole SPL management and consequent evolution: it involves 3 sub-phases:

1) *Product Portfolio Scoping*: 1. Identify products that the SPL should be developing, producing, marketing and selling (product "roadmap"); 2. The study of common and variable feature of existing products should provide guidance to meet business objectives and face SPL evolution; 3. Schedule for introducing products to the market. It is input to Domain Scoping.

2) *Domain Scoping*: identify and bound functional or organizational areas that SPL will impact to provide sufficient reuse potential to justify the SPL creation.

3) *Assets Scoping*: identify the boundaries of core assets, providing a first glance at common and variable assets. It identifies reusable assets and calculate the cost/benefit estimated from each asset in order to determine whether an organization should launch an SPL.

Major outputs of PL Scoping are: the asset proposal; it includes major assets (functional areas and high-level common and variable features of all SPL products) that will be included in an SPL with their quantified costs and benefits estimation results. The features defined in the asset proposal directly affect Domain Requirements Engineering (DRE) and Application Requirements Engineering (ARE) shown in Figure 1. More than one asset proposal can be made to find out an optimal set of products and assets. The asset proposal defines also a schedule for delivering specific products to customers and for bringing them to market.

B. Domain Requirements Engineering (DRE)

It has to adhere to the specification of the SPL's high-level features provided by PL Scoping. Based on these features, it creates detailed common and variable requirements sufficient to guide subsequent Domain Design (where the SPL RA is designed), realization and testing phases. It involves 5 sub-phases: - Domain requirements elicitation, - Domain requirements analysis, - Domain requirements specification, - Domain requirements validation, and - Domain Requirements Management (to handle changes in requirements).

C. Domain Design (DD)

It draws upon the specifications to develop an SPL architecture that enables the realization of the planned commonality and variability within the SPL. The main goal is to produce the RA, defining general SPL structure and

² Software Engineering Institute, MIT, Carnegie Mellon

textures. RA reflects additional internal variability introduced by technical solutions besides the external variability, i.e., commonality and variability in the user's perceived requirements. It involves the sub-phases: - RA design, - RA evaluation (quality assurance technique), and - Domain Design Management (to handle changes in the RA design).

D. Domain Realization/Implementation

Design and implementation of reusable loosely coupled software components and configurable interfaces, implementing common and variable artefacts offered by RA. Domain realization includes configuration mechanisms to realize variability domain implementation, such as building and buying components supporting the RA infrastructure. They are not yet executable applications.

E. Domain Testing/Validation

It validates the domain artefacts created in previous phases and generates domain test artefacts that can be reused later on in Application Testing. Testing in this context means review, validation and verification of artefacts as well as eventually testing some available implementations.

In this work, only the Domain Scoping sub-phase, within the PL Scoping phase will be applied to a case study, to illustrate our approach.

IV. THE PL SCOPING PROCESS: PLScoP

The adaptation of the ISO/IEC 26550 PL Scoping phase is constituted by the PLScoP process that is outlined in what follows:

A. PLScoP Context

The PL Scoping guidelines of the ISO/IEC 26550 standard were completed by integrating to the Domain Scoping sub-phase, the stages described by Bjørner [20] for classic DE for single software systems, not considered for an SPL context. However, they provide a nice technique to specify domain knowledge, based on facets and stakeholders' viewpoints [26].

A *facet* is defined in [20] as a finite set of generic forms of describing a domain from different stakeholders' perspectives or viewpoints, namely, business processes, support technology, management & organization, rules & regulations, human behaviour, and intrinsics; complete definitions of these terms were presented in [26].

The facet notion is not new in Software Engineering [29]; according to [20], each facet represents a view of the domain from different perspectives; the union of these views conforms the complete domain view, called Domain Model; moreover, the special *intrinsics facet* is a facet containing descriptors or attributes (entity, function, event, behaviour) necessary to describe all the other facets (see Table I); the intrinsics notion has allowed us to include software quality, specified by the ISO/IEC 25010 standard [13], as a new intrinsic descriptor. Software quality is then considered to specify all other facets.

Table I: Intrinsics to Describe all Domain Facets with the New Quality Descriptor

Intrinsics descriptors	Description
Entities	Represent the phenomena and concepts of the domain
Functions	Operations (actions) performed on the entities
Events	Imply changes in entities by function invocations, i.e., actions in the domain
Behaviour	Sequences of actions and events affecting domain entities
Quality	Specified by the standard quality model ISO/IEC 25010 [13]. It is associated to the Business Processes facet as quality goals (obtained from NFR) required by FR (functions, activities or tasks), to facets Support Technology as quality supported by architectural styles, patterns or mechanisms involved, and Rules & Regulations as quality required by domain business rules [26]. Product quality must be specified to guarantee that SPL products will hold acceptable industrial quality levels.

In particular, this work involves Business Processes, Support Technology, and Rules & Regulations facets, since the final aim of DE is to build an SPL reference architecture; Management & Organization and Human Behaviour facets can be also described in terms of quality, using models, such as CMMI³ where organizational practices are deeply involved, but this topic is outside the scope of the present work.

Notice that in the Domain Scoping adaptation from [11] integrating Bjørner's domain development [20], a huge number of business processes can be derived from the so called Domain Description Units (DDU) specifications from the declarations of different stakeholders groups expressing their viewpoints. However, this complete specification is outside the PL Scoping spirit, which aims to offer a quick "glance" of the SPL feasibility and limitations, hence only the presentation of few basic modelling elements are considered sufficient to illustrate the "How to do" of our process.

B. PLScoP Context

The adaptation of the ISO/IEC 26550 PL Scoping guidelines, is presented in what follows; notice that Asset Scoping and Portfolio Scoping are left as the last steps, since according to [11], the order in which they are executed depends on the organization building the SPL, and we have major interest here in Domain Scoping; however, if a study of existing market products for the domain has to be done, Product Portfolio Scoping should be executed first to perform this study, and its output should be input to the Domain Scoping step.

PLScop process

1. *Domain Scoping:*
Input: Domain informal documentation provided by the organization requiring the SPL, Domain Quality Model (DQM) specified by ISO/IEC 25010 [13].
 - a) *Stakeholders Identification:*

³ Capability Maturity Model Integration

Input: visits, interviews, workshops, questionnaires (specific techniques for each one of these activities should be specified).

- Identify groups of stakeholders with similar interests in the organization requiring the SPL.

Output: list of stakeholders' groups type: text

b) *Domain Acquisition:*

Input: List of Stakeholders' groups

- Capture and gather information from stakeholders into *declarations* to build Domain Description Units (DDU) for each stakeholder viewpoint;

Output: DDU type: table

c) *Domain Analysis:*

Input: DDU

Analyse DDU, study possible inconsistencies, and business processes are extracted first from DDU [20], to specify the Business Processes facet from a stakeholder viewpoint relevant to the domain; it is represented by a table, UML [30] and BPMN diagrams, using intrinsic facet descriptors. Quality issues are included as a new intrinsic facet descriptor.

Then other facets, relevant to the domain, are also specified according to domain specific stakeholders' viewpoints; for example, in our case they will be *Support Technology* from the *Domain Engineers viewpoint*, and *Rules & Regulations* from the *Directors of healthcare governmental institutions viewpoint*.

The business processes extracted from the DDUs for these stakeholders' viewpoints are identified among the behaviours present in the facets, and specified as new Business Processes facets, considering the respective stakeholders' viewpoint.

Output: Facets specifications type: table; UML and BPMN diagrams for business processes

d) *Domain Modelling:*

Input: Facets specifications

A domain description is obtained from all the facets specifications by intrinsics; this document should be focused on commonality and variability of the entities involved. According to [20] a domain model is a meaningful domain description; it will be represented integrating the BPMN specifications for all business processes considered.

Output: Domain Model type: BPMN. To have a more general specification of the domain model, including all facets, an ontological approach could be considered.

2. *Asset Scoping*

Input Domain Model specification

Information on core assets will be extracted from domain model to conform the SPL Core Asset Repository, which will be informally described into the Asset Proposal document, analysing here also economical factors for the SPL feasibility.

Output: Asset Proposal document

3. *Product Portfolio Scoping*

Input: Available documentation on existing products

Existing products in the domain are assumed to exist; they will be studied to infer about the SPL products that can be developed, main capabilities and limitations; this study could be preformed applying an extractive or bottom-up process to construct automatically a draft *candidate architecture (CA)* [31] using reengineering techniques to handle similarity analysis of the products' components [12]. Notice that since in general widely used products on the

market are considered, results of the existing products study are included into the market study on the SPL feasibility.

Output: Product Roadmap: CA: type: graph or UML diagram

Notice that the Product Roadmap artefact includes the candidate architecture artefact which is a first broad draft of the RA that can be built [12]. This possibility should be considered, to have also an additional input to the DRE phase, thus reducing the required effort there. This initial candidate architecture has imbedded the domain knowledge, extracted from existing products about main common and variant components, which can be enriched with the more general information obtained from the Domain Model. It will become also part of the Asset Proposal.

C. *PLScoP: Advantages and Limitations*

If a first candidate architecture can be constructed or is available for a domain, our PLScoP, and in particular the Portfolio Scoping step, can be transformed into a process to perform the RA evolution, i.e., management of changes, in DRE and DD phases. The analysis of existing products is not an easy task, and it depends much on the available market products documentation, requiring a considerable effort to apply reengineering techniques [12]. If this draft architecture is not available, the Domain Scoping step of our PLScoP is still crucial to construct a detailed Domain Model that will help to delimit clearly the SPL scope and functional and non functional granularity, to reduce the effort in the subsequent DE phases where the RA is built.

The advantage of our process is to have combined top-down and bottom-up approaches to specify the domain: top-down is considered in the philosophical Børner's approach [20], which starts with the domain decomposition into organizational business processes specified by intrinsics descriptors, and it is generally used in SPLE [9][11]. The bottom-up approach is proposed to be used in the Portfolio Scoping step also in [9][11], to have a broad picture of the SPL present and future products, by studying domain existing products in the organization proposing the SPL or in different organizations with similar domains. In this sense, our proposition takes advantages from the combination of [9] and [11], reducing general weaknesses of DE lifecycle.

V. APPLICATION TO THE HEALTHCARE INFORMATION SYSTEMS DOMAIN

In what follows, the Healthcare Information Systems (HIS) Domain for the SPL will be briefly discussed to be applied to illustrate PLScoP.

A. *SPL Domain: Healthcare Information Systems (HIS)*

HIS [32] are software intensive systems, i.e., complex integrated information systems, generally located in different and distant institutions and with mandatory (priority) NFR, such as interoperability, availability and security. Interoperability (technical), the HIS crucial quality property for *Electronic Health Records (EHR)* management and sharing, is the ability of two or more systems or components to exchange information and to use the exchanged information; semantic interoperability refers to use a common terminology or

language to communicate systems; process interoperability incorporates business processes and healthcare professionals must standardise business rules to ensure that health information is properly recorded, such as the transfer of information between systems is consistent and complete [33]. The general architecture is a hybrid event-based style, SOA⁴/Layers [12][31], see Figure 2 (from Wikipedia). HIS must facilitate transparent sharing of different kinds of medical information such as EHR and laboratory&imaging results, offering also telemedicine services that can be performed online at remote locations, with wide support of information technology. The use of standards such as HL7, HL7 CDA, LOINC⁵, and DICOM⁶ are mandatory for interoperability of EHR and laboratory&imaging results [25][32][33][34].

Nevertheless, in actual medical practice, SPL for HIS have not yet been completely defined, developed and adopted; the lack of agreement on medical standards and psychosocial issues makes difficult the interoperability of EHR, and HIS general adoption is still difficult, even if specific laws and regulations towards these goals have been promulgated worldwide.

B. Software Quality Modeling in SPL Domain Scoping

Quality has been defined in general as a level of excellence, conformance with specifications, requirements satisfaction, defect free, accomplishment of customer demands [35], and also related to human aspects such as usability and satisfaction [36]. The early specification in PL Scoping of software quality will facilitate to map this quality into all subsequent DE activities; this information on domain quality will be specified as a Quality Model (DQM), reflected into the Domain Model, and registered in the Asset Core Repository. As we have already pointed out, quality assurance is crucial in an industrial software production context to guarantee SPL evolution and the massive assets reuse, impacting on the quality of all products of the SPL family [37].

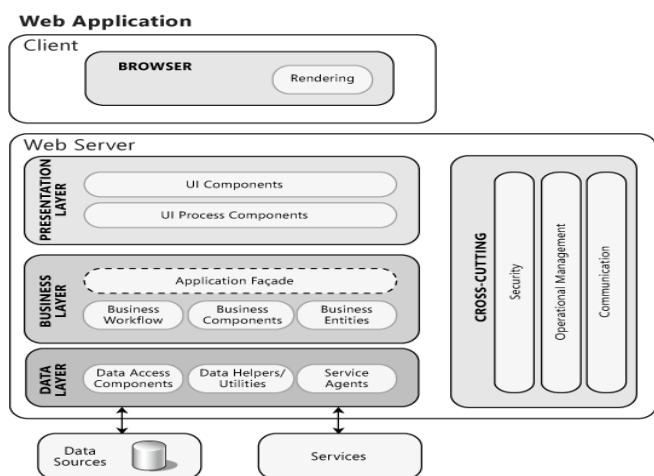


Figure 2: Hybrid Architecture SOA/Layers

The ISO/IEC 25010 Quality Model standard [13] to specify software product quality is part of the SquaRE series of

⁴ Service-Oriented Architecture

⁵ Logical Observation Identifiers Names and Codes

⁶ Digital Communication in Medicine

standards of the International Standards Organization (ISO). This series focuses on quality, requirements and evaluation of software products. It states compatibility with other ISO standards on quality measures and process quality [38]. The central document of the series is the known ISO/IEC 25010 Quality Model [13], describing a hierarchical framework where quality is decomposed into levels of characteristics, sub-characteristics, etc., until the attributes or measurable elements. The Product Quality Model will be used here, since we are in a software development context; Figure 3 [25][32], shows its adaptation to specify the HIS domain quality w.r.t. the SPL family of software products.

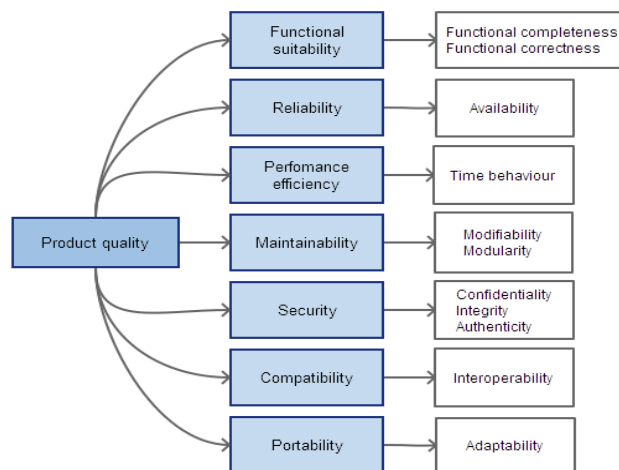


Figure 3: ISO/IEC 25010 HIS Domain Quality Model (HIS-DQM)

C. Case Study Scope

For this work, the HIS domain for the SPL is restricted to its basic functionalities (EHR-HIS), namely EHR management, patient attention with patient appointment scheduling and capture of demographic data, emission of medical reports, basic administrative services for patient attention; imaging and laboratory services, hospital rooms management, nursing services, urgencies, general hospital administration, etc. will not be considered here [25][34]. Note that the example has been simplified, since only HIS elements at a high granularity level have been treated, avoiding low-level details to facilitate the illustration of our approach, following also the spirit of the PL Scoping phase.

On the other hand, the main facets that will be specified in this work are Business Processes, Support Technology, and Rules&Regulations. The viewpoints considered are: Doctors' group for Business Processes, Domain Engineers' group for Support Technology, and Directors of healthcare governmental institutions for Rules&Regulations. Facets, stakeholders' groups, and viewpoints were selected on the bases of the SPL DE development process, whose main goal is to design a Reference Architecture.

D. Application of PLScop to the HIS Case Study

Following the basic steps defined in the *Domain Scoping* sub-phase, we have:

1) *Stakeholders Identification*: The technique used consists in doing the expertise of domain engineer extracted from visits,

interviews, workshops or questionnaires. Different viewpoints are identified to conform the stakeholders' groups: Doctors, Domain Engineers and Directors of Healthcare governmental institutions.

2) *Domain Acquisition*: The technique used consists in statements or Declarations formulated by stakeholders to illustrate their viewpoints.

DDU construction:

The declarations are grouped into the *Domain Description Unit (DDU)* and they are represented textually as a table (see Tables II, III, and IV) for each one of the identified stakeholders' groups.

Table II: DDU for EHR-HIS Domain from the Doctors' Viewpoint

EHR-HIS Domain Description Unit	
Viewpoint	Stakeholders' Group: Doctors
<i>Declarations</i>	
Patient is attended in hospital under scheduled appointment	
If it is the patient first appointment, a new EHR must be created by nurse, else nurse retrieves patient existing EHR	
Patient EHR is accessed by doctor	
New exams and laboratory results can be added to patient EHR by doctor, if it is the case	
Diagnosis and medical orders for patient are produced by doctor to conclude medical attention	
A new appointment is scheduled by nurse if required by Doctor	
Medical equipment and material can be required by doctor to provide adequate medical attention	

Table III:DDU for EHR-HIS Domain from the Domain Engineers' Viewpoint

EHR-HIS Domain Description Unit	
Viewpoint	Stakeholders' Group: Domain Engineers
<i>Declarations</i>	
HIS is supported by an hybrid architecture SOA/Layers	
The HIS architectural style supports mainly modifiability, interoperability, performance (time-behaviour), and security services are provided by Internet protocols, crosscutting all layers; availability however depends on internet connection;	
Transmission Layer is managed by a Web Server that communicates all other layers	
Clients access HIS by a browser in Presentation Layer, which connects to the Transmission Layer via a Web Server	
Medical units should have wide range internet and intranet access	
Main EHR-HIS functionalities must be supported: patient appointment services, EHR management and emission of medical reports.	

Table IV:DDU for EHR-HIS Domain from Directors of Healthcare Governmental Institutions' Viewpoint

EHR-HIS Domain Description Unit	
Viewpoint	Stakeholders' Group: Directors of Healthcare Governmental Institutions
<i>Declarations</i>	
Digitalize EHR with standard format to achieve sharing among doctors and national and international healthcare institutions	
Have Database of national and international specialists	
Develop a Web platform to manage on-line appointment services	

3) *Domain Analysis*: The technique used consists in the initial domain knowledge is captured from the DDU's (see Table II, III and IV, and described textually in Table III as business processes.

Facets specification:

In Table V, business processes specific to the Business Processes facet are derived first [20] from DDU, considering the Doctors' viewpoint. Tables VI and VII will specify business processes specific to facets Support Technology and Rules & Regulations, respectively; only one process will be specified for each stakeholder's group viewpoint to abridge this presentation.

Table V: Business Process Derived from the DDU of Doctors' Viewpoint

Viewpoint	Stakeholders' Group: Doctors
<i>Business Process</i>	<i>Description</i>
Appointment Services	From the arrival of a patient to hospital to attend a scheduled appointment, check or create new patient EHR including capture of demographic data and general patient information by nurse; EHR management, medical consultation, diagnosis, emission of medical order

Table VI: Business Process Derived from the DDU of Domain Engineers' Viewpoint

Viewpoint	Stakeholders' Group: Domain Engineers
<i>Business Process</i>	<i>Description</i>
EHR Management	Consider in the User Interface (UI) component in Presentation Layer, the access to the EHR Management system in Process Layer; provide EHR access, modification, sharing and registering in database in Data Layer. A Transmission Layers should be present for network services, and it crosscuts all other layers.

Table VII: Business Process Derived from the DD of Directors of Healthcare Governmental Institutions' Viewpoint

Viewpoint	Stakeholders' Group: Directors of Healthcare Governmental Institutions
<i>Business Process</i>	<i>Description</i>
On-line appointment services	Provide precise appointment services with specific specialist; handle requests' volume, provide secure access to appointment services; have wide-range and reliable connection facility; have a friendly user interface

Different notations can be used to specify facets from stakeholders' viewpoints with intrinsic descriptors, each one offering different specification granularity; from each specification, more details are extracted; in this work the following notations will be used:

- informal textual specification by tables (see Tables VIII, X, and XI),
- semi-formal UML [30] diagrams (see Figure 4, 6, and 7) for all facets,
- semi-formal BPMN [14] diagrams for business processes in the Business Process facet (see Figure 5 [39]).

In this context, a semi-formal notation language means that it has well-defined syntax and semantics; however it cannot be verified mathematically. Formal languages, such as VDM, Z, B, and RAISE/RSL, also mentioned in [20], are used to specify high assurance systems, to reduce errors in requirement definitions of safety-critical software systems. The HIS domain is basically constituted by non-safety-critical integrated enterprise systems; hence we chose UML and BPMN standard

notations, widely used by the software community in this domain.

Table VIII: Textual Specification of Doctors Viewpoint for Appointment Services Business Process Facet Specified by Intrinsic

Domain	EHR-HIS – Healthcare institution Requiring the SPL			
Viewpoint	Stakeholders' Group: Doctors			
Appointment Services Business Process Facet				
Entities	Functions	Events	Behavior	Quality
EHR	EHR Access EHR Register	<i>EHR is found or created</i> <i>EHR is requested by doctor; doctor attends patient medically and EHR is modified</i>	Nurse confirms the existence of patient EHR or creates an EHR for new patient; nurse provides EHR to doctor (<i>EHR is provided to doctor</i>) Doctor reviews patient EHR, adds laboratory and/or examinations results, if any, to patient EHR; patient is attended medically; EHR is modified (<i>EHR is reviewed, modified registered</i>)	<i>Security (confidentiality, authenticity, integrity) to access EHR: confidentiality or access policy is different for doctors and nurses, authenticity is needed to identify user, and integrity is required for data consistency.</i> <i>Availability of on-line connection to access EHR, time behavior to quick access to EHR, modifiability to change EHR, availability-persistency to retrieve always EHR and interoperability for EHR sharing.</i>
Diagnosis	Perform diagnosis	<i>Medical hand-outs or catalogues are required on-line; on line consultation with other doctors in different healthcare institutions or locally to perform diagnosis; diagnosis is provided and registered on EHR</i>	Doctor requests on-line hand-outs or catalogues; he can consult on-line with other doctors in different healthcare institutions or local doctors, and patient's EHR must be shared by other doctors; review laboratory and/or examinations results; doctor produces diagnosis(<i>Request of on-line hand-outs</i>)	<i>Time behavior and availability of on-line connection to access quickly hand-outs, catalogues and other doctors</i>
Resources: Medical Materials (*)	Medical materials request	Doctor <i>requires medical material</i> for consultation	Doctor requests medical materials calling authorized staff (or using the EHR-HIS system if this facility is available) (<i>Request of medical material</i>)	<i>Physical availability of materials, time behavior to receive materials from authorized staff (or availability of on-line connection to access the EHR-HIS system for materials request, if this facility is supported)</i>
Resources: Medical Equipment (*)	Medical equipment request	Doctor <i>requires special equipment</i> for diagnosis	Doctor requests medical equipment calling authorized staff (or using the EHR-HIS system if this facility is available) (<i>Request of medical equipment</i>)	<i>Physical availability of equipment, time behavior to receive equipment from authorized staff (or availability of on-line connection to access the HER-HIS system for equipment request if this facility is supported),</i>
Medical Appointment	Emission of new appointment	<i>New appointment is required</i> for patient, if necessary; appointment is scheduled	Doctor registers patient for a new appointment; the appointment is scheduled. (<i>Request/schedule new appoint.</i>)	<i>Availability of on-line connection to access EHR for Medical Appointment</i>
Medical Order	Emission of medical order	<i>Medical order is elaborated</i> for present consultation	Doctor emits medical order (<i>Emit/register medical order</i>)	<i>Availability of on-line connection to access EHR to emit and register Medical Order</i>

(*) These services will not be considered for EHR-HIS in this work; the abridged behavior name is specified within () in italics

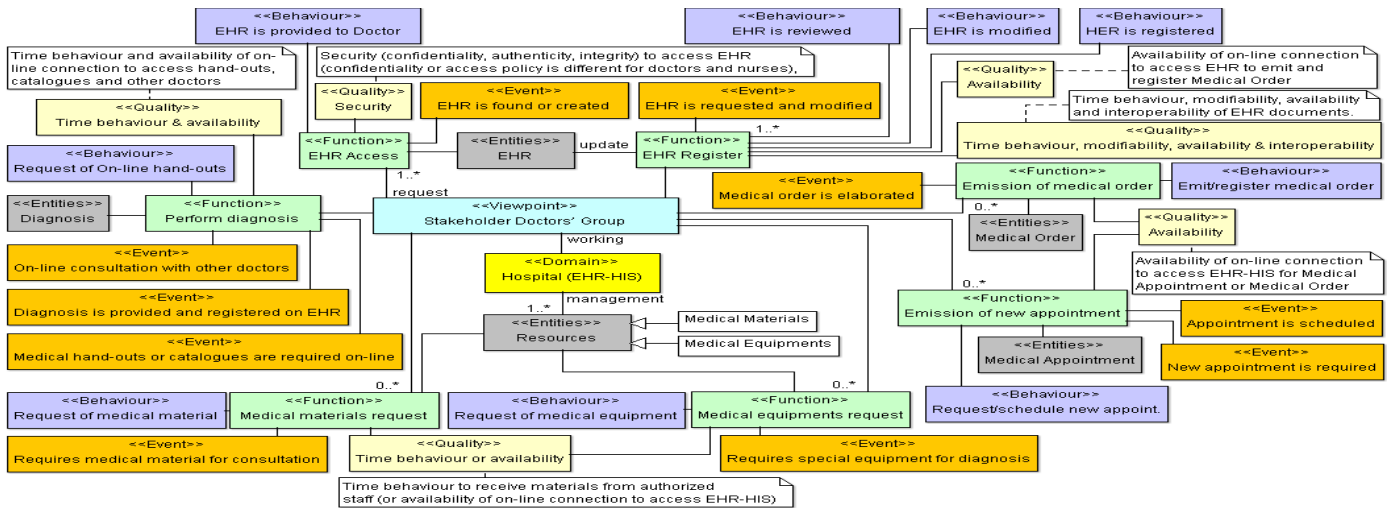


Figure 4: UML Specification for Doctors' Viewpoint of the Appointment Services Business Process Facet

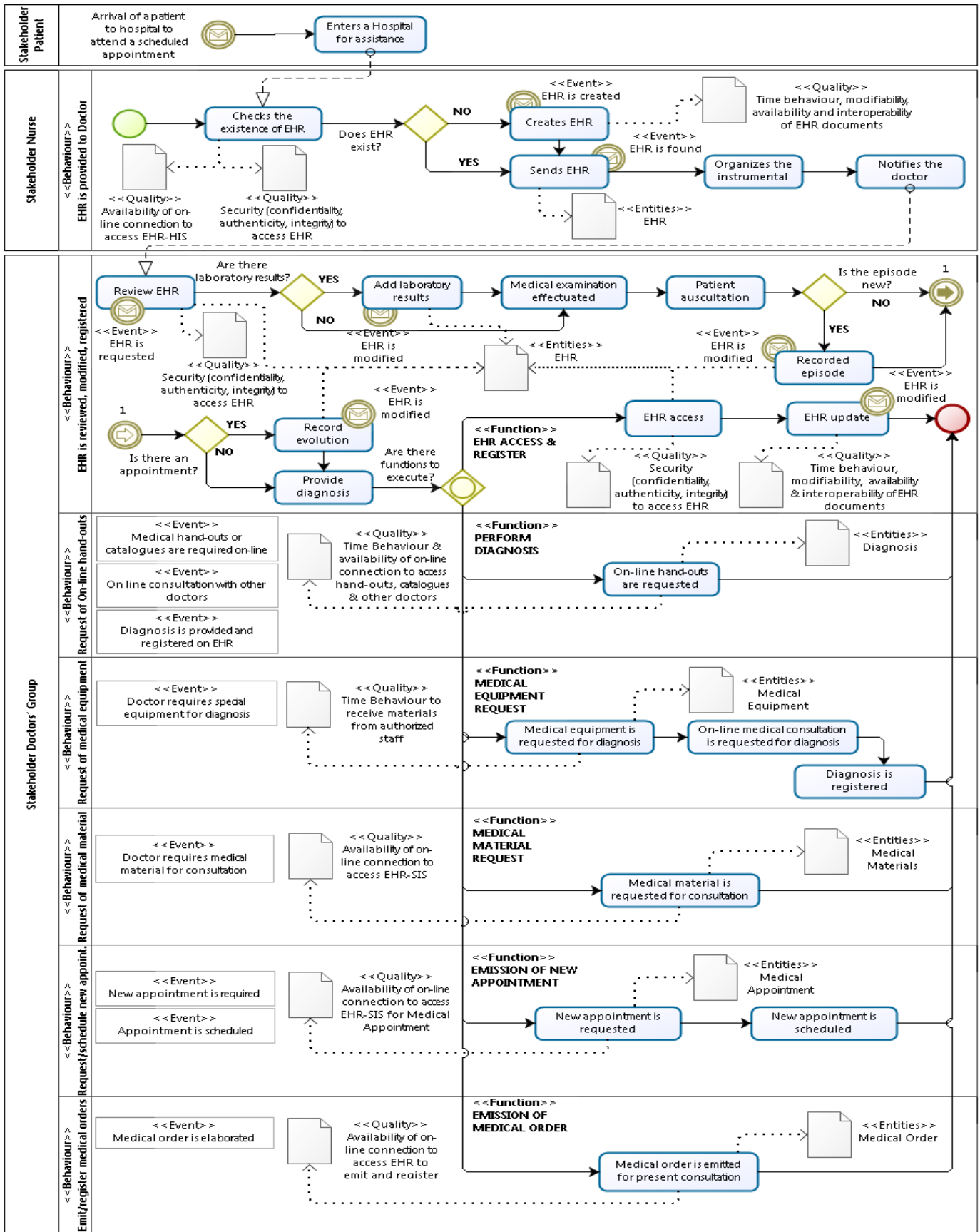
















Figure 5: Appointment Services Process from Doctors' Viewpoint Specified from the Intrinsic Facet, Expressed in BPMN

Table IX: BPMN Notation Used in the Business Process Facet Specification

BPMN Notation	Symbol	Description	Interpretation (for Figure 5)
	Task	Simple or atomic activity representing the work in an organization. They consume resources; it is not detailed further.	They can be mapped into SPL RA architectural components or services
	Exclusive gate	Control element of data workflow. A unique path is selected among several alternatives	It represents the SPL variability.
	Inclusive gate	Control element of the data workflow. One or more path (s) can be selected from several alternatives	It represents the SPL variability.
	Starting event	It initiates a process	It represents the starting or entry point of a business process.
	Starting message event	A process initiates when a message is received	It represents the events arising from an active business process.
	Ending event	It indicates the end of a workflow	It represents the end of some behavior in the business process.
	Terminal ending event	It indicates that a process ends, even if there are active workflows	It represents the end of a business process.
	Intermediate link event	It allows the connection of two process sections.	It allows a connection between two sections of the business process.
	Swimlanes (Pool)	It is a process container; it represents participant, entity or role.	It represents the stakeholders' viewpoints.
	Swimlanes (Lane)	They are pool's subdivisions; it represents participants within an organization.	It represents the different behaviors of the business process.
	Connector object: Sequence	It represents the workflow and the sequence of activities.	It represents the execution flow of the activities in a business process.
	Connector object: Message	It represents interactions among processes or pools.	It represents changes of stakeholders' viewpoints.
	Associations	They are used to relate additional information on the process.	They relate entities, activities or functions with required quality properties.
	Artefact: data object	They provide additional information on the process; they show the information required by an activity, such as input/output.	Document specifying the quality required by entities, activities or functions, in each behavior.

Other business processes can be derived from the analysis of the other facets, such as the on-line appointment services process (see Table VI) from the DDU representing the viewpoint of Directors of healthcare governmental institutions, or the construction of EHR management system (see Table VII) process from the DDU representing the viewpoint of Domain Engineers; however, they will not be considered for this study to abridge the presentation.

Table IX describes the BPMN symbols used in the *Appointment Services Process from Doctors' Viewpoint*, specified from the intrinsics facet. A glimpse on SPL variability can be inferred from the inclusive and exclusive logic gates, since they reflect alternative workflows; they imply a sequence of actions to be performed to achieve a functionality, i.e. functional variability; however, since each activity has associated its quality property, this also can imply non functional variability. This point has to be signalled, because in the domain modelling by the intrinsic descriptors, which was represented in UML (see Figure 4), variability cannot be shown.

In consequence, the use of BPMN is advantageous for our approach, because it contributes to show variability at business process level, which will be mapped later-on into the SPL RA variability model.

4) *Domain Modelling*: The technique used consists in the facet specifications with the intrinsics, by tables and UML diagrams, obtained in step (c) (see Tables VIII, X and XI, and Figures 4, 6, 7). Notice that we have only the Appointment Services Business Process facet, which is specified in BPMN in Figure 5, and it is an example of a Domain Model.

However, to have the complete Domain Model picture, the other business processes derived from DDU in Tables VI and VII, EHR management, from the Domain Engineer viewpoint and On-line appointment services, from the Directors of healthcare governmental institutions viewpoint respectively, are found as behaviours in the corresponding facet specification (see Tables X and XI), and they can be specified by intrinsics as new Business Process facets, as it was done for Appointment Services in Table VIII. From the business process facet analysis, see Table VIII and Figures 4 and 5, and from the BPMN specifications, we obtain information on:

- The clear identification of the involved stakeholders.
 - Possible SPL variants from a particular viewpoint, such as the security quality property (see Figure 5), which can be solved proposing later on different available technological mechanisms or services. Note that quality properties are reflected in all viewpoints specifications due to the intrinsic quality descriptor that has been introduced in the adaptation of the domain modeling approach of [20].
 - Fine-grained functionalities present in the process as “functions” that can be mapped into large-grained architectural components or services with the BPMN “Aggregation” construct.
- Quality that must be satisfied by each functionality present in the process is specified as a comment, since BPMN has no notation for quality properties.
- The entities or objects produced or manipulated from/by functionalities.

Table X: Textual Specification by Intrinsic of Domain Engineers' Viewpoint for the Support Technology Facet

Domain	EHR-HIS – Healthcare institution requiring the SPL			
Viewpoint	Stakeholders' Group: Domain Engineers			
Support Technology Facet				
Entities	Functions	Events	Behaviour	Quality
Presentation Layer: User Interface (UI)	HIS on-line access	HIS is accessed through UI access buttons	Access to HIS main functionalities by authorized persons (EHR access)	Security (confidentiality, authenticity, integrity) to access EHR-HIS (confidentiality or access policy is different for doctors or nurses); availability-persistence of connection to access EHR-HIS; authenticity is required for user identification, and integrity refers to maintain and assure data accuracy and consistency; time behaviour response time to retrieve HER; adaptability-scalability refers to add new medical standards
Process Layer: EHR-HIS functionalities	Patient System EHR Management System Report Systems	Patient appointment services including demographic and general information data collection EHR management: access, registering, modification, sharing Emission of medical reports and billing services	Scheduling services for patient appointments (Appoint. Mang.) EHR access, recording, modification, sharing; provide access to medical catalogues for diagnosis, to on-line consultation for diagnosis, emission of diagnosis, consultation and/or addition to EHR of new laboratory & examination results (EHR management) Medical reports and billing management: edition, access, modification, registering (Reports/Billing Manag.)	Correctness-Precision: for computation algorithm, security, availability-persistence Interoperability, adaptability-scalability, availability-persistence, security Correctness-precision, availability-persistence, security
Data Layer: Data Base	Data Base Management System	Provide all database services	Allows interoperability, adaptability, persistence, integrity for database services (Provide quality DB services)	Interoperability, adaptability-scalability, availability-persistence, security (integrity)
Transmission or communication) Layer: Network	Internet Communication protocols	Provide information exchange	Allows efficient, reliable, secure information exchange; information is exchanged independently from the platform (Provide quality information exchange)	Time behaviour, availability-persistence, adaptability-scalability, security

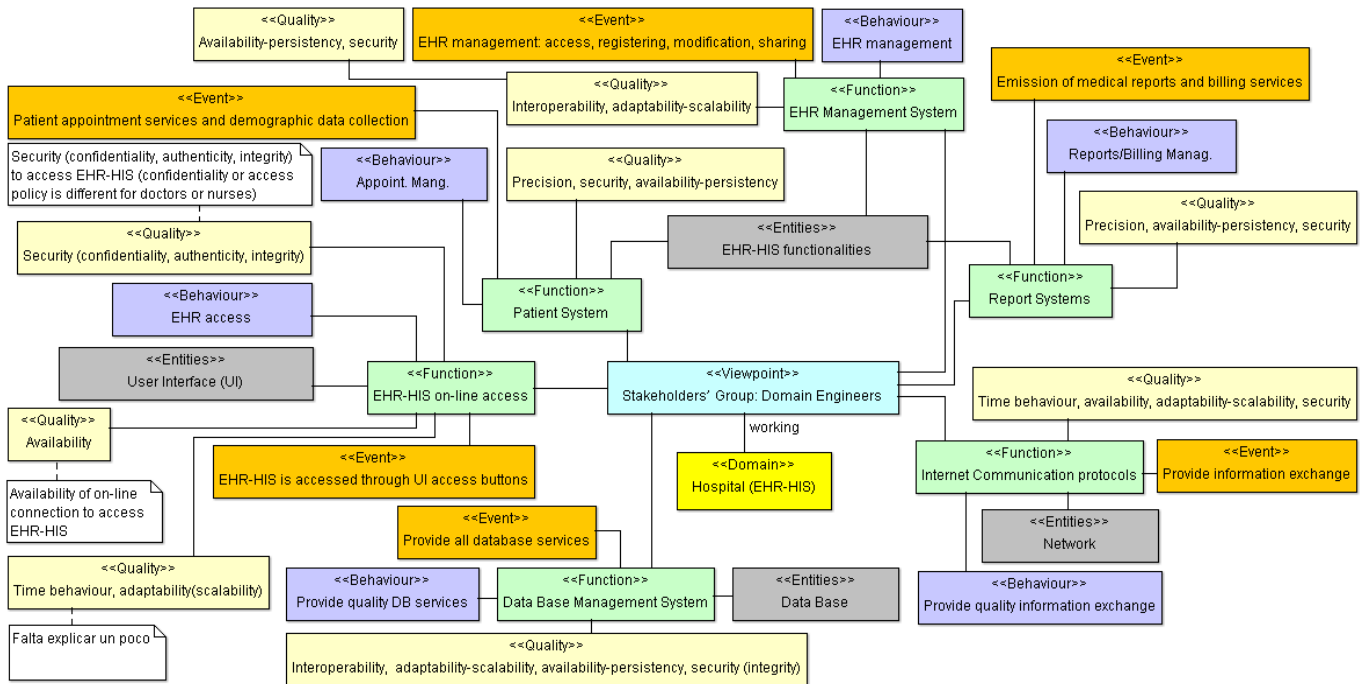


Figure 6: UML Specification by Intrinsic of Domain Engineers' Viewpoint for the Support Technology Facet

Table XI: Textual Specification by Intrinsic of Directors of Healthcare Governmental Institutions' Viewpoint for the Rules&Regulations Facet

Domain	EHR-HIS – Healthcare institution requiring the SPL			
Viewpoint	Stakeholders' Group: Directors of Healthcare Governmental Institutions			
Rules & Regulations Facet				
Entities	Functions	Events	Behaviour	Quality
EHR	EHR sharing	Healthcare records is digitalized using some standard format	Healthcare records are shared (EHR sharing)	Interoperability
Database of specialists	Data Base Management System	Provide all data base services; Retrieve, record, modify specialist data	Allows interoperability, adaptability, persistency, integrity for database services (Provide database services)	Interoperability, adaptability-scalability, availability-persistency, security (integrity)
Appointment	Appointment scheduling service	Appointment is assigned or kept in a waiting list; appointment is reported to patient; priority is managed	Provide precise appointment services with specialist; handle volume of requests, provide secure access to appointment services; have available connection facility; have a friendly user interface (Provide on-line appoint. services)	Adaptability-scalability, availability-persistency, security, usability, precision

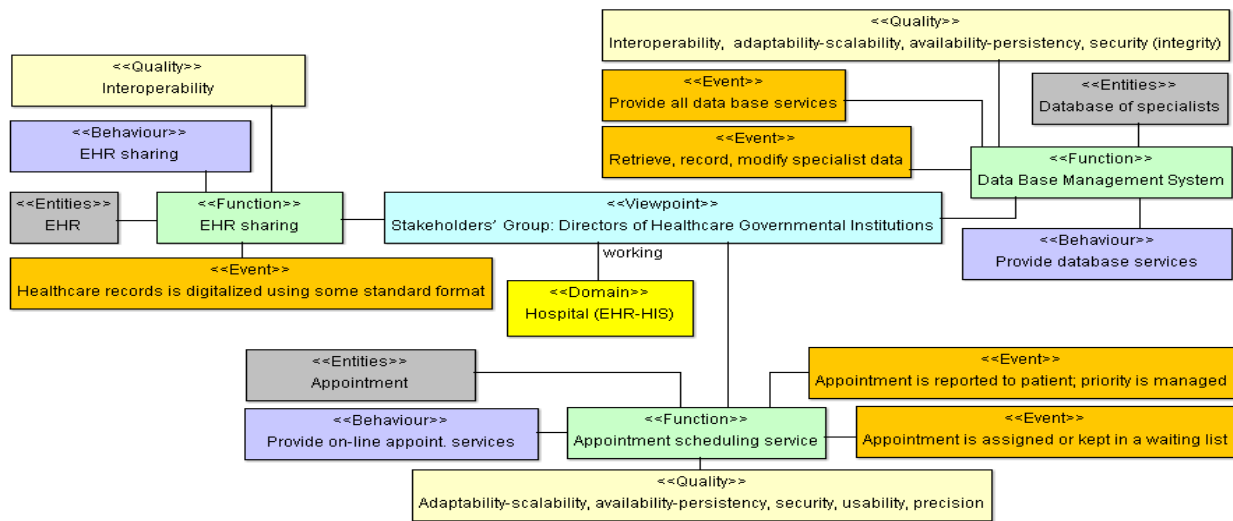


Figure 7: UML Specification by Intrinsic of Directors of Healthcare Governmental Institutions' Viewpoint for the Rules&Regulations Facet

VI. CONCLUSION

It is known that general frameworks explain the “What to do” about things, but not the “How to do” to make things work. We presents the Domain Scoping sub-phase of the PL Scoping phase within the DE lifecycle, following the SPL Reference Model ISO/IEC 26550. The “How to do” is taken from Bjørner’s domain modelling, which involves the facet notion and the stakeholders’viewpoint, focusing on business process modelling. Our main contribution is the specification of the Domain Scoping step as a systematic and repeatable process, centred on the early specification of quality properties as descriptors involved in all facets, considering their clear traceability, reflected into the BPMN representation; this issue will facilitate later on the reference architecture evolution. Notice also that the derivation of the business process specification in BPMN can be automatized from the UML representation of the facets; it is also a widely known and used notation, to bridge the gap between business processes and their implementation, for example as Web services [14]. It is claimed that the effort spent in PL Scoping will reduce the huge effort required in DRE and DD phases [21]. The Domain

Scoping step of our PLScoP is crucial to construct a detailed Domain Model that will help to delimit clearly the SPL scope and functional and non functional granularity, to reduce the effort in the subsequent DRE and DD phases, where the RA is built. Our Domain Scoping process is applied to a complete case study in the Healthcare Information System domain to illustrate our approach. A more complete specification of PLScoP, and subsequent DRE and DD phases of the DE lifecycle to construct the SPL RA, are on-going works.

ACKNOWLEDGMENT

We wish to thank the referees for their useful and pertinent comments. Financial support to this research has been provided by the DARGRAF No. 03-8730-2013-2 project of the CDCH (Consejo de Desarrollo Científico y Humanístico), and from the Postgraduate Studies in Computer Science, Faculty of Science, Universidad Central de Venezuela.

REFERENCES

- [1] I. Reinhartz-Berger, A. Sturm, T. Clark, S. Cohen, and J. Bettin, (Editors), *Domain Engineering. Product Lines, Languages and Conceptual Models*, Springer, 2013.

- [2] E. Berard, *Essays in Object-Oriented Software Engineering*, New York: Prentice Hall, 1992.
- [3] A. Helfferich, G. Herzwurm, and S. Jesse, *Software Product Lines and Service-Oriented Architecture: A Systematic Comparison of Two Concepts*, in proceedings of the First Workshop on Service-Oriented Architectures and Software Product Lines (SOAPL 2007), Kyoto, Japan, May 2007.
- [4] P. Istoan, G. Nain, G. Perrouin, and J. Jezequel, *Dynamic Software Product Lines for Service-Based Systems*, in proceedings of the 9th IEEE International Conference on Computer and Information Technology (CIT'09), INRIA, France, October 2009.
- [5] P. Istoan, *Software Product Lines for Creating Service Oriented Applications*, Masters Internship at IRISA Rennes research institute, TRISKELL research team, Rennes, June 2009.
- [6] F. Medeiros, E. de Almeida, and S. de Lemos, *Designing a Set of Service-oriented Systems as a Software Product Line*, in proceedings of the Software Components, Architectures and Reuse (SBCARS 2010), Fourth Brazilian Symposium, IEEE, Salvador, Bahia, Brazil, September 2010.
- [7] A. Korff, *Implementing ISO 26550 Model-based*, in proceedings of the 6th edition of the International Academic-Industrial Conference in Complex Systems Design & Management (CSD&M 2015), Paris, France, November 2015.
- [8] T. Kähkölä, *Standards Initiatives for Software Product Line Engineering and Management within the International Organization for Standardization*, in proceedings of the 2010 43rd Hawaii International Conference on System Sciences (HICSS 2010), Koloa, Hawaii, USA, January 2010.
- [9] K. Pohl, G. Bockle, and F. Van Der Linden, *Software Product Lines Engineering: Foundations, Principles, and Techniques*, Springer, 2005.
- [10] Q. Munir and M. Shahid, *Software Product Line: Survey of Tools*, Master's thesis, Linköping University, Department of Computer and Information Science, 2010.
- [11] ISO/IEC NP 26550, *Software and Systems Engineering – Reference Model for Software and Systems Product Lines*, ISO/IEC JTC1/SC7 WG4, 2013.
- [12] F. Losavio, O. Ordaz, and V. Esteller, *Quality-based Bottom-up Design of Reference Architecture Applied to Healthcare Integrated Information Systems*, in proceedings of the 2015 IEEE 9th International Conference in Research Challenges in Information Science (RCIS 2015), Athens, Greece, May 2015.
- [13] ISO/IEC 25010, *Systems and Software Engineering -- Systems and Software Quality Requirements and Evaluation (SQuARE) -- System and Software Quality Models*, ISO/IEC JTC1/SC7/WG6, 2011.
- [14] Object Management Group (OMG), *Business Process Model and Notation (BPMN)*, version 2.0, available in: <http://www.omg.org/spec/BPMN/2.0/>, 2011.
- [15] K. Lee, K. Kang, and J. Lee, *Concepts and Guidelines of Feature Modeling for Product Line Software Engineering*, in proceedings of the 7th International Conference on Software Reuse (ICSR-7): Methods, Techniques, and Tools, Austin, TX, USA, April 2002.
- [16] S. Apel, D. Batory, C. Kästner, and G. Saake, *Feature-Oriented Software Product Lines*, Springer, 2013.
- [17] N. Siegmund, M. Rosenmuller, M. Kuhleemann, C. Kastner, S. Apel, and G. Saake, *SPL Conqueror: Towards Optimization of Non-functional Properties in Software Product Line*, *Software Quality Journal*, vol. 20, no. 3-4, pp. 487-517, September 2012.
- [18] B. Kitchenham, *Guidelines for Performing Systematic Literature Reviews in Software Engineering*, Version 2.3, EBSE Technical Report, Software Engineering Group, School of Computer Science and Mathematics, Keele University, UK and Department of Computer Science, University of Durham, UK, 2007.
- [19] M. de Moraes, E. de Almeida, and S. Romero, *Systematic Review on Software Product Lines Scoping*, in proceedings of 6th Experimental Software Engineering Latin American Workshop (ESELAW 2009), São Carlos, Brasil, November 2009.
- [20] D. Björner, *Software Engineering 3 Domains, Requirements, and Software Design*, Texts in Theoretical Computer Science, EATCS Series, Editors: W. Brauer G. Rozenberg A. Salomaa, Springer-Verlag Berlin Heidelberg, 2006.
- [21] I. Da Silva, P. Neto, P. O'Leary, E. De Almeida, and S. de Lemos Meira, *Software Product Line Scoping and Requirements Engineering in a Small and Medium-sized Enterprise: An Industrial Case Study*, *Journal of Systems and Software*, vol. 88, pp. 189-206, 2014.
- [22] L. Northrop and P. Clements, *A Framework for Software Product Line Practice, Version 5.0*, Product Line Practice Initiative, SEI (Software Engineering Institute), 2012.
- [23] J. Herrera, F. Losavio, and A. Matteo, *RDS de Enfoques y Técnicas para la Construcción de Arquitecturas en un Contexto de Líneas de Productos de Software*, *Revista Venezolana de Computación (ReVeCom)*, vol. 1, no. 1, pp. 17-25, Junio 2014.
- [24] J. Bayer, O. Flege, P. Knauber, R. Laqua, D. Muthig, K. Schmid, T. Widen, and J. M. DeBaud, *PuLSE: a Methodology to Develop Software Product Lines*, in proceedings of the 1999 Symposium on Software Reusability (SSR'99), ACM, Los Angeles, USA, May 1999.
- [25] F. Losavio, O. Ordaz, and I. Santos, *Proceso de Análisis del Dominio Ágil de Sistemas Integrados de Salud en un Contexto Venezolano*, *Revista Venezolana de Información, Tecnología y Conocimiento, ENL@CE*, vol. 12, no. 1, pp.101-134, Enero-Abril 2015.
- [26] J. C. Herrera, F. Losavio, and O. Ordaz, *Ingeniería del Dominio con el Estándar ISO/IEC 26550 para LPS Considerando la Faceta Calidad*, in proceedings of the Conferencia Nacional de Computación, Informática y Sistemas (CoNCISA 2015), Valencia, Venezuela, Octubre 2015.
- [27] J. M. DeBaud, *A Systematic Approach to Derive the Scope of Software Product Lines*, in proceedings of the 21st International Conference on Software Engineering (ICSE'99), Los Angeles, CA, USA, May 1999.
- [28] K. Schmid, *Customizing the PuLSE Product Line Approach to the Demands of an Organization*, in proceedings of the 7th European Workshop on Software Process Technology (EWSPT'2000), Kaprun, Austria, February 2000.
- [29] P. Kruchten, *Architectural Blueprints—The “4+1” View Model of Software Architecture*, in proceedings of the Conference on TRI-Ada'95, Anaheim, CA, USA, November 1995.
- [30] Object Management Group (OMG), *Unified Modelling Language Superstructure*, version 2.0 (formal/05-07-04), <http://www.omg.org/spec/UML/2.0>, August 2005.
- [31] M. Shaw and D. Garlan, *Software Architecture. Perspectives of an Emerging Discipline*, Prentice-Hall, 1996.
- [32] Institute of Medicine (US), *Key Capabilities of an Electronic Health Record System: Letter report*, Committee on Data Standards for Patient Safety, National Academies Press, 2003.
- [33] HIQA (Health Information and Quality Authority), *Overview of Healthcare Interoperability Standards*, 2013.
- [34] S. Samilovich, *OpenEMR – Historia Clínica Electrónica de Código Abierto y Distribución Gratuita, Apta para su Uso en el Sistema de Salud Argentina*, JAIHO CASI, 2010.
- [35] M. Allauddin, F. Azam, and M. Zia, *A Survey of Quality Assurance Frameworks for Service Oriented Systems*, *International Journal of Advancements in Technology*, vol. 2, no. 2, pp. 188-198, 2011.
- [36] W. Suryn, *Software Quality Engineering: a Practitioner's Approach*, IEEE Wiley & Sons, Inc., Hoboken, New Jersey, 2014.
- [37] H. González, *Integration of Quality Attributes in Software Product Line Development*, Master Thesis en Ingeniería del Software, Métodos Formales y Sistemas de Información (ISMFSI), 2012.
- [38] S. Wagner, *Software Product Quality Control*, Springer, 2013.
- [39] K. I. Farroñay and A. J. Trujillo, *Sistema de Registro de Atención Médica para un Centro de Salud de Nivel I-3 de Complejidad*, Doctoral dissertation, Universidad Peruana de Ciencias Aplicadas (UPC), 2013.