SILVANA PEREIRA DETRO

A Framework for Managing Process Variability Through Process Mining and Semantic Reasoning: An Application in Healthcare

DOCTOR OF THE
PONTIFICAL CATHOLIC UNIVERSITY OF PARANA
IN INDUSTRIAL AND SYSTEMS ENGINEERING

DOCTEUR DE L'UNIVERSITE DE LORRAINE
EN AUTOMATIQUE, TRAITEMENT DU SIGNAL ET DES IMAGES, GENIE INFORMATIQUE

CURITIBA
2017

Ecole Doctorale IAEM Lorraine
PPGEPS

THÈSE

Présentée pour l'obtention du titre de

Docteur de l'Université de Lorraine

en Automatique, traitement du signal et des images, génie informatique

et du titre de

Doutor da Pontifícia Universidade Católica do Paraná

en Engenharia de Produção e Sistemas

par

Silvana PEREIRA DETRO

Un cadre de configuration des variantes de processus à travers la fouille de processus et le raisonnement sémantique : une application dans le cadre de la santé.

A framework for managing process variability through process mining and semantic reasoning: an application in healthcare

Soutenue publiquement le 15/12/2017 devant le jury composé de :

Président :	Jean Marcelo Simao	Professeur à UTFPR, Brésil
Rapporteurs :	Silvana Quaglini	Professeur à l' Università degli Studi di Pavia, Italy
	Henderik Proper	Professeur a Radboud University Nijmegen, Netherland
Examinateurs :	Eduardo Portela	Professeur a PUC-PR, Bresil (Directeur)
	Hervé Panetto	Professeur a l'Universite de Lorraine (Directeur)
	Mario Lezoche	MCF a l'Universite de Lorraine (Co-Directeur)
	Eduardo Rocha Loures	Professeur a PUC-PR, Bresil (Co-Directeur)

Industrial and Systems Engineering Graduate Program – PPGEPS/PUCPR

Centre de Recherche en Automatique de Nancy, UMR 7039, Université de Lorraine - CNRS

Dados da Catalogação na Publicação
Pontifícia Universidade Católica do Paraná
Sistema Integrado de Bibliotecas – SIBI/PUCPR
Biblioteca Central

D483a
2018

Detro, Silvana Pereira
A framework for managing process variability through process mining and semantic reasoning: an application in healthcare / Silvana Pereira Detro ; orientador: Hervé Panetto. – 2018.
160 f. : il. ; 30 cm

Tese (doutorado) – Pontifícia Universidade Católica do Paraná, Curitiba, 2018
Bibliografia: f. 139-160

1. Engenharia industrial. 2. Controle de processo – métodos estatísticos. 3 Mineração de dados. 4. Negócios – Processamento de dados. I. Panetto, Hervé. II. Pontifícia Universidade Católica do Paraná. Programa de Pós-Graduação em Engenharia Industrial e de Sistemas. III. Título

CDD 22. ed. – 670.42

Pontifical Catholic University of Paraná – PUCPR
Polytechnic School
Industrial and Systems Engineering Graduate Program - PPGEPS

APPROVAL TERM

Silvana Pereira Detro

A FRAMEWORK FOR MANAGING PROCESS VARIABILITY THROUGH PROCESS MINING AND SEMANTIC REASONING: AN APPLICATION IN HEALTHCARE

Thesis submitted in partial fulfilment of the requirements for the degree of Doctor of Industrial and Systems Engineering in the Industrial and Systems Engineering Graduate Program of the Polytechnic School, Pontifical Catholic University of Parana.

Prof. Dr. Eduardo Alves Portela Santos
Supervisor – PPGEPS/PUCPR

Prof. Dr. Hervé Panetto
(Supervisor – CRAN/UL)

Prof. Dr. Eduardo de Freitas Rocha Loures
(Co-Supervisor – PPGEPS/PUCPR)

Prof. Dr. Mario Lezoche
(Co-Supervisor – CRAN/UL)

Prof. Dr. Henderik Alex Proper
(External Member – Radboud University Nijmegen)

Prof. Dr. Silvana Quaglini
(External Member – University of Pavia)

Prof. Dr. Jean Marcelo Simão
(External Member – Federal University of Technology - Paraná)

Curitiba, 15th December of 2017.

Address: Rua Imaculada Conceição, 1155 - Prado Velho – CEP: 80215-901 - Curitiba - Paraná - Brasil
Tel: +55 41 3271-2579 www.pucpr.br/ppgeps

ACKNOWLEDGEMENTS

Many people have contributed for the development of this work. First and foremost, my deepest thankful to my supervisors Eduardo Alves Portela and Hervé Panetto and my co-supervisors Eduardo de Freitas Rocha Loures and Mario Lezoche. Each one supported me in different but complementary ways. Professor Eduardo Portela guided me in every detail of this work. He always encouraged me and gave me advises, not just about my work but also for my life. Professor Hervé Panetto always gave me good advices and suggestions to improve my work. He supported me during the development of the ontologies, and taught me to pay attention to the details.

Professor Eduardo Loures helped me to understand how the different fields could be connected. He encouraged me in difficult moments by sharing his own experience. Professor Mario Lezoche help me to understand the use of ontologies and the semantic reasoning, which is an important part of the work and it was completely new for me. He also gave a wonderful reception in Nancy and supported me during my stay there.

I would also like to extend my appreciation to Prof. Silvana Quaglini and Prof. Henderik A. Proper who agreed to be the reviewers of this thesis, and Prof. Jean Marcelo Simão for agreed to be the examiner of this work.

I would like to thank to CAPES (Coordenação de Aperfeiçoamento de Pessoal de Nível Superior) and to PUCPR (Pontifícia Universidade Católica do Paraná) for their financial support during the development of this work. I would like to express my gratitude to CRAN laboratory for providing me a great research environment during the time that I studied at France. I would also like to extend my gratitude for all the support and assistance provided by PPGEPS.

My sincere appreciation to Professor Osiris Canciglieri Junior that always gave me important advises about different issues. I would also like to extend my appreciation to Dr Anderson Szejka for all the support provided during the development of this work. Thank you to Dr. Yongxin Liao that gave me important advices at the beginning of the development of the ontologies. I also would like to thank Dr. Alexis Aubry that gave me suggestions to improve my work. I wish to thank to Dr Claudia Moro and Lilian Mie Mukai that helped me to understand all the proceedings performed during the treatment provided for patients diagnosed with acute ischemic stroke.

My big thank you goes to my friends in France, as well those at home in Brazil. Nobody has been more important to me in the pursuit of this project than the members of my family. I would like to thank my mother, Fani, whose love and guidance are with me in whatever I pursue. I wish to thank my sisters, Gilmara and Silmara, my brothers-in-law, Fernando and Valdeci, my nephews, Lucas, Leonardo, Eduardo, Arthur and Bernardo, and my niece Isabella for their love.

Finally, thank God for giving me the strength, knowledge, ability and opportunity to undertake this research study.

“Imagination is more important than knowledge. For knowledge is limited, whereas imagination embraces the entire world.”

(Albert Einstein, 1929)

ABSTRACT

The efficiency of organizations relies on its ability to adapt their business processes according to changes that may occur in the dynamic environment in which they operate. These adaptations result in new versions of the process model, known as process variants. Thus, several process variants can exist, which aim to represent all the related contexts that may differ in activities, resources, control flow, and data. Thus, has emerged the concept of customizable process model. It aims to adapt the process model according to changes in the business context. A process model can be customized by representing the process family in one single model enabling to derive a process variant through transformations in this single model. As benefits, this approach enables to avoid redundancies, promotes the model reuse and comparison, among others. However, the process variant customization is not a trivial-task. It must be ensured that the variant is correct in a structural and behavioural way (e.g. avoiding disconnected activities or deadlocks), and respecting all the requirements of the application context. Besides, the resulting process variant must respect all requirements related to the application context, internal and external regulations, among others. In addition, recommendations and guidance should be provided during the process customization. Guidance help the user to customize correct process variants, i.e., without behavioural problems. Recommendations about the process context help the user in customizing process variants according specific requirements. Recommendations about the business context refers to providing information about the best practices that can improve the quality of the process. In this context, this research aims to propose a framework for customizing process variants according to the user's requirements. The customization is achieved by reasoning on ontologies based on the rules for selecting a process variant and in the internal/external regulations and expert knowledge. The framework is composed by three steps. The first step proposes to identify the process variants from an event log through process mining techniques, which enable to discover the variation points, i.e., the parts of the model that are subject to variation, the alternatives for the variation points and the rules to select the alternatives. By identifying the process variants and their characteristics from an event log, the process model can be correctly individualized by meeting the requirements of the context of application. Based on these aspects, the second step can be developed. This step refers to the development of the questionnaire-model approach. In the questionnaire approach each variation point is related to a question, and the alternatives for each question corresponds to the selection of the process variants. The third step corresponds to apply two ontologies for process model customization. One ontology formalizes the knowledge related with the internal and/or external regulations and expert knowledge. The other refers to the variation points, the alternatives for them and the rules for choosing each path. The ontologies then are merged into one new ontology, which contain the necessary knowledge for customize the process variants. Thus, by answering the questionnaire and by reasoning on the ontology, the alternatives related with the business process and the recommendations about the business context are provided for the user. The framework is evaluated through a case study related to the treatment of patients diagnosed with acute ischemic stroke. As result, the proposed framework provides a support decision-making during the process model customization.

Keywords: Process model customization; Configuration; Process mining, Ontologies, Semantic reasoning.

RÉSUMÉ

Les organisations doivent relever le défi d'adapter leurs processus aux changements qui peuvent survenir dans l'environnement dynamique dans lequel elles opèrent. Les adaptations dans le processus aboutissent à plusieurs variantes de processus, c'est-à-dire dans différentes versions du modèle de processus. Les variantes de processus peuvent différer en termes d'activités, de ressources, de flux de contrôle et de données. Ainsi, le concept d'un modèle de processus personnalisable est apparu et il vise à adapter le modèle de processus en fonction des exigences d'un contexte spécifique. Un modèle de processus personnalisable peut représenter toutes les variantes de processus dans un modèle unique dans lequel les parties communes ne sont représentées qu'une seule fois et les spécificités de chaque variante sont préservées. Alors, grâce à des transformations dans le modèle de processus générique, une variante de processus peut en être dérivée. En tant qu'avantages, cette approche permet d'éliminer les redondances, favorise la réutilisation, entre autres. Cependant, la personnalisation des modèles de processus n'est pas une tâche triviale. La personnalisation doit assurer que la variante obtenue est correcte du point de vue structurel et comportemental, c'est-à-dire la variante obtenue ne doit pas présenter d'activités déconnectées, d'interblocages actifs ou d'interblocages, entre autres. En outre, la variante de processus doit satisfaire à toutes les exigences du contexte de l'application, aux réglementations internes et externes, entre autres. De plus, il est nécessaire de fournir à l'utilisateur des directives et des recommandations lors de la personnalisation du processus. Les directives permettent la personnalisation correcte des variantes de processus, en évitant les problèmes de comportement. Les recommandations concernant le contexte de l'entreprise rendent possible l'amélioration du processus et aussi la personnalisation des variantes en fonction des besoins spécifiques. Dans ce contexte, cette recherche propose un cadre pour la personnalisation des variantes de processus en fonction des besoins de l'utilisateur. La personnalisation est réalisée grâce à l'utilisation d'ontologies pour la sélection des variantes. Le cadre est composé de trois étapes. La première correspond à l'identification des variantes à partir d'un journal d'événements au moyen de techniques d'exploration de processus, qui permettent de découvrir des points de variation, c'est-à-dire les parties du processus sujettes à variation, les alternatives disponibles pour chaque point de variation et les règles de sélection des alternatives disponibles. L'identification des variantes de processus et de leurs caractéristiques à partir d'un journal des événements permet de personnaliser un modèle de processus en fonction du contexte de l'application. À partir de ces aspects, la deuxième étape peut être développée. Cette étape concerne le développement d'un questionnaire, dans lequel chaque question est liée à un point de variation et chaque réponse correspond à la sélection d'une variante. Dans la troisième étape, deux ontologies sont proposées. La première formalise les connaissances liées aux réglementations externes et internes et aux connaissances des spécialistes. La deuxième ontologie se réfère aux points de variation, aux alternatives existantes pour chaque point de variation et aux règles liées à la sélection de chaque alternative. Ensuite, ces ontologies sont intégrées dans une nouvelle ontologie, qui contient les connaissances nécessaires pour personnaliser la variante de processus. Ainsi, à travers le questionnaire et le raisonnement sémantique, la variante est sélectionnée et les recommandations concernant le processus d'affaires sont fournies en fonction de la sélection de l'utilisateur lors de la personnalisation du processus. Le cadre proposé est évalué au moyen d'une étude de cas liée au traitement des patients chez qui un AVC ischémique aigu a été diagnostiqué. Les recommandations obtenues grâce à l'approche développée fournissent un support pour la prise de décision lors de la personnalisation du modèle de processus.

Mots-clés : modèle de processus personnalisable, exploration de processus, ontologie, raisonnement sémantique.

RESUMO

Organizações enfrentam o desafio de adaptar seus processos de acordo com mudanças que podem ocorrer no ambiente dinâmico em que operam. Adaptações no processo resultam em diversas variantes de processo, isto é, em diferentes versões do modelo de processo. As variantes de processo podem diferir em atividades, recursos, fluxo de controle e dados. Assim, surgiu o conceito de modelo de processo customizável, que tem como objetivo adaptar o modelo de processo de acordo com os requisitos de um contexto especifico. Um modelo de processo customizável pode representar todas as variantes de processos em um único modelo, no qual as partes comuns são representadas apenas uma vez e as especificidades de cada variante é preservada. Assim, por meio de transformações no modelo de processo genérico uma variante de processo pode ser derivada. Como benefícios, esta abordagem possibilita eliminar redundâncias, promove o reuso, entre outros. No entanto, a customização de modelos de processo não é uma tarefa trivial. A customização deve garantir que a variante obtida seja correta tanto do ponto de vista estrutural quanto comportamental, ou seja, a variante obtida não deve apresentar atividades desconectadas, livelocks ou deadlocks, entre outros. Além disso, a variante de processo deve respeitar todos os requisitos do contexto de aplicação, regulações internas e externas, entre outros. Em adição, é necessário fornecer ao usuário orientações e recomendações durante a customização do processo. Orientações permitem a correta customização de variantes de processo, evitando problemas comportamentais. Recomendações a respeito do contexto do negócio possibilitam a melhoria do processo e também a customização de variantes de acordo com requisitos específicos. Neste contexto, esta pesquisa propõe um framework para a customização de variantes de processo de acordo com os requisitos do usuário. A customização é realizada através do uso de ontologias para a seleção de variantes. O framework é composto de três passos. O primeiro corresponde a identificação das variantes a partir de um registro de eventos por meio de técnicas de mineração de processos, as quais possibilitam a descoberta dos pontos de variação, isto é, as partes do processo que estão sujeitos à variação, as alternativas disponíveis para cada ponto de variação e as regras para a seleção das alternativas disponíveis. A identificação das variantes de processo e suas características com base em um log de eventos, permite customizar um modelo de processo de acordo com o contexto de aplicação. Baseado nestes aspectos, o segundo passo pode ser desenvolvido. Este passo refere-se ao desenvolvimento de um questionário, no qual cada pergunta esta relacionada a um ponto de variação e cada resposta corresponde a seleção de uma variante. No terceiro passo, duas ontologias são propostas. A primeira formaliza o conhecimento relacionado às regulações externas e internas e o conhecimento de especialistas. A segunda ontologia refere-se aos pontos de variação, às alternativas existentes para cada ponto de variação e às regras relacionadas a seleção de cada alternativa. Em seguida, estas ontologias são integradas em uma nova ontologia, a qual contém o conhecimento necessário para customizar a variante de processo. Desta forma, por meio do questionário e do raciocínio semântico, a variante é selecionada e as recomendações a respeito do processo de negócio são fornecidas de acordo com a seleção do usuário durante a customização do processo. O framework proposto é avaliado através de um estudo de caso relacionado ao tratamento de pacientes diagnosticados com acidente vascular cerebral isquêmico agudo. As recomendações obtidas por meio da abordagem desenvolvida fornecem um suporte a tomada de decisão durante a customização do modelo de processo.

Palavras-chave: modelo de processo customizável, mineração de processos, ontologia, raciocínio semântico.

LIST OF ILLUSTRATION

Figure 1 - Main stages of the design science research 20
Figure 2 - Outline of the Thesis 22
Figure 3 - Process model customization through process mining and ontologies 25
Figure 4 - A taxonomy of process flexibility needs 27
Figure 5 - Process variants and process family 28
Figure 6 - Approaches to define a customizable process model 31
Figure 7 - Corrected and incorrected customized process model 32
Figure 8 - Types of process mining: discovery, conformance and enhancement 35
Figure 9 - The stack of ontology mark-up languages 44
Figure 10 - Process variants for handling medical examinations 52
Figure 11 - Configuration alternatives of a Configurable OR connector 54
Figure 12 - EPC model for the illustrative process for handling medical examination 56
Figure 13 - Ports in a C-YAWL 57
Figure 14 - Configurable YAWL for the illustrative process for handling medical examination 59
Figure 15 - C-iEPC model for handling medical examinations 60
Figure 16 - ADOM for handling medical examination 64
Figure 17 – Configurative Process Modelling approach for handling medical examinations . 66
Figure 18 - PESOA approach for handling medical examinations 68
Figure 19 - Provop approach for handling medical examinations 69
Figure 20 - Questionnaire-model related with the process for handling medical examination 71
Figure 21 - Example of the feature diagram 74
Figure 22 - Framework for customize process variants 81
Figure 23 - Generating logs using CPN Tools 84
Figure 24 - Example of a Coloured Petri Net 85
Figure 25 - Declarations for the CPN model 86
Figure 26 - Decision mining analysis 89
Figure 27 - Example of a decision tree 91
Figure 28 - Decision mining analysis 92
Figure 29 - Example of the rules in the process model 92
Figure 30 - Example of invisible activities 93
Figure 31 - Example of duplicate activities 94

Figure 32 - Example of loops in a process model 94
Figure 33 - Example of a mandatory and an optional variation point........ 95
Figure 34 - Development of the questionnaire based on the decision tree 97
Figure 35 - Ontology based in the variation points........ 98
Figure 36 - SWRL rules 100
Figure 37 - Ontology based on the regulations 100
Figure 38 - Concepts in the merged ontology........ 101
Figure 39 - Merged ontology 102
Figure 40 - Result of the reasoning step........ 103
Figure 41 - Information obtained in the event log related with the treatment of patients diagnosed with acute ischemic stroke 106
Figure 42 - Information for developing the process model........ 109
Figure 43 - Process model developed based on the event log, expert knowledge and the clinical guideline 111
Figure 44 - Excerpt of the process model developed based on the event log, expert knowledge, and clinical guideline........ 112
Figure 45 - Process model obtained by applying the heuristic miner algorithm........ 113
Figure 46 - Decision tree obtained by applying the decision mining analysis........ 114
Figure 47 - Relation between variation points 116
Figure 48 - Concepts of the ontology of the variation points........ 117
Figure 49 - SWRL rules related with the variation points 118
Figure 50 - Variation points selection after the reasoning step........ 118
Figure 51 - Recommendations obtained after the reasoning step 120
Figure 52 - Concepts in the merged ontology........ 121
Figure 53 - Adjusts performed in the merged ontology........ 122
Figure 54 - SWRL rule of the ontology related to the variation points 122
Figure 55 - SWRL rule of the ontology related to the clinical guideline about the ischemic stroke........ 123
Figure 56 - SWRL rules for selecting the variation point related to haemorrhagic symptoms 124
Figure 57 - Results obtained after the reasoning step 124
Figure 58 - Questions related to the patient’s symptoms........ 125
Figure 59 - Results of the reasoning step for customizing a process model........ 126
Figure 60 - Fragment of the process variant customized through the merged ontology........ 127

LIST OF TABLE

Table 1 - Classification of ontologies 43
Table 2 - Constraints for the configuration of connectors 53
Table 3 - Link between domain facts and configurable activities.. 73
Table 4 - Evaluation results.. 77
Table 5 - Relation between a decision tree and an ontology.. 98
Table 6 - Variation points, alternatives, and the rules.. 114

CONTENTS

1. INTRODUCTION 14
1.1. RESEARCH GOAL 17
1.2. RESEARCH QUESTIONS 17
1.3. METHODOLOGY 19
1.4. CONTRIBUTIONS 21
1.5. OUTLINE OF THE THESIS 21
2. BACKGROUND 24
2.1. BUSINESS PROCESS MODEL AND FLEXIBILITY 26
2.2. BUSINESS PROCESS VARIABILITY 28
2.2.1. Managing business process variability 29
2.3. BUSINESS PROCESS MINING 34
2.3.1. Types of process mining 34
2.3.2. Process Mining algorithms 36
2.3.3. Challenges for process mining 38
2.4. ONTOLOGIES 40
2.4.1. Elements of an Ontology 41
2.4.2. Classification of ontologies 41
2.4.3. Ontology Languages 44
2.4.4. Techniques for reusing ontologies 46
2.5. SYNTHESIS 47
3. RELATED WORKS 49
3.1. PROCESS MODEL CUSTOMIZATION 49
3.2. ILLUSTRATIVE PROCESS FOR HANDLING MEDICAL EXAMINATIONS 51
3.3. CONFIGURABLE NODES APPROACHES 52
3.3.1. Configurable Event-driven Process Chains (C-EPC) 52
3.3.2. Configurable YAWL (C-YAWL) 57
3.3.3. Configurable Integrated EPC (C-iEPC) 59
3.3.4. Application-Based Domain Modelling (ADOM) 62
3.4. CONFIGURATION MECHANISMS FOR CUSTOMIZING PROCESS MODEL 65
3.4.1. Element annotation 65
3.4.2. Activity Specialization 67
3.4.3. Fragment Customization 68

3.5. DECISION SUPPORT TECHNIQUE 70

3.5.1. Questionnaire-Model approach 70

3.5.2. Feature Diagram 73

3.6. PROCESS MINING AND CUSTOMIZABLE PROCESS MODEL 74

3.7. ONTOLOGIES AND CUSTOMIZABLE PROCESS MODEL 76

3.8 SYNTHESIS.. 76

4. FRAMEWORK FOR CUSTOMIZING PROCESS VARIANTS THROUGH PROCESS MINING AND SEMANTIC REASONING 80

4.1. DISCOVERING THE PROCESS MODEL 83

4.2. OBTAINING AN EVENT LOG 83

4.2.1. Coloured Petri Net 85

4.2.2. Extracting process variants.. 88

4.3. DEVELOPING THE QUESTIONNAIRE-MODEL APPROACH 96

4.4. USING ONTOLOGIES FOR PROCESS MODEL CUSTOMIZATION 97

4.5. SYNTHESIS……………………………………………………………………….... 103

5. CASE STUDY.. 105

5.1. CASE DESCRIPTION.. 105

5.2. CONDUCTING THE CASE STUDY.. 106

5.3. DISCUSSION OF THE RESULTS.. 128

6. DISCUSSION AND CONCLUSIONS 130

APPENDIX A – QUESTIONNAIRE FOR PROCESS VARIANT SELECTION 136

APPENDIX B.1 – ONTOLOGY RELATED WITH THE VARIATION POINTS....... 140

APPENDIX B.2 – ONTOLOGY RELATED WITH THE CLINICAL GUIDELINE FOR THE TREATMENT OF ACUTE ISCHEMIC STROKE......................................141

APPENDIX B.3 – MERGED ONTOLOGY 142

7. REFERENCE 143

1. INTRODUCTION

To maintain competitiveness, organizations need to be able to adapt its processes in a fast and flexible way according to the business requirements or according to changes that may happen in the environment in which they operate (VALENÇA, 2013). These changes may happen for different reasons such as changes in the internal and external regulations, customers' attitudes, new technologies, among others.

These aspects lead to the existence of several versions of the same process model which need to be managed. These versions, called process model variant or process variant, may have the same or similar business objective, but differing in their logic due the application context (AYORA et al., 2012). As consequence, organizations need to deal with a large number of processes. For example, Hallerbach (2009a) reports a study case in the automotive industry where more than 900 process variants related with vehicle repair and maintenance in a garage were found. Another example is presented by Li (2010), in which the author analysed the processes for handling medical examinations and identified more than 90 process variants.

The process variants can be managed in one of two ways: by maintaining the process variants separately in repositories or by maintaining them in a single process model from which the process variants can be individualized (LA ROSA et al., 2017). However, it would be inefficient to design each process variant from scratch since this is a complex, error prone and time-consuming task. Also, maintaining large number of business processes is costly for organizations (ASSY CHAN and GAALOUL, 2015; AYORA et al., 2012).

Thus, many approaches have been developed focusing on individualize a process variant from a single model which represents the behaviour of all process variants, such as PESOA (PUHLMANN et al., 2005), Provop (HALLERBACH, BAUER and REICHERT, 2008), C-EPC (ROSEMANN and VAN DER AALST, 2007), C-iEPC (LA ROSA et al., 2008; LA ROSA et al., 2011), among others. These approaches are known as customizable process models (LA ROSA, 2017 et al.; ASADI et al., 2014). In this way, the customizable process model is a step forward enabling to take advantage of the commonalities between the related variants but maintaining its differences (AYORA et al., 2013; SCHONENBERG et al., 2008).

A customizable process model represents the complete behaviour related to a business context enabling to obtain a process variant to a particular situation according to transformations in the process model. The process model elements that can be customized through transformations are known as variation points. This method fosters model reuse and facilitates the maintenance and management of the process variants (DERGUECH, VULCU, and BHIRI, 2010, AYORA et al., 2015). As drawback, the original process model can be very complex and difficult to understand (LA ROSA et al., 2017).

The goal of customizing a process model is to adapt the model according to the user's individual needs. Thus, customizing a process model means that only the desired behaviour is depicted by the process model (GOTTSCHALK, 2009).

There are two ways that a process model can be customized: by restriction or by extension (LA ROSA et al., 2017). In the customization by restriction (also called configurable process model), the customizable process model represents the behaviour of all process variants. Then, a process variant is individualized by removing the undesired behaviour. In the customization by extension, the customizable process model represents the most common behaviour. In this approach, the customization is performed by extending the process behaviour (LA ROSA ET AL., 2017; ASADI et al., 2014).

Designing the single process model representing all process variants and the adjustments for individualizing the process variants is a challenging task (HALLERBACH, BAUER & REICHERT, 2010). When a process variant is obtained it is necessary to ensure the structural and behavioural correctness (i.e., that all nodes are connected and the process do not present deadlocks or livelocks) (LA ROSA, 2009; VAN DER AALST et al., 2008).

Ensuring the evolution is also a challenge faced by the customizable process model. The need for evolution happens when there is a need to introduce new variation points and/or new variants. Another challenge is the re-configuration of a running process variant instance necessary to allow it to switch from the current process variant model to another one (AYORA et al., 2012). La Rosa et al. (2017) points out that there is a need for methods and tools to support the user in the creation, use, and maintenance of the customizable process models. Also, little attention has been paid in providing guidance for the users during process model customization.

A customizable process model is characterized by the existence of points in which multiple variants exist (LA ROSA, DUMAS, and TER HOFSTEDE, 2009). These points are known as variation points (VALENÇA, 2013; TORRES et al., 2012; AYORA et al., 2012). Thus, the process model is customized by selecting an alternative in the variation points. Each alternative has attached rules that define its selection. These rules are related with the requirements of the application context. In this way, process mining techniques can be applied to discover these rules.

Process mining is a technique that analyses an event log, which record all the information about the process execution, thus enabling to extract a process model, monitor deviations by comparing model and log, discover the social network, automated construction of simulation models, among others (VAN DER AALST et al., 2011).

There are several algorithms to mining a process model aiming to analyse different aspects of the process model such as the control-flow or process perspective (e.g., control flow mining, heuristic miner, region mining, alpha algorithm), organizational perspective (e.g., Social Network Miner) and the data or case perspective (e.g., Decision Miner) (LAKSHMANAN and KHALAF, 2013; CHENG and KUMAR, 2015; VAN DER AALST and WEIJTERS, 2004; VAN DER AALST, REIJERS and SONG, 2005; ROZINAT and VAN DER AALST, 2006).

Thus, applying process mining enables to verify if the process variants follows the requirements, enabling also to discover problems that may exist. As result, process mining can provide understanding about the information need for the process variant customization and in which point of the process the information must be available.

The process model customization relies on the decisions that are performed in the variation points. The choices available for each variation point are based on the information from the context in which the process model should be employed. Thus, this information defines the combination of available choices and may be expressed as configuration requirements (hard constraints) and configuration guidelines (recommendations) (VALENÇA, 2013).

Recommendations can be provided to guide the user in selecting the process variant that fits better with the user's needs. Besides, these recommendations can be related with the process model customization (i.e., business process rules), but also related with the business context aiming to improve the business process. For example, some recommendations may be not related with the choice of an available alternative related to a variation point, but related with best practices that should be followed.

As result, the amount of information for customizing a process model may be extremely large. In this way, ontologies can be applied to support the process model customization. Ontologies formalizes the concepts of a domain and the relations between them aiming to provide a shared and common understanding of a domain that can be communicated between people and application systems (FENSEL, 2000).

In this way, ontologies can structure all the information need for customizing a process model. In addition, ontologies enable the use of semantic reasoners, which can be applied for deriving new knowledge, ensure the quality of the ontology, to find contradictory concepts, to derive implied relationships, among others (HAAV, 2004; MARTINEZ-GIL, 2015; OBITKO, 2007; ABBURU, 2012). Considering all the aspects mentioned previously, the research goal of this research can be defined.

1.1. RESEARCH GOAL

Based on the drawbacks related with the process model customization and considering the advantages provided by the process mining techniques and the semantic reasoning, this research pursues the following research goal:

Develop an approach for process model customization, which provides a decision-making support for the user and enable to individualize a process model that respects the user's requirements and the internal and external regulations

by:

- Applying process mining techniques to build a customizable process model, enabling to identify the process variants and the rules for selecting them;
- Formalizing the relevant knowledge about the business context in ontologies, and through semantic reasoning provide support for process model customization.

1.2. RESEARCH QUESTIONS

In order to achieve the research goal, this research can be divided in two parts. The first part aims to provide an understanding about the aspects related with the customizable process model, process mining, ontologies and the relations between them. The second part aims to providing an understanding about how customize the process model by means of the process mining and the semantic reasoning.

Based on the objectives from the first part the research questions can be decomposed into sub-questions. Thus, answering these sub-questions lead to answer the related research questions and consequently to achieve the research goal:

- **RQ1: How to customize a process model in order to obtain a process variant that correctly represent a business context?**
    - **RQ1.1: What are the aspects that need to be considered when building a customizable process model?**
    - **RQ1.2: How the existing approaches proposes to customize a process model?**
- **RQ2: What are the theoretical and practical arguments motivating the application of process mining to discover customizable process models?**
    - **RQ2.1: What are the process mining techniques that can be applied to identify the aspects related with the process variants enabling to improve the customizable process model?**
    - **RQ2.2: How to improve the process model from which each process variant is individualized to consider scenarios that are not available in the event log?**
- **RQ3: What are the theoretical and practical arguments motivating the use of ontologies for process model customization?**
    - **RQ3.1: Can ontologies be applied to provide decision making support during the process model customization?**

Based on the results of the first research part, the second part was developed. We point out that a different result of the first part would lead to a different set of research questions and, thus a set of methods to solve the problem. The first part showed that process mining techniques and ontologies can improve the process model customization. This result led to the development of an approach to customize a process model. Based on this, the research question of the second part can be formulate:

- **RQ4: How process mining and ontologies can be applied to customize a process model according to all the requirements related to a particular business context?**

Considering the business context related with the process model customization, the research goal and the research questions, three hypotheses need to be formulate:

- **H1: All the knowledge on a domain, including the related regulations, has been captured and formalized in an ontology.**
- **H2: An event log is available, which can have partial information about the business execution.**
- **H3: The privacy issues, that may exist, have been solved.**

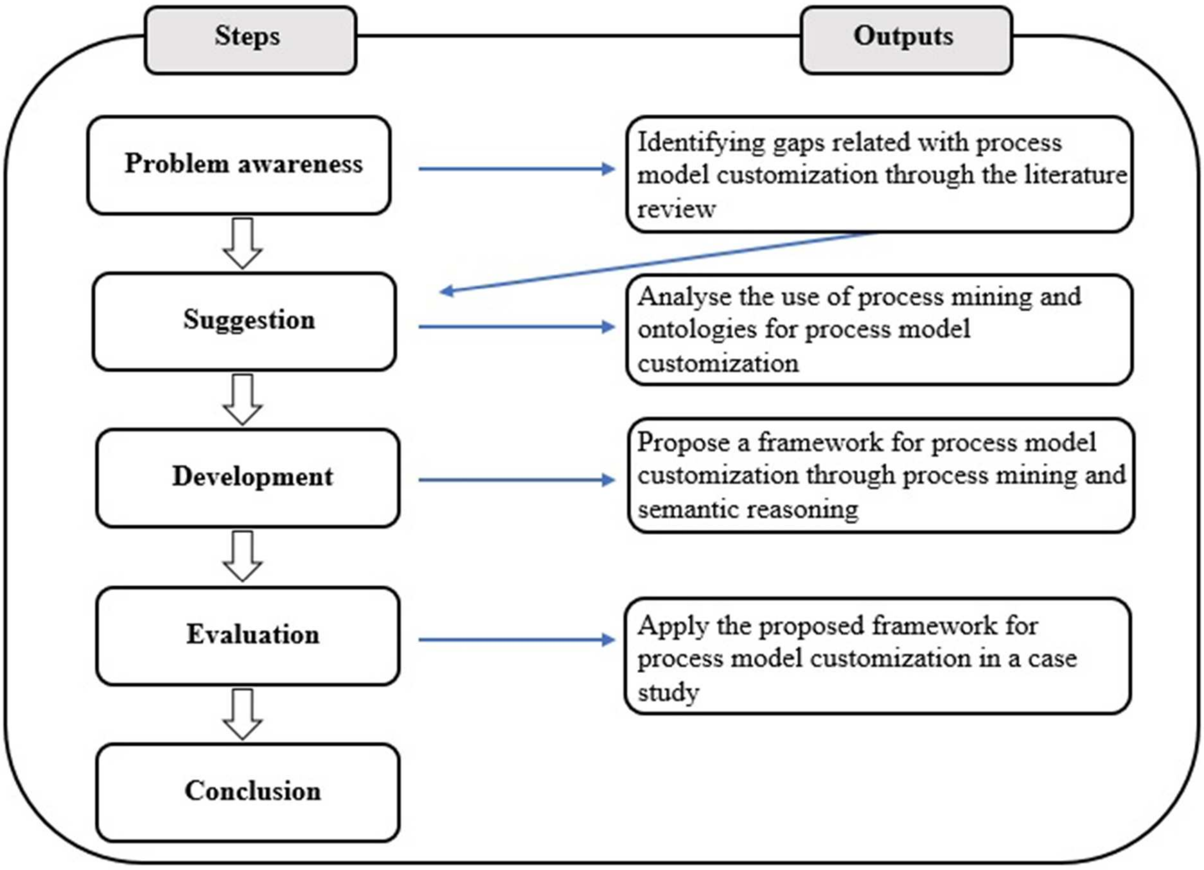
These hypotheses are supported by related researches in the corresponding domains. H1 is supported by studies on knowledge discovery, conversion and formalization (GRUBER, 1993; GUARINO, 1995). For hypothesis H2 has considered studies dealing with the aspects related with event log such as incompleteness and noise, among others (MARUSTER et al., 2006, FOLINO et al., 2009; ROGGE-SOLTI et al., 2013; VAN DER SPOEL, VAN KEULEN and AMRIT, 2012). H3 is possible to be achieved and privacy issues, related with the data protection during the mining process have been discussed by several authors such as, Oliveira and Zaiane, (2002); Yoo et al., (2016); Burattin, Conti and Turato, (2015), among others.

1.3. METHODOLOGY

A scientific research is composed by a set of actions aiming to discover the solution for a problem through scientific procedures (MINAYO, 1993). There are several procedures for developing a research ranging from informal to the strictly scientific procedures (KOTHARY, 2004).

The development of this research follows the principles of the design science research method (MARCH AND SMITH, 1995). The design science research is composed by a set of analytical techniques which consists in a rigorous process of projecting artefacts to solve problems, evaluate what was planned or what is being executed and communicate the results (LACERDA et al. 2013). The main stages of the design science research are depicted in Figure 1.

Figure 1 - Main stages of the design science research



Source: adapted from Takeda et al., 1990

As shown in Figure 1, the first step to conduct the design science research is the formalization of a problem in the context under analysis. The literature review is one of the methods to identify gaps and/or problems that need to be solved. Thus, through the literature review the gaps related to the customization of process model can be identified. The second element refers to identify ways to solve the problem identified in the first step. In this way, in this step is analysed whether process mining and semantic reasoning can be used to solve the problems previously identified (DRESCH et al., 2015; VAISHNAVI, KUECHLER and PETTER, 2004; TAKEDA et al., 1990).

The next element is the development of artefacts (things or processes) to solve the problem previously identified, to make contributions, to evaluate projects and to communicate the results. Thus, in this step the framework for process model customization through process mining and semantic reasoning is proposed. The forth element is the evaluation of the artefact considering the criteria of the proposed solution. For evaluation, the proposed framework is applied in a case study related with the healthcare environment. The last element refers to communicate the results of the proposed solution (DRESCH et al., 2015; VAISHNAVI, KUECHLER and PETTER, 2004; TAKEDA et al., 1990).

1.4. CONTRIBUTIONS

The main contributions of this thesis can be summarized as follows:

- An approach to guide users during the process model customization. The approach also enables to provide recommendations about the business context, thus enabling to improve the business process. In addition, the resulting process variant follows the user's requirements and the regulations related to a particular business context.

- The possibility to discover which information is necessary for customizing the process model, when the information must be available and how a decision made in one point of the process model impact the other decisions.

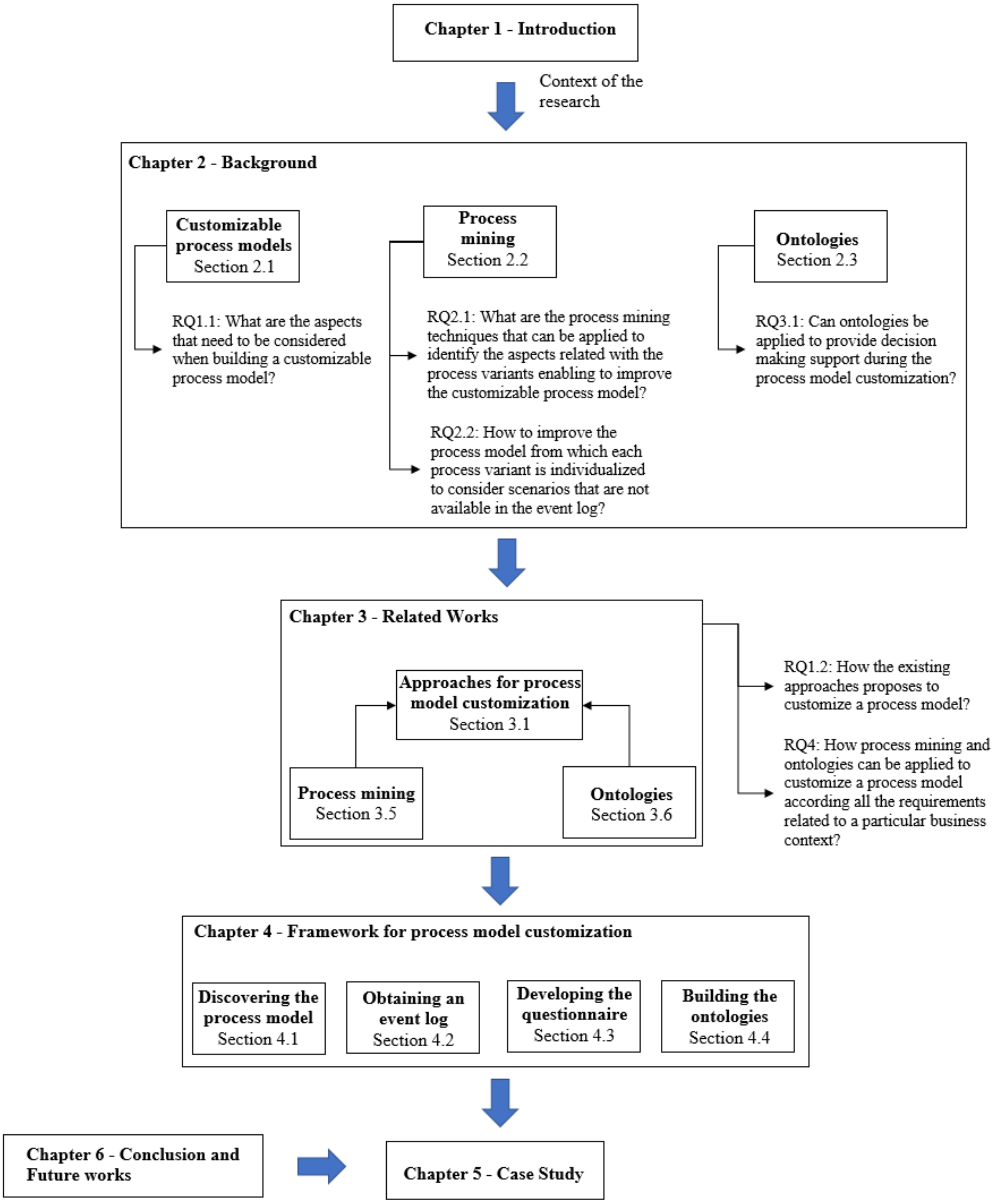
- The possibility of applying process mining to identify process variants from the data related to the business process execution, thus enabling to verify if the rules to select them are correctly defined. It also enables to identify deviations that may exist in the customizable process model and to obtain a process model closer to the daily activities performed in the context under analysis.

- The approach also enables to obtain a process model that contain all the behaviours that may exist in the business context under analysis, not just the behaviour existing in the data about the process executions. The approach also enriches the process model with expert knowledge and internal and external regulations enabling to obtain a prescriptive process model which addresses all the relevant issues about the business context.

1.5. OUTLINE OF THE THESIS

This thesis is structured into 5 chapters aiming to answer the research questions and thus, the research goal as depicted in Figure 2.

Figure 2 - Outline of the Thesis



The first Chapter sets the research context and the research goal summarizing the research's contributions and including the research methodology. Chapter 2 presents the literature review about customizable process model regarding its aspects, methods and challenges. Process mining is also discussed, including the type of process mining analysis and algorithms. Finally, ontologies are defined, including its elements, classifications, languages among others. The development of these chapter provides the understanding of the concepts behind the development of a customizable process model, the process mining techniques and the ontologies, thus enabling to answer the questions RQ1.1, RQ2.1, RQ2.2 and RQ3.1.

Chapter 3 addresses the approaches for process model customization which enable to identify its characteristics and contributions. Moreover, the application of process mining and ontologies to customize process models is discussed, thus leading the answers for questions RQ1.2 and RQ4.

Based on the knowledge gathered in the previous chapters, a framework is developed for process model customization, which is depicted in Chapter 4. This chapter present each step for the framework development. Chapter 5 presents a case study to validate the framework for process model customization related to the treatment provided to the patients diagnosed with acute ischemic stroke. Finally, Chapter 6 concludes the thesis by summarizing the contributions and discussing the further research.

2. BACKGROUND

This chapter discusses in more detail the main areas addressed in this research. As mentioned previously, a customizable process model is capable of represent all the behaviour that the business process may have according to the business requirements. The customization of a process model relies on the choices made in the points of the model that are subject to variation (i.e., variation points). Thus, rules are attached with the alternatives related with the variation points. These rules are defined based on the internal and the external regulations (AYORA et al., 2016; LA ROSA et al., 2017; KUMAR and YAO, 2012).

However, if the rules are not defined correctly or if the process model is not correctly modelled, then the resulting process variants are also incorrect. Process mining can be applied to identify the process variant in a customizable process model and the rules for select them, enabling to verify if the rules are correctly defined, i.e., respecting all the requirements from the business context (ROZINAT and VAN DER AALST, 2006b). Process mining can also show deviations that may exist in the customizable process model, thus enabling its improvement (BOSE and VAN DER AALST, 2012; HUANG et al., 2013). The resulting process model reflects the business context and can be correctly customized.

In each variation point, the user need to make a decision, which impact the customization of the process model (VALENÇA et al., 2013; LA ROSA, DUMAS and TER, 2009). However, even if the user is familiarized with the process, may not be easy to estimate every impact, mainly in highly dynamic process models. Thus, guidance and recommendations during the customization is essential to help the user in the selection of a choice that best fits its goal (LA ROSA et al., 2017; ROSEMANN and VAN DER AALST, 2007). Besides, recommendations can also be made in relation to several aspects related with the business process context, not just the rules related with the variation points. However, these recommendations are only relevant if they are provided according to the selection made by the user in each variation point.

The definition of the rules related with each alternative in the variation points, as well as the recommendations, relies on the internal and external regulations, which may involve a large amount of information. In this way, the information related with the business context and the information related with the process variants (variation points, alternatives and rules) can be formalized in ontologies. The ontology can be used to carry out a reasoning process. Thus, reasoning tools enable to make inference from the formalized knowledge on the ontologies providing as output guidance and recommendations for the user. Based on these concepts, Figure 3 can be developed:

Figure 3 - Process model customization through process mining and ontologies

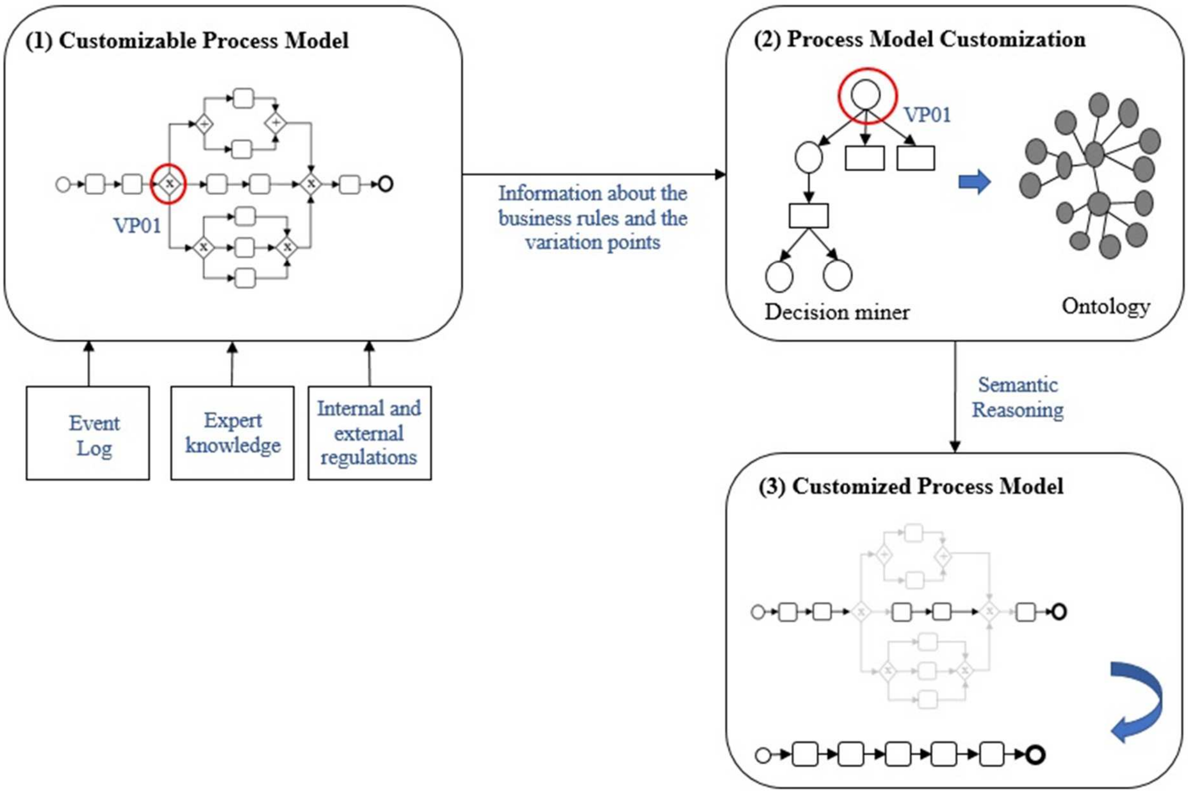


Figure 3 shows that decision mining and ontologies can be applied for process model customization. Decision mining can be used to discover the variation points, the alternatives for them and the rules for select them. And the ontology can formalize the knowledge for customizing the process model.

Therefore, this chapter addresses the relevant background knowledge to reach the objectives proposed in this research. Section 2.1 introduces the concept of flexible business process. Section 2.2 discusses how variability in business process can be managed. Section 2.3 discusses the type of process mining and the analysis provided by this technique, enabling to identify the algorithms that can be applied to obtain a customizable process model. This section also identifies some drawbacks regarding the process mining techniques and the solutions that have been proposed to overcome them, including the combination between the business process and semantic technologies. Section 2.4 discusses the concept of ontology, focusing on the aspects needed to build an ontology from existing knowledge.

2.1. BUSINESS PROCESS MODEL AND FLEXIBILITY

A business process can be defined as a set of connected activities which together pursue a particular business goal. The business processes can be linked with organizational structure defining functional roles and organizational structures. In addition, it may involve one department or can cross departmental borders or involve different organizations (WEBER and REICHERT, 2012).

Business process management is the combination among the knowledge from information technology and the knowledge from management sciences and its application to operational business processes. It includes methods, techniques, and tools to support the design, enactment, management, and analysis of such operational business processes (VAN DER AALST, 2004; VAN DER AALST, 2013).

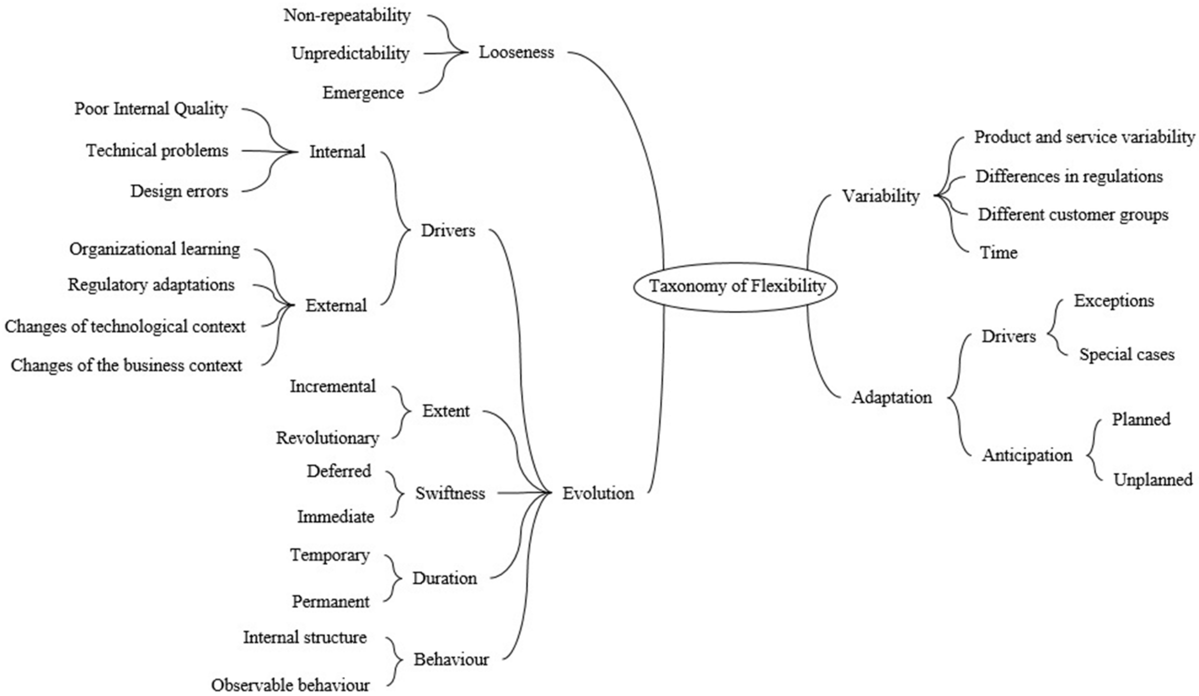
The need for flexible behaviour of an organization requires also more flexible information systems (PROPER and VAN DER WEIDE, 1995). Thus, Process Aware Information Systems (PAIS) has been developed in order to deal with flexible business process. PAISs are able to deal with exceptions and uncertainty, change the execution of single business cases on the fly, deal with variability and support the evolution of business process models (REICHERT and WEBER, 2012).

According to Weske (2007), flexibility is the main driving force behind business process management in an organizational level, where strategic business processes are investigated, and at operational level, where human interaction workflows and system workflows are important concepts for realizing business processes.

Cambridge Dictionary (2017) defines flexibility as the ability to change or be changed easily to suit different situations. According to Merriam-Webster dictionary (2017) flexibility is the capability to adapt to new, different, or changing requirements.

In turn, process flexibility is the ability of a process model to adapt according to the foreseen and unforeseen changes that may happen in the environment in which they operate. Flexibility is related with those parts of the process model that should be changed and with the parts that need to stay the same (VAN DER AALST, 2013, SCHONENBERG et al., 2008). Weber and Reichert, (2012) define a taxonomy of process flexibility as shown in Figure 4.

Figure 4 - A taxonomy of process flexibility needs



Source: Reichert and Weber, (2012).

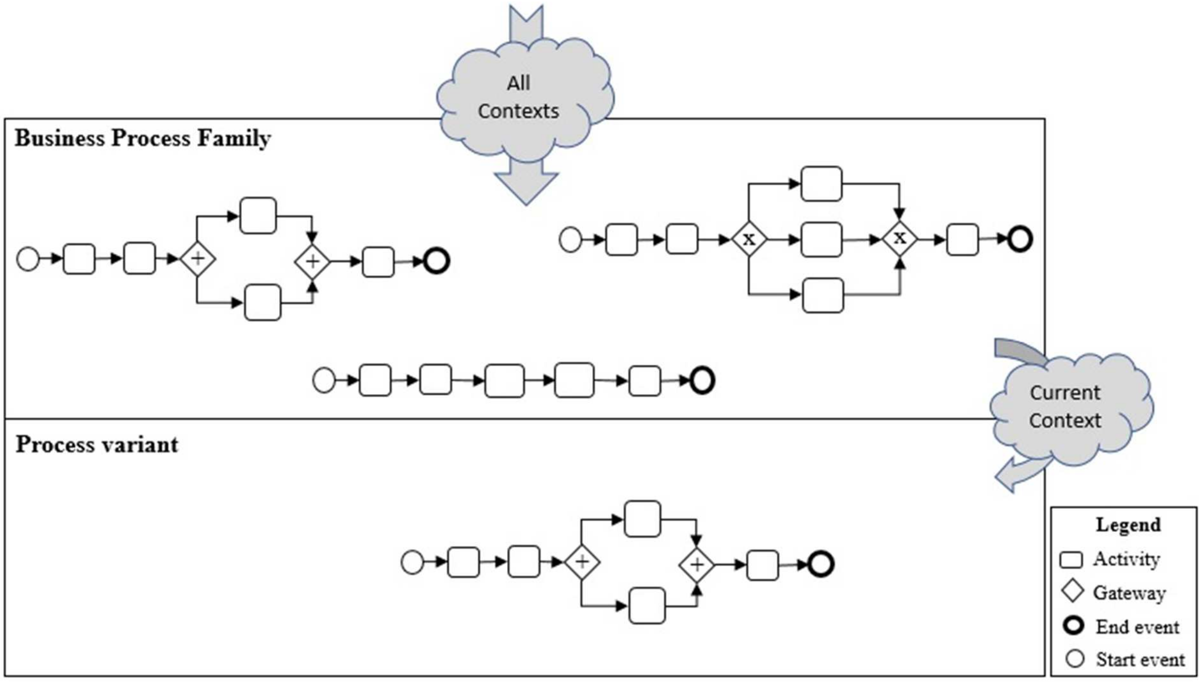
Reichert and Weber, (2012) described each element of the taxonomy of process flexibility as:

- *Looseness*: process in this category cannot be fully prespecified. The goal of the process is known a priori, however the process model logic cannot be determined and it might change during the process execution.
- *Evolution*: represents the ability of the process to evolve in order to ensure the alignment between the real-processes and the PAISs.
- *Variability*: is related with the development of process variants, which as mentioned before, share the same or similar business objective but differ in their logic. Thus, a process variant is an adaption of the original process model aiming to represent a specific set of requirements from a business context.
- *Adaptation*: refers to the need to adapt one or several process instances in order to realign the computerized process with real-world processes.

2.2. BUSINESS PROCESS VARIABILITY

Business process variability is the ability of a process to adapt to the changes in the environment or to its changing requirements (VALENÇA et al., 2013). Thus, through some transformations the process model can be changed to represent only the desired behaviour (GOTTSCHALK, 2009). As result, several process models can be derived from the same process. Figure 5 shows three versions of the same process model.

Figure 5 - Process variants and process family



Source: adapted from Ayora et al. (2012).

Each related process model is called a process variant, and a collection of process variants is known as a process family (AYORA et al., 2016). The process variants may follow the same or similar business objectives. However, they differ in their logic due the differences of their varying application contexts. Thus, the respective process variants can contain common activities but they also differ from each other because some activities are only relevant for a specific context of application (AYORA et al., 2015; REICHERT and WEBER, 2012).

The process variant reflects the awareness of process constraints and requirements which provide valuable insight into work practice, help externalize previously tacit knowledge and provide valuable feedback on subsequent process design, improvement and evolution (MAHMOD AND CHIEW, 2010). However, managing process variability is a non-trivial task as it requires specific standards, methods and technologies to support process variability (VALENÇA et al., 2013).

2.2.1. Managing business process variability

When dealing with process variability one of two options must be chosen. The first option is to define and maintain the process variants in separate process models, which is known as multi-model. The result is a highly redundant model in which the process variants are not strongly connected with each other. It also does not provide any support for combining or merging existing variants to new ones (HALLERBACH, BAUER and REICHERT, 2009a). Besides, designing and implementing each process variant from scratch and maintaining it separately would be inefficient and costly for companies (AYORA et al., 2013).

The second option is the single-model, which aims at supporting the representation of a family of business process variants via a single model. As a drawback, it leads to highly complex process models that are difficult to comprehend, analyse and that are expensive to maintain. Besides, normal branching cannot be distinguished from the ones representing a variant selection (HALLERBACH, BAUER and REICHERT, 2009a; LA ROSA, et al., 2017). As benefits, it eliminates redundancies by representing variant commonalities only once. It also fosters model reuse, i.e., parts of the model can be shared among multiple variants and reduces modelling efforts (AYORA et al., 2012; AYORA, et al., 2016).

Considering these two options, several approaches have been developed to model families of business process variants into a single model, which enable to obtain a process variant through some transformations such as add, delete or move, that can be applied in the process model. The literature refers to such consolidate model by different terms.

Some authors refer to these models as configurable process model, which is a process model that is capable of representing the complete process family. Thus, a process variant can be configured in a behavioural and structural way (AYORA et al., 2012; GOTTSCHALK et al., 2009). The behavioural approach integrates all process variants into one process model representing the commonalities and differences reflecting the complete behaviour of all variants. The structural approach, represents the parts of the models that can be separately changed. Thus, only the commonalities are represented in the model (called base process model) to which structural changes may be applied to derive process variants (TORRES et al., 2012).

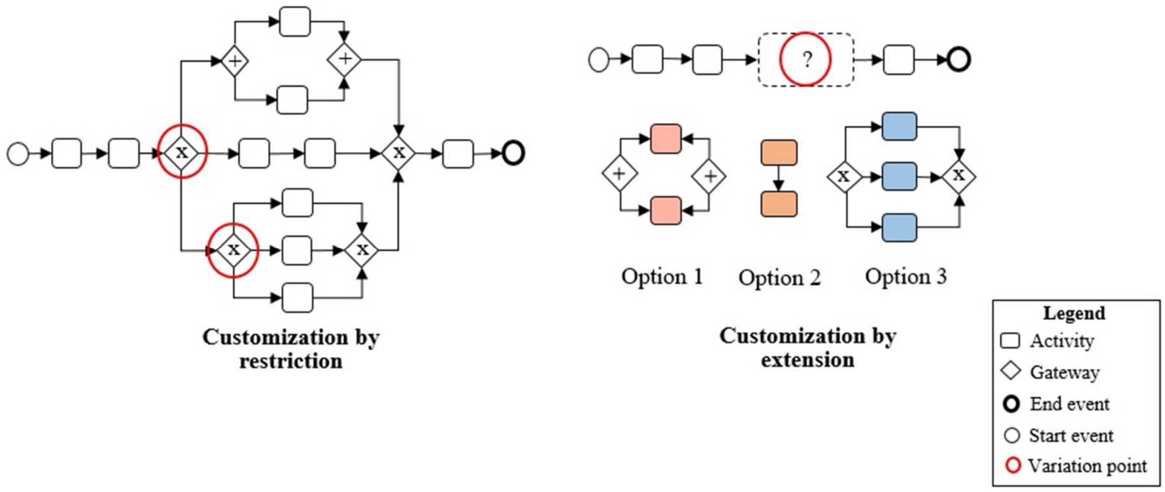
The representation of a process family into a single process model is also defined as a Customizable Process Model. Two approaches were defined to derive the process variants: the customization by restriction and customization by extension (LA ROSA et al., 2017).

Customization by restriction refers to a customizable process model that contains all behaviour of all process variants. In this approach, the customization is achieved by restricting the behaviour of the customizable process model using the delete operation. In the second approach, the customizable process model represents the most common behaviour, or the behaviour that is shared by most process variants. For the customization, the behaviour needs be extended using the insert and modification operators to represent a particular situation (LA ROSA et al., 2017; ASADI et al., 2014). It can be noted that the customization by restriction and by extension corresponds to the behavioural and structural approaches respectively.

Some authors, such as Asadi et al. (2014) and Assy, Chan and Gaaloul (2015), refer to the term Configurable Reference Process Model, which represents a family of similar process models and describes multiple variants of a process model in an integrated way. Reference process models are conceptual models that illustrate generic solutions for a certain domain. These models capture common knowledge and best practice. However, there is no comprehensive support for explicitly describing variation points (REICHERT and WEBER, 2012, LA ROSA et al., 2017; MEERKAMM and JABLONSKI, 2011).

In this research, were adopted the definitions proposed by La Rosa et al., (2017). Thus, customizable process model is the term used and the types of customization are referred as customization by restriction and customization by extension. However, the terms customization and configuration can be used interchangeably. The Figure 6 presents an example of the customization by restriction and by extension.

Figure 6 - Approaches to define a customizable process model



Source: adapted from Ayora et al. (2012).

Figure 6 shows that both approaches for customizing the process model are characterized by the existence of the variation points, i.e., the parts of the process model that are subject to variation. By analysing these points, one can identify the alternatives available for each variation point and the associate rules, i.e., the reason(s) to select the alternatives. These aspects are essential to represent process variability (TORRES et al., 2012; AYORA et al., 2012).

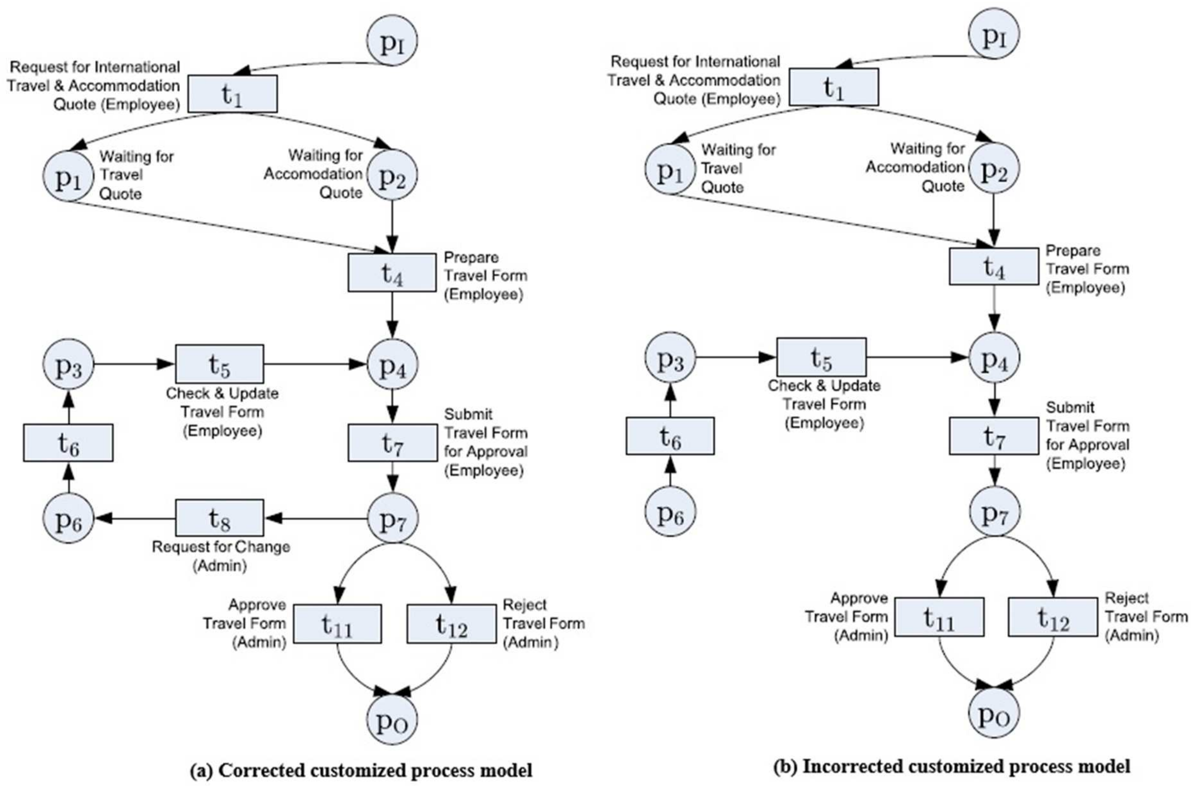
Thus, the selection of a process variant relies on the decisions made at the variation points (LA ROSA, DUMAS and TER HOFSTEDE, 2009). Such decisions can be made at design time or at runtime. Decisions made at design time are known before process execution and affect all instances of the customized process. Decisions at run-time have effect for one or few process instances. Besides, a decision in one variation point can have direct implication on the other variation points (LA ROSA et al., 2017; AYORA et al., 2016; GRÖNER et al., 2013).

The selection of the most suitable variant is called customization or configuration. Once all variation points have been configured, start the process called individualization, which refers to derive a process variant by dropping those parts of the model that are no longer needed (LA ROSA, DUMAS and TER HOFSTEDE, 2009; REICHERT and WEBER, 2012).

After the creation of the customizable process model, it must be ensured that the resulting process variants are correctly in a structural (or syntactical) and behavioural (or semantical) way. Structural correctness ensures that, during the configuration, the selected activities are re-connected, avoiding disconnected nodes (VAN DER AALST et al., 2008, LA ROSA, 2009). In other words, structural correctness means that every edge is on a path from a start node to an end node (ROSA et al., 2013).

Behavioural correctness is related to ensure that the model is sound, i.e., with no deadlocks or livelocks (VALENÇA et al., 2013). Deadlocks happen when the path reaches a non-final state without any outgoing transitions. The transition system may also have livelocks, that are similar to deadlocks, except that the states of the processes involved in the livelock constantly change with regard to one another, none progressing. Livelock is a special case of resource starvation. The general definition only states that a specific process is not progressing, i.e., some transitions are still enabled but it is impossible to reach one of the final states (VAN DER AALST, 2011). Additionally, the configurable process model must be validated, i.e., it must be ensured that the business requirements are properly reflected by the model (AYORA et al., 2015). Figure 7 shows an example of a correct (a) and an incorrect (b) customized process model.

Figure 7 - Corrected and incorrected customized process model



Source: adapted from van der Aalst et al., (2010).

Figure 7 presents two process variants related with a customizable process model for travel requisition approval as a Petri net. A Petri net is a graphically and mathematical tool consisting of a directly graph with two nodes: transitions (drawn as bars or boxes) and places (drawn as circles) connected by arcs (MURATA, 1989; DESEL and ESPARZA 1995; REISIG and ROZENBERG, 1998). Figure 7(a) shows a net correctly customized enabling to reach the end state. In Figure 7(b), the elements t5, p3, t6 and p6 are unreachable. Thus, this net is not correctly customized, but it contains behavioural and structural problems.

When configuring process variants, one challenge is to design a single basic process model from which the process variants can be configured. Another challenge is to design, model and structure the adjustments that may be applied to configure the different process variants to this basic process model (HALLERBACH, BAUER and REICHERT, 2009a).

Asadi et al., (2014) define four important challenges when dealing with process model customization:

- *Variability complexity:* refers to capture and model how the process variants can differ from each other;
- *Modelling complexity:* incorporate the selection points (i.e., variation points and process variants) and dependencies in a single model in addition to process logic increases the complexity of the process model for development and customization;
- *Delta requirements:* it is unlikely that every requirement is covered by the process model. Thus, customization approaches should provide the developers with mechanisms for making changes to the customized process model to fit it to target application requirements;
- *Customization validation:* the customization approach should guarantee the correctness and compliance of the process variants with respect to the specified configuration and behavioural constraints and inform process engineers of possible inconsistencies.

According to Ayora et al., (2012) the run-time flexibility and evolution of single process variants have not been sufficiently considered so far. Run-time flexibility is concerned with the configuration decisions that only can be made at run-time when the related information is available. Thus, the challenge is related to decide by whom, when, and based on which information run-time configurations may be made. Another challenge is the re-configuration of a running process variant instance necessary to allow it to switch from the current process variant model to another one. Evolution of single process variants refers to the run-time situations where the process variant needs to change to realign its specification to real-world business case. In addition, changes in a single process variant model may require checking whether other process variants are affected as well.

2.3. BUSINESS PROCESS MINING

Information systems such PAIS (Process Aware Information System), ERP (Enterprise Resource Planning) systems, CHS (Case Handling Systems) and CRM (Customer Relationship Management) systems are extremely efficient in record the data about the execution of business process. Thus, the analysis of the recorded data can provide insights about the business process.

Process mining aims to discover, monitor, and improve business process by extracting knowledge from the data logs recorded by the information systems (MANS et al., 2013). As process mining is based on real data, it addresses the problem that most business have very limited information about what is happening in their organization. Thus, it can be considered as a proficient means for helping organizations understanding their actual way of working and can serve as a foundation for process improvement (WESKE, 2012; GÜNTHER et al., 2008).

The base of process mining are the event logs (also known as ‘history’, ‘audit trail’ and ‘transaction log’) that contain information about the instances (also called cases) processed in systems, the activities (also named task, operation, action or work-item) executed for each instance, at what time the activities were executed and by whom, named respectively as timestamp and performer or resource. Event logs may store additional information about events as costs, age, gender, etc. (JANS et al., 2011; VAN DER AALST, 2012).

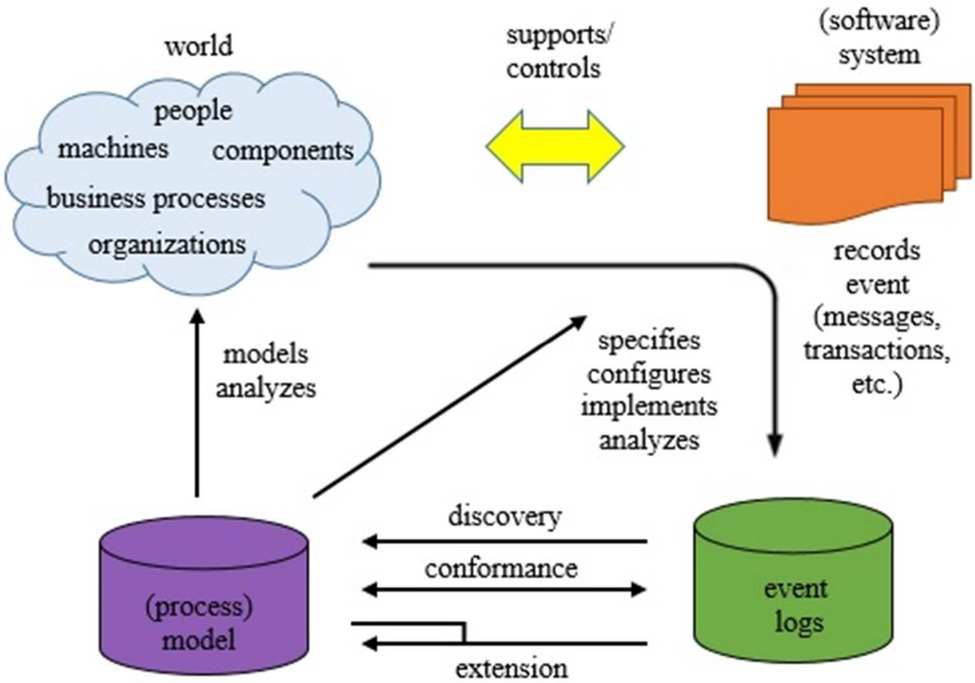
In order to use process mining, some assumptions are made (VAN DER AALST, et al., 2012):

- Each event refers to an activity (i.e., a well-defined step in the process);
- Each event refers to a case (i.e., a process instance);
- Each event can have perfomer also referred to as originator (the person executing or initiating the activity);
- Events have a timestamp and are totally ordered.

2.3.1. Types of process mining

Process mining techniques can provide three types of analysis: discovery, conformance, and enhancement, as shown in Figure 8.

Figure 8 - Types of process mining: discovery, conformance and enhancement



Source: adapted from van der Aalst, 2011.

According to the Figure 8, there is a relation between “world” and some (software) system. The system may support or control all kinds of processes taking place in the real world. Moreover, most systems record events about the activities that have been executed in the event log format. The Figure also show the existence of the process models that can model the “world” and the systems. Example of process models are BPMN diagrams, EPCs, UML activity diagrams, social networks, among others (VAN DER AALST, 2007).

As shown in Figure 8, discovery is the first type of process mining. This type aims the automatic extraction of the process model from the log data without any a priori information. Conformance is the second type, which aiming at checking if modelled behaviour matches the observed behaviour. This comparison shows where the real process deviates from the modelled one. Moreover, process mining techniques can quantify the level of conformance and diagnose differences. The third type, enhancement, seeks to detect deviations to enrich the model, e.g., show bottlenecks in a process model by analysing the event log (ROZINAT et al., 2009; VAN DER AALST and DUSTDAR, 2012).

Orthogonal to the three types of process mining, there are at least three perspectives (SONG and VAN DER AALST, 2008; VAN DER AALST, 2011; HOMAYOUNFAR, 2012):

- *Control flow perspective*: focuses on the control flow, i.e., the ordering of activities. The goal of this perspective is to find a good characterization of all possible paths, generating a process model that reflects the current observable process in reality. Exists some algorithms that can be used in this perspective as control flow mining, heuristic miner, region mining, alpha algorithm, etc.;

- *Organizational perspective*: focuses on information hidden in the log and describes the organizational structure in terms of roles, organization units, handover of work or social networks;
- *Case perspective*: focuses on properties of cases which can be characterized by their path in the process, by the originators working on a case values or by the values of the corresponding data elements;
- *Time perspective*: concern with the timing and frequency of events.

According to Günther et al., (2008) process mining can support the following tasks:

- *Usage profiling*: besides describing the actual use of the application, the process model may show that some features are used by a particular group of people or that they are never used;
- *Reliability improvement*: the model result can show failures and the analysis of these failures through process mining can help to find root causes for reliability problems;
- *Usability improvement*: enable to locate and to quantify these deviations from actual use to intended use through conformance checking.
- *Remote diagnostics and servicing*: process mining can help to predict failures and, if an errors occurs, the event log may be used to find the core problem and take counter measures.

2.3.2. Process Mining algorithms

There are many algorithms that can be used in process mining. One of the first discovery algorithm is the α-algorithm, which extract a Petri net that gives a concise model of the behaviour seen in a set of event traces (LAKSHMANAN and KHALAF, 2013). The α-algorithm is simple, it can deal with concurrency and many of its ideas have been embedded in more complex and robust techniques. However, the α-algorithm has problems dealing with noise, infrequent/incomplete behaviour and complex routing constructs (VAN DER AALST, 2010).

Noise refers to incorrect logged information. Traces may be incomplete when certain events are missed, as resulting the log does not allow deriving the process model. Infrequent behaviour indicates the execution of exceptional paths in the process. Further, inconsistencies can arise from naming conventions. These problems can result from data entry problems, faulty data collection instruments, data transmission or streaming problems and other technology limitations (LAKSHMANAN and KHALAF, 2013; CHENG and KUMAR, 2015; VAN DER AALST and WEIJTERS, 2004).

As a consequence of these drawbacks, several process mining algorithms were developed. Some of them are discussed below:

- *Heuristic Miner*: is a discovery algorithm that can deal with noise and low frequent behaviour in event logs (WEIJTERS, VAN DER AALST, 2006). This technique extends alpha algorithm by considering the frequency of event and sequences of event and traces in the log and then infering direct graphs using heuristic techniques (FERNÁNDEZ-LLATAS et al., 2013).
- *Fuzzy Miner*: is a configurable process discovery algorithm that mines behaviour of unstructured process models (GÜNTHER and VAN DER AALST, 2007). It applies a variety of techniques, such as removing unimportant edges, clustering highly correlated nodes in to a single node, and removing isolated node clusters (FERNÁNDEZ-LLATAS et al., 2013).
- *Trace Clustering*: allows split the event log into homogeneous subsets and for each subset a process model is created (SONG, GÜNTHER, VAN DER AALST, 2009). Thus, as each subset have similar traces, the process model of each subset is more concise and understandable compare to the process model from the entire event log (MONTANI and LEONARDI, 2014).
- *Genetic Miner*: using genetic operators, this algorithm seeks to find a process model that can replay all the traces comprised in the log. Its main advantages are the ability to discover non-trivial process structures and its robustness to deal with noise (BRATOSIN, SIDOROVA and VAN DER AALST, 2010).
- *Social Network Miner*: focuses on the relations among individuals (or groups of individuals) acting in the process (VAN DER AALST, REIJERS and SONG, 2005).
- *Decision Miner*: aims at the detection of data dependencies that affect the routing of a case. The algorithm identifies the parts of the model where the process is split into alternative branches. Then, based on data attributes associated to the cases in the event log, the rules for following one route or the other are discovered (ROZINAT and VAN DER AALST, 2006).

These various algorithms are supported by ProM[1] (PROM, 2017), which is an extensible framework that supports a wide variety of process mining techniques in the form of plug-in. The advance of process mining leveraged the development of other tools as Disco (FLUXICON, 2017), Perceptive Process Mining (PERCEPTIVE SOFTWARE, 2017).

Besides, some techniques may be combined with process mining to perform mining of business processes. Some examples are Markovian approach (ROGGE-SOLTI and WESKE, 2015), neural network (MAITA et al., 2017) and cluster analysis (REBUGE and FERREIRA, 2012).

2.3.3. Challenges for process mining

The process mining techniques are well developed, however, despite its benefits, there are some issues to be overcome. Van der Aalst and Dustdar (2012) highlights some challenges faced by process mining:

- *Finding, merging and cleaning event data*: it refers to the efforts to extract event data suitable for process mining such as: data might be distributed over a variety of sources, event might be incomplete, containing outliers or events at different levels of granularity;
- *Dealing with complex event logs having diverse characteristics*: while some event logs may be extremely large, making it difficult to handle, others event logs are so small that not enough data is available to make reliable conclusions.
- *Creating representative benchmarks*: it is important to have benchmark to evaluate the existent tools and stimulate the creation of new tools.
- *Dealing with concept drift*: it refers to the situation in which process is changing while being analyzed. Concept drift in a process can be discovered by splitting the event log into smaller logs and analyzing the "footprints" of the logs. Therefore, additional research and tool support are needed to adequately analyze concept drift.
- *Improving the representational bias used for process discovery*: i.e, the class of process models that can be discovered. The presentational bias determines the search space and potentially limites the expressiveness of the discovery model;

[1] Process Mining framework. Process Mining Group, Math&CS department, Eindhoven University of Technology, http://www.promtools.org

- *Balancing between quality criteria*: in process mining, the quality of a process model can be evalutate through four competing quality dimensions: fitness, it describes to what extent a model can reproduce the behaviour presented in the log; simplicity, it describes the (perceived) conmplexity of a model; precision, the model does not allow for "too much" behaviour; and generalization, it shows the most likely underlying model that is not invalidated by the next set of observations. One of the challenges is to balance between "overfitting" (the model is too specific and only allows for the "accidental behavior" observed) and "underfitting" (the model is too general and allows for behavior unrelated to the behavior observed);
- *Cross-organizational mining*: it is the case when event logs of multiple organizations are available for analysis. In principle, there are two settings for cross-organizational process mining: the first case is when different organizations work together to handle process instances and the second is when different organizations are essentially executing the same process while sharing experiences, knowledge or a common infrastructure;
- *Providing operational support*: process mining can be used for online operational support, such as to detect deviations from the predefined process, and then generate alerts. Process mining can also be used to predict actions that can be take based on predictive models built using historical data and, based on such predictions, one can also build a recommender systems that propose particular actions to reduce costs or shorten the flow time;
- *Improving usability for non-experts*: this challenge refers to the user-friendly interfaces that automatically sets parameters and suggests suitable types of analysis.

Another challenge is related to the fact that the mining techniques are unable to reason over the concepts behind the labels in the log. It is very common the situation where different activities are represented by the same label or different labels are described by the same activity. For this reason, before data analysis is necessary the pre-processing step. To overcome this challenge, semantic technologies were combine with BPM, thus emerging the concept of semantic business process mining (PEDRINACI and DOMINGUE, 2007; De Medeiros et al., 2007).

The basic idea of semantic process mining is to annotate the log with the concept in an ontology, this action will let the inference engine to derive new knowledge (DETRO et al., 2016). The combination of the semantics and the processes can help to exchange process information between the applications in the most correct and complete manner, and/or to restructure business processes by providing a tool for examining the matching of process ontologies (SZABÓ AND VARGA, 2014).

2.4. ONTOLOGIES

A process model is customized through the decisions made at the variation points. However, the customization is not a trivial task. A decision in one point may influence the selection of other parts of the model. But is not possible to foresee this dependency without support tools. Thus, guidance related to the process model customization and the business context help the user in obtain a correctly process model, i.e., according to the user's requirements and respecting the rules of the application context. Thus, ontologies can be used to provide decision-making support during the process model customization.

Ontology has its origins in philosophy and refers to the study of being as such (GAŠEVIĆ, DJURIC and DEVEDŽIC, 2009). Artificial intelligence (AI) borrowed the word and changed its meaning. In AI, the main question is what an AI system has to reason about to be able to perform a useful task (BORST, 1997). Now, its importance is being recognized in research fields as diverse as knowledge acquisition (TUDORACHE et al., 2013), medicine (ARSENE, DUMITRACHE and MIHU, 2015), knowledge representation (YAO and GU, 2013), language engineering (GUIZZARDI et al., 2015), among others.

There are many definitions of the concept of ontology. The most accepted one is from Gruber (1995), which states that an 'Ontology is an explicit specification of a conceptualization', meaning that ontology is a description of the concepts and relationships that exist in a domain (GAŠEVIĆ, DJURIC and DEVEDŽIC, 2009; FU, 2016; SHARMAN, KISHORE and RAMESH, 2007, SERNA and SERNA, 2014).

An ontology on a certain domain aims to capture, represent, share, (re)use and exchange the common understanding about the concepts in the domain, their taxonomies, classification, their relationships and the domain axioms (GAŠEVIĆ, DJURIC and DEVEDŽIC, 2009). Ontologies enhance knowledge sharing and reuse across different applications (NECHES et al., 1991). Ontology is also used to unify Databases, Data Warehouses, and knowledge bases vocabularies, in order to overcome the obstacles of knowledge integration, which basically consists on merging past and new knowledge (DJELLALI, 2013).

2.4.1. Elements of an Ontology

The formalization and implementation of an ontology vary according to the different knowledge representation and the corresponding language. However, they have some components in common (CORCHO, FERNÁNDEZ-LÓPEZ, GÓMEZ-PÉREZ, 2007):

- *Classes*: represent concepts that are used to represent objects and relations in a described domain. Usually organized in taxonomies through which inheritance mechanisms can be applied. Metaclasses can also be defined in the frame-based knowledge representation paradigm. In the metaclasses, the instances are classes. The metaclasses establish different layers of classes in the ontology where they are defined. Thus, they usually allow gradations of meaning.
- *Relations*: represent a type of association between the concepts on a domain. Relations in ontologies are usually binary. The first argument is known as the domain of the relation and the second is the range.
- *Formal axioms*: model sentences that are always true. Normally, used to represent knowledge that the other components can not formally define. They are useful to infer new knowledge, to verify the consistency of the ontology itself or the consistency of the knowledge stored in a knowledge base.
- *Instances*: represent elements or individuals in an ontology.

2.4.2. Classification of ontologies

There are several classifications of ontologies based on different parameters: degree of formality; level of specification, level of accuracy; level of generality; expressiveness, among others.

Gómez-Pérez, Fernández-López and Corcho, (1991) classify an ontology according to the level of specification of relationships among the terms gathered in the ontology:

- *Lightweight*: provides just a taxonomy of related terms and concepts, with very few cross-taxonomical links (properties), very few logical relations between the concepts, and very few axioms and constraints imposed on the concepts;
- *Heavyweight*: include a number of properties, axioms and constraints to lightweight ontologies. However, these ontologies are harder to manage.

Guarino, (1998) classifies an ontology according to their level of generality:

- *Top-level ontologies*: describe very general concepts like space, time, event, action, etc., and are independent of a particular problem or domain;
- *Domain and task ontologies*: specialize the terms introduced in the top-level ontology. They describe, respectively, the vocabulary related to a generic domain (like medicine, or automobiles) or a generic task or activity (like diagnosing or selling);
- *Application ontologies*: describe concepts depending both on a particular domain and task, which are often specializations of the related ontologies.

Van Heijst, Schreiber and Wielinga (1997) classify ontologies according to two dimensions: the amount and type of structure of the conceptualization and the subject of the conceptualization. With respect the first dimension, the authors identify three categories:

- *Terminological ontologies*: specify which terms are used to represent the knowledge in the domain of discourse;
- *Information ontologies*: specify storage structure of database. One example of this class of ontologies at the medical field is proposed by Rector et al., (1993). At this level, the model provides a framework for recording the basic observations of patients, but it makes no distinction between symptoms, signs, treatments, etc.
- *Knowledge modelling ontologies*: specify the conceptualization of the knowledge. This kind of ontology, usually have a richer internal structure. At this level, the observations are grouped to describe the decision-making process.

Regarding the second dimension, which is related to the subject of the conceptualization, four categories are distinguished:

- *Application ontologies*: contain all the definitions that are needed to model the knowledge required for a particular application. Usually, these ontologies take concepts from domain and generic ontologies and extend the knowledge by representing method- and task-specific components;
- *Domain ontologies*: express conceptualizations that are specific for particular domains;
- *Generic ontologies*: define concepts considered to be generic across many fields. Often, the concepts in the domain ontologies are defined as specializations of concepts in the generic ontologies;

- *Representation ontologies*: intend to be neutral with respect to world entities. They provide a representational framework without making claims about the world.

Table 1 summarizes the ontology classifications:

Table 1 - Classification of ontologies

Aspect	Proposed by	Types
Level of Specification	Gómez-Pérez, Fernández-López and Corcho, (1991)	Lightweight Heavyweight
Level of Generality	Guarino, (1998)	Top-level ontologies Domain and task ontologies Application ontologies
Level of generality	Fensel (2000)	Generic or common-sense ontologies Representational ontologies Domain ontologies Method and task ontologies

Source: Adapted from Tankelevičienė, 2008 and Hadzic et al., 2009.

An ontology classified as lightweight includes concepts, concept taxonomies, relationships between concepts and properties that describe concepts. On the other hand, when adding axioms and constraints to a lightweight ontology, the result is a heavyweight ontology (CORCHO, FERNÁNDEZ-LÓPEZ, and GÓMEZ-PÉREZ, 2003).

According Guarino (1998), very general concepts, such as time, space, object, among others, are described in top-level ontologies. Generic domain is described in domain ontologies and task ontologies describes generic tasks or activities. Finally, the concepts from a particular domain and task are described in application ontologies.

Regarding the level of generality, ontologies classified as generic or common-sense describes the general knowledge about the world (e.g., time, space, etc.). Representational ontologies describe representational entities without defining what should be represented. The knowledge related to a particular domain is described in a domain ontology. Method and tasks ontologies describes, respectively, terms specifics to PSM (Problem solving methods) and terms specific for particular tasks (FENSEL, 2000).

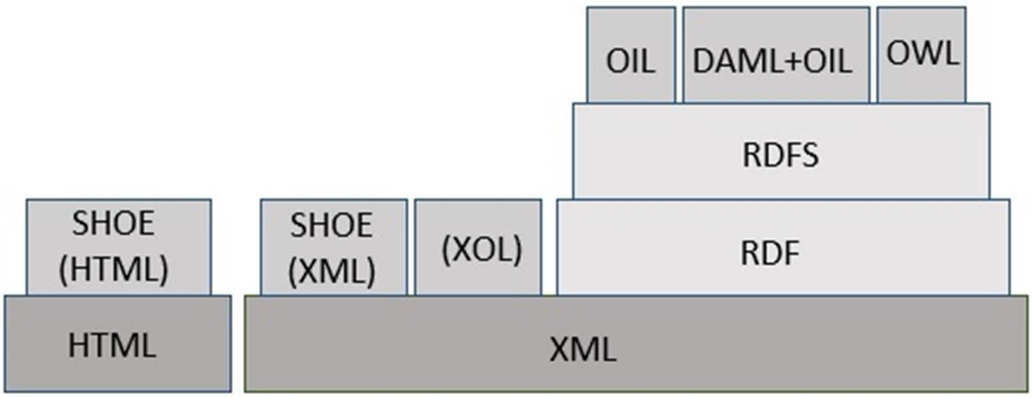
2.4.3. Ontology Languages

Ontologies are formalized by means of a logical language, describing the structure of the world that considers all objects involved within the domain of study, their possible states and all relevant relationships between them (SHARMAN, KISHORE and RAMESH, 2007).

Many languages have been proposed to build an ontology. Therefore, choosing a language is an important step, because different kinds of knowledge-based applications need different language features. Thus, the main point of choosing a language is based mainly on what the ontology will represent or be used for (TAYE, 2010).

Initially, the proposed ontology languages were built using AI modelling techniques based on first order logic, frames, and description logic. Then, for exploiting the characteristics of the Web, web-based ontology languages (or ontology mark-up languages) were developed. Their syntax is based on mark-up languages such as HTML and XML, whose purpose is the data presentation and data exchange respectively (CORCHO, 2010). The relationships among these languages are shown in Figure 9.

Figure 9 - The stack of ontology mark-up languages



Source: adapted from Corcho, Fernández-López and Gómez-Pérez, 2003.

The first mark-up language developed was SHOE (Simple HTML Ontology Extension). Combining frames and rules, SHOE was first developed as an HTML extension, with the aim of incorporating machine-readable semantic knowledge in HTML compliant or other WWW documents (CORCHO and GÓMEZ-PÉREZ, 2000). After the adoption of XML as a standard language for exchanging information on the WEB, SHOE syntax was modified to use XML and other ontology languages were built on the XML syntax (CORCHO, FERNÁNDEZ-LÓPEZ and GÓMEZ-PÉREZ, 2003).

XOL (XML-Based Ontology Exchange Language) was designed to be used as an intermediate language for transferring ontologies among different database systems, ontology-development tools, or application programs (KARP, CHAUDHRI and THOMERE, 1999).

RDF (Resource Description Framework) is the standard language for the creation of metadata describing Web resources. RDF is a foundation for processing metadata, it provides interoperability between applications that exchange machine-understandable information on the Web (W3C, 2004). However, RDF data model does not provide mechanisms for defining the relationships between properties (attributes) and resources, thus the Resource Description Framework Schema (RDFS) was developed (GÓMEZ-PÉREZ and CORCHO, 2002). RDFS provides the capabilities of vocabularies, taxonomies and ontologies. Its allow describing taxonomies of classes and properties. It defines the domain and range of the RDF classes and its properties (KHAN and KUMAR, 2014).

Based on the advent of RDF, a few more languages have been developed, including: Ontology Interchange Language (OIL), DARPA Agent Mark-up Language + OIL (DAML+OIL) and Web Ontology Language (OWL) (SONG, ZACHAREWICZ and CHEN, 2013).

OIL was proposed for describing and exchanging ontologies. Its syntax and semantics are based on existing proposals (OKBC, XOL, and RDF(S)) (GÓMEZ-PÉREZ and CORCHO, 2002). OIL has a precise semantics that forms a necessary foundation for effective reasoning support. However, this language does not enable to define the default-value, to provide the meta-class, and to support the concrete domain. Besides, the translation between OIL and RDF is no longer guaranteed (KALIBATIENE and VASILECAS, 2015).

DAML+OIL is an ontology language aiming to extend the syntactic interoperability to the semantic interoperability. It is specifically designed for use on the Web. DAML+OIL provides a set of constructs to create machine readable and understandable ontologies and to mark-up information (HORROCKS, 2002).

OWL is now a standard semantic web ontology language, recommended by W3C for the modelling ontologies. OWL is derived from the DAML and built upon the RDF (SONG, ZACHAREWICZ and CHEN, 2013).

Employing a rich set of operators, such as, intersection, union and negation, OWL is based on a logical model which allows the use of a reasoner. A reasoner can whether all the statements and definitions in the ontology are mutually consistent, and can also recognize which concepts fit under which definitions. OWL can be used to carry out logical inferences, derive knowledge and import and reuse other ontologies (BEIMEL and PELEG, 2011;

MENÁRGUEZ-TORTOSA and FERNÁNDEZ-BREIS, 2013; KALIBATIENE and VASILECAS, 2015).

OWL has three increasingly expressive sublanguages (McGUINESS and HARMELEN, 2004):

- *OWL Lite*: can be used to express taxonomy and simple constraints. For example, while it supports cardinality constraints, it only permits cardinality values of 0 or 1.
- *OWL DL (Description Logic)*: supports maximum expressiveness while retaining computational completeness (all conclusions are guaranteed to be computable) and decidability (all computations will finish in finite time). OWL DL includes all OWL language constructs, but they can be used only under certain restrictions (for example, while a class may be a subclass of many classes, a class cannot be an instance of another class).
- *OWL Full*: supports maximum expressiveness and the syntactic freedom of RDF with no computational guarantees. OWL Full allows an ontology to augment the meaning of the pre-defined (RDF or OWL) vocabulary. It is unlikely that any reasoning software will be able to support complete reasoning for every feature of OWL Full.

2.4.4. Techniques for reusing ontologies

After its development many ontologies have been created for the same domain by different experts with different points of view, using different tools and with different levels of details, granularity, completeness and their own focus. Thus, to enable the reuse of the ontologies some techniques were developed in order to overcome these differences:

- *Integration*: is required when building a new ontology by reusing other ontologies already available (CORCHO, FERNÁNDEZ-LÓPEZ and GÓMEZ-PÉREZ, 2007). The domain of the integrated ontology is different from the domain of the resulting ontology, but there may be a relation between both domains. When the integrated ontology is reused by the resulting ontology, the integrated concepts can be (1) used as they are, (2) adapted or modified, (3) specialized or (4) augmented by new concepts, among others (PINTO, GÓMEZ-PÉREZ and MARTINS, 1999).
- *Merging*: is related to building an ontology unifying knowledge of several ontologies into a single one. Thus, the subject of the merged ontologies is the same, although the level of generality may not be the same (PINTO, GÓMEZ-PÉREZ and MARTINS, 1999). Here, correspondences are stablished among the ontologies, and it must be

determined the set of overlapping concepts, concepts that are similar in meaning but have different names or structure and concepts that are unique to each of the sources (NOY and MUSEN, 2000).

- *Alignment*: establishes links between ontologies, however, the original ontologies are kept separately, i.e., they are not merged (CORCHO, FERNÁNDEZ-LÓPEZ and GÓMEZ-PÉREZ, 2007). Alignment usually is performed when the ontologies cover domains that are complementary to each other (NOY and MUSEN, 1999).

The ontology merging and alignment are supported by ontology mappings, which support several other operations such as translation, reconciliation, coordination, articulation, etc. Mapping could provide a common layer from which several ontologies could be accessed and hence could exchange information in semantically sound manners (KALFOGLOU and SCHORLEMMER, 2003). Thus, AMROUCH and MOSTEFAI (2012) define ontology mapping as formal expressions describing a semantic relationship between two (or more) concepts belonging to two (or more) different ontologies.

2.5. SYNTHESIS

This chapter addressed issues related with the process model customization, process mining and ontologies. The first section discussed mainly the approaches for dealing with business process variability. The literature shows that a process model can be classified by restricting the process model behaviour or by extending the process model behaviour. In the customization by restriction, the process model represents the process family in a single-model. On the other side, in the customization by extension, the process model represents the most common behaviour of the process family.

In both approaches, three aspects are essential to customize a process model: the variation points, the alternatives available for the variation point and the rules for choosing the alternatives. These aspects can be identified through process mining technique, which is addressed in Section 2.2. Process mining is applied to discover, monitor and improve the process behaviour. The heuristic miner can be applied to discover the decision points, i.e., the variation points. However, this technique cannot provide any knowledge about the rules for select the alternatives available for the variation points. Thus, the decision miner can be applied for discover these rules. The decision miner also enables to understand the dependencies between the variation points.

Section 3 discusses some aspects related to ontology, such its elements, classifications and the languages for building them. The classification of an ontology relies on different aspects such as expressiveness, generality, formality, among others. The selection of a language for building an ontology relies on the purpose of the ontology. Finally, some activities to enable the reuse of ontologies are discussed.

3. RELATED WORKS

The previous chapter discussed the process model customization focusing on the approaches and the aspects that need to be considered to obtain a process variant that correctly represent a business context. The chapter also addressed the challenges for customizing a process model. Process mining techniques and ontologies have also been addressed in the previous chapter. The study of these topics demonstrate which process mining techniques can be applied to build a customizable process model and that ontologies can be applied for decision-making support by providing recommendations during the process model customization.

Based on the finds provided by the previous chapter, it is necessary to understand how process model customization has been addressed in relation with the aspects and challenges related with process model customization, which enable to identify the existing drawbacks. Thus, this chapter discusses some approaches for process model customization. In addition, this chapter also analyses how process mining and ontologies have been applied in relation with the customization of process models.

Thus, the first section delimits the methods for process model customization analysed in this research. An illustrative process model (Section 3.2) is presented to demonstrate the applicability of each approach. Then, each method for process model customization is presented (Section 3.3). Some methods provide decision support during process model customization by applying different techniques. Thus, section 3.4 discussed these techniques. The relationships between process mining techniques and customizable process model are discussed in Section 3.5 and Section 3.6 shows how ontologies have been applied in customizable process models. Finally, a discussion about the discovered gaps and the contributions of this research is presented in Section 3.7.

3.1. PROCESS MODEL CUSTOMIZATION

Several approaches have been developed for process model customization. These approaches customize process model in different ways. These differences are mainly related to the configuration mechanisms applied to derive a process variant, which can be classified as (LA ROSA et al., 2017; REICHERT and WEBER, 2012):

- *Node Configuration*: are points in the process subject to variation, in which options are assigned. Thus, the process model is customized by selecting one option per configurable node. A configurable node can be defined as activities, events, gateways, resources and objects associated with activities. A configurable activity, event, resource and object can be customized by keeping the element as ON (the element remains in the customized process model) or they are switched OFF (the element is removed or hidden, thus not appearing in the customized process model). The customization of a configurable node can be realized to an equal or more restrictive gateway, in such a way that the customized process model produces the same or fewer execution traces than the customizable process model.

- *Element Annotation*: graphically annotate a model element (activities, events, gateways, sequence flows, resources and objects) with properties of the application domain. The annotate element is a variation point. Domain conditions assign the domain properties to model elements. Thus, customization is performed by selecting domain properties. When a domain condition is set as false, the related model element is removed from the process model.
- *Activity specialization*: the variation points are activities defined as abstract and optional. The process model customization is performed by selecting one or more variants that have been assigned to the activities defined as variation points. An optional activity is a variation point that can be specialized to an empty activity, i.e., it can be switched off. Variants can also be assigned to activity attributes such as objects and resources, which become variation points.
- *Fragment customization:* the process model is customized through change operations, which enable to delete, insert, move or modify a process fragment. For customizing a process model, the base process model contains adjustment points, which serve as stable reference points for prespecified changes.

The configuration mechanisms presented previously, define the variation points as different elements of the process model such as gateways and activities. The use of a configuration mechanism defines the type of customization, how the process model customization is performed and how it is presented for the user. For example, the configurable node shows to the user all choices existent for each variation point and, the customization is performed by restricting the process model behaviour. The fragment customization shows the options for each variation point separately and, this mechanism enables to customize the process model by extending or restricting the process model behaviour.

Due the characteristics of the configurable node mechanism (also called variation point), this chapter focuses on approaches that customize the process model through this mechanism. Configurable nodes are the points of the business process model where a process fragmented is selected, i.e., they are the points of the process model that are subject to variation. Business rules related to these points define the selection of an option instead another.

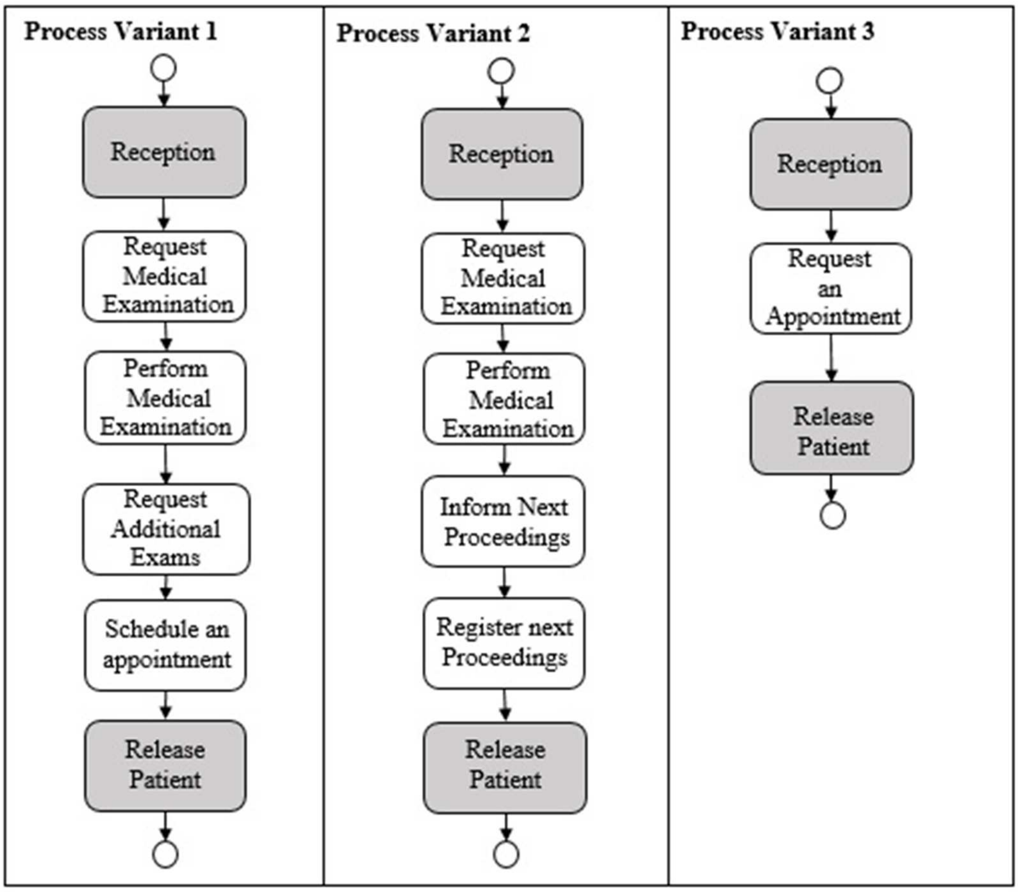
The approach envisaged in this research aims at customizing the process model through configurable nodes. Thus, the aim here is to identify the main characteristics of these approaches and the gaps that may exist. The approaches are analysed considering the following aspects: structural and behavioural correctness of the resulting process model, guidance and recommendations during customization, the relationships between the variation points, which enable to evaluate the impact of the configuration decisions on the model. To discuss the approaches, an illustrative process model is presented.

3.2. ILLUSTRATIVE PROCESS FOR HANDLING MEDICAL EXAMINATIONS

Let us consider the process variants for handling medical examinations depicted in Figure 10. The process variants contain two common activities in grey-shaded (e.g., Reception and Release Patient). However, some activities may or may not be performed according to some rules, which means that these activities are subjected to variation.

When the patient has an appointment, the medical examination is requested. Otherwise, an appointment is requested for another day. If a patient has an appointment, the medical examination is performed. The physician may request an additional exam for the patient, in which case a new appointment need to be scheduled. Otherwise, the patient is informed about the next proceedings (which can be related to the end of the treatment or information about the next steps). In this case, the proceeding must be registered.

Figure 10 - Process variants for handling medical examinations



3.3. CONFIGURABLE NODES APPROACHES

This section discusses the approaches for customizing a process model by applying rules into the configurable nodes. The approaches have been analysed regarding five criteria: structural and behavioural correctness, guidance, recommendations, and the relation between variation points. The results were evaluate considering to what extent the approach in question covers each evaluation criterion. Thus, a "+" indicate a criterion that is fulfilled, a "-" indicate a criterion that is not fulfilled and a "+/-" to indicate partial fulfilment. points. The results are presented at the end of this chapter.

3.3.1. Configurable Event-driven Process Chains (C-EPC)

Configurable EPC (ROSEMANN and VAN DER AALST, 2007) is an extension of EPC, aiming at capturing the variations in the process model. It consists of events, activities and connectors, such that connectors and activities can be configurable. To allow customization, variation points are identified and configuration requirements are assigned to them to restrict the model behaviour.

Activities can be included (ON), excluded (OFF) or conditionally skipped (OPT). Regarding the first two alternatives, the decision about keeping or discarding the activity in the resulting process variant may be made at configuration time. The last alternative allows deferring this decision to the run-time.

Three types of connectors (AND, Exclusive OR, OR) can be used to model splits and joins. The configurable connectors may be restricted at built-time. Connectors may only be configured to a connector being equally restrictive. This means that the derived process model should have the same or less execution traces than the original model (LA ROSA, 2009). Table 2 summarizes the configuration alternatives of configurable connectors.

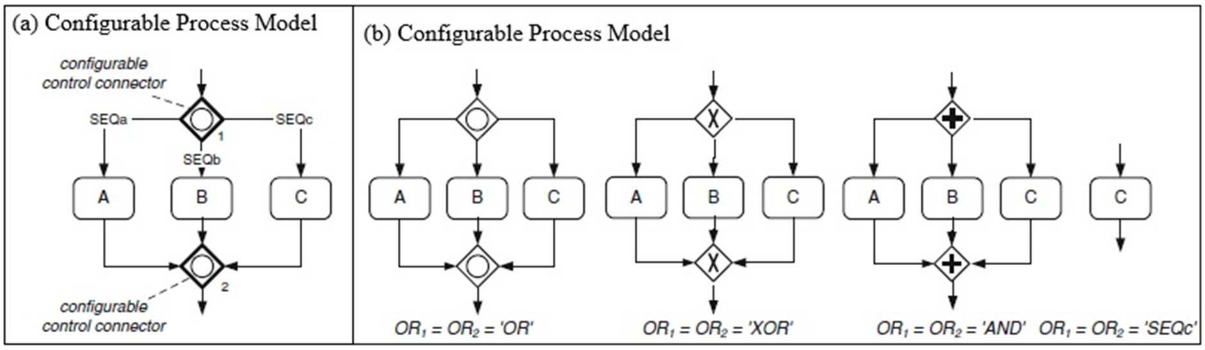
Table 2 - Constraints for the configuration of connectors

Configurable Connector \ **Connector Type**	**OR**	**XOR**	**AND**	**SEQ**
Configurable OR	X	X	X	X
Configurable XOR		X		X
Configurable AND			X	

Source: Rosemann and van der Aalst (2007).

According to Table 2, a configurable connector OR can be configured (i.e., its behaviour can be changed) to all types of connectors: OR, XOR, AND, SEQ. The configurable connector XOR is configured to a XOR or a SEQ connector. A SEQ connector is an outgoing or incoming branch. A configurable connector AND is only configured to another AND or a OR connector, meaning that no particular configuration is available (ROSEMANN and VAN DER AALST, 2007). Figure 11 depicts an example of the configuration alternatives of a configurable OR connector.

Figure 11 - Configuration alternatives of a Configurable OR connector



Source: Reichert and Weber, 2012.

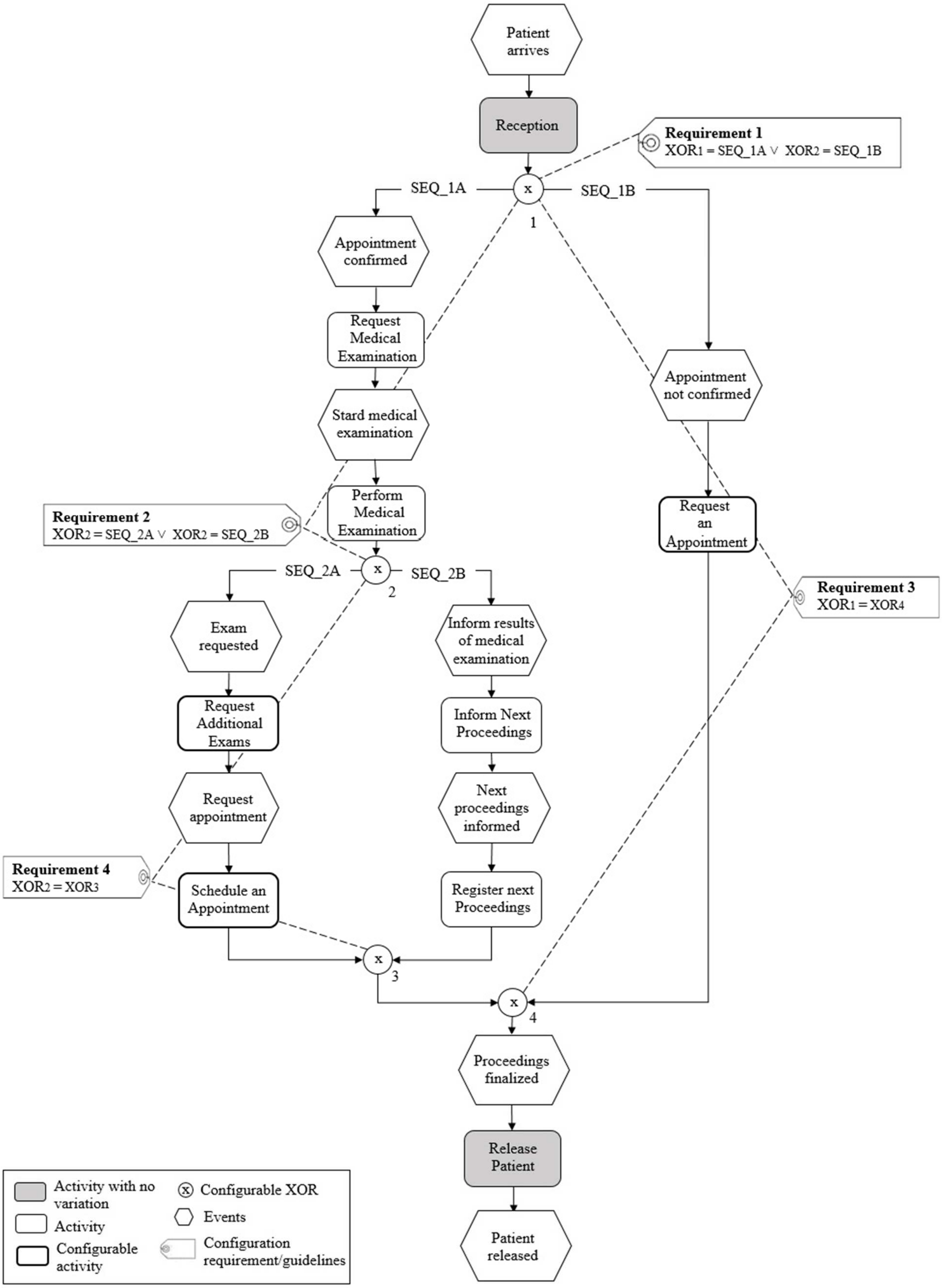
Figure 11(a) shows two configurable connectors OR, which according to the configuration choices can be configured to four process variants (Figure 11(b)). Configuration requirements define constraints that are expressed in the shape of the configurable connectors. Those connectors will define local configuration choices. Guidelines can also be enclosed to the configurable nodes, however while the requirements are mandatory, the guidelines only provide recommendations for the process customization. Both, configuration requirements and guidelines are expressed through the logical predicates.

The Figure 12 depicts an example of the EPC model for the illustrative process for handling medical examination. Configurable connectors are denoted as thick circles. Configurable activities are denoted as thick rectangles. Configuration requirements and guidelines are denoted by dotted lines connecting the configurable nodes through the logical expression. The EPC model has four requirements.

- In the first requirement, a choice need to be made between the activities ‘Request an Appointment’ and ‘Request Medical Examination’. If the activity ‘Request an Appointment’ is set as ON, the activities related with the SEQ_1A are set as OFF, thus an appointment is scheduled and the patient is released. Otherwise, SEQ_1A is set as ON and the activity ‘Request Medical Requirement’ is selected.
- In Requirement 2, a choice need to be made between the activities ‘Request Additional Exams’ and ‘Inform Next Proceedings’. If the activity ‘Request Additional Exams’ is set as ON, the activity ‘Schedule an Appointment’ is also set as ON. However, if the activity ‘Inform Next Proceedings’ is set as ON, then the activity ‘Register Next Proceedings’ is also set as ON.

After assigning each configurable node with a variant, an algorithm is applied to derive an EPC from the configured EPC. The algorithm removes arcs not involved in the selected sequence. Also, all nodes without input and output arcs are removed, thus ensuring the structural correctness of the process variant. The behavioural correctness of the process variants is guarantee by the requirements attached with each variation point. C-EPC model is not executable. Besides, the guidelines provided are related with the behaviour of the variation points, not in terms of business choices. It is not easy to identify the relationships among all the variation points, making difficult to evaluate the impact of the configuration decisions on the model.

Figure 12 - EPC model for the illustrative process for handling medical examination

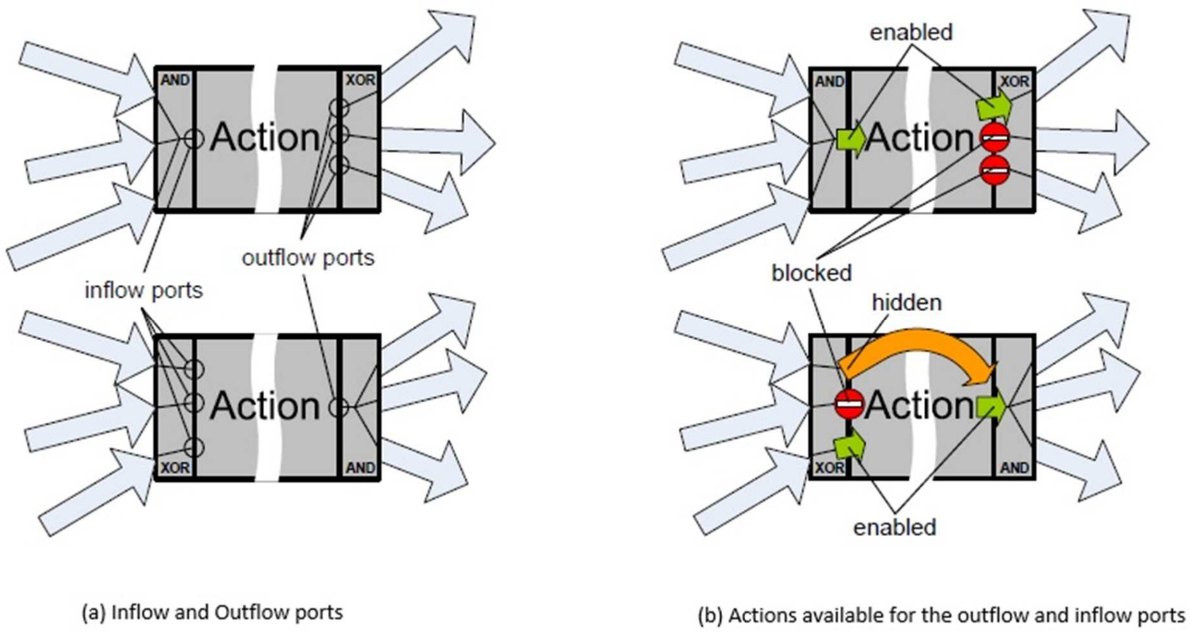


3.3.2. Configurable YAWL (C-YAWL)

Configurable workflows (VAN DER AALST et al., 2006, GOTTSCHALK et al.2008) aim at customizing workflows by enabling or disabling actions in such models. It focuses on executable business process models, although configurable workflows can also be applicable to non-executable modelling languages as well. To apply the configurable workflow approach, the Configurable YAWL (C-YAWL) was developed.

Triggering an activity enables its execution. Typically, triggers are represented by arcs pointing into an activity. These arcs can have different meanings due the different joining patterns (AND-join and XOR-join) for the preceding paths leading into the activity. The combination of incoming paths through which an activity can be triggered is called an inflow port. When the action is completed, it releases the case via the arcs leaving the action through one distinct outflow port, which triggers all paths connected to this outflow port. Figure 13 shows an example of an inflow and outflow ports.

Figure 13 - Ports in a C-YAWL



Source: Gottschalk et al., 2008.

Ports are the configurable elements in C-YAWL. Every port can be enabled or blocked, while inflow ports can also be hidden. If an inflow port is enabled, it allows the triggering of the action through this port. In a blocked inflow port, no case can flow into the action through this port. If an action is triggered via a hidden inflow port, the action itself is skipped and the case is directly forwarded to one of the outflow ports (usually but not necessarily a default output port) (GOTTSCHALK et al., 2008).

An outflow port can be enabled or blocked. If an outflow port is enabled, it can be selected to release the case. A blocked outflow port cannot be selected as the used outflow port. However, at least one outflow port must always be enabled, thus allowing the cases to leave a triggered action. Figure 14(a) depicts an example of the C-YAWL.

The first activity ('Reception') of the process model is used to route the process flow. This task has only one incoming arc from the input condition. Therefore, its join has only one input port which always needs to be enabled. The activity's XOR-split has two output ports: one to trigger the path to condition 0a, which leads to activity 'Request an Appointment', and one that trigger the path to condition 0b, leading to activity 'Request Medical Examination'. The XOR-join of the activity 'Reception' is the activity 'Release Patient', which is always enabled because it is triggered by choosing both conditions (0a and 0b).

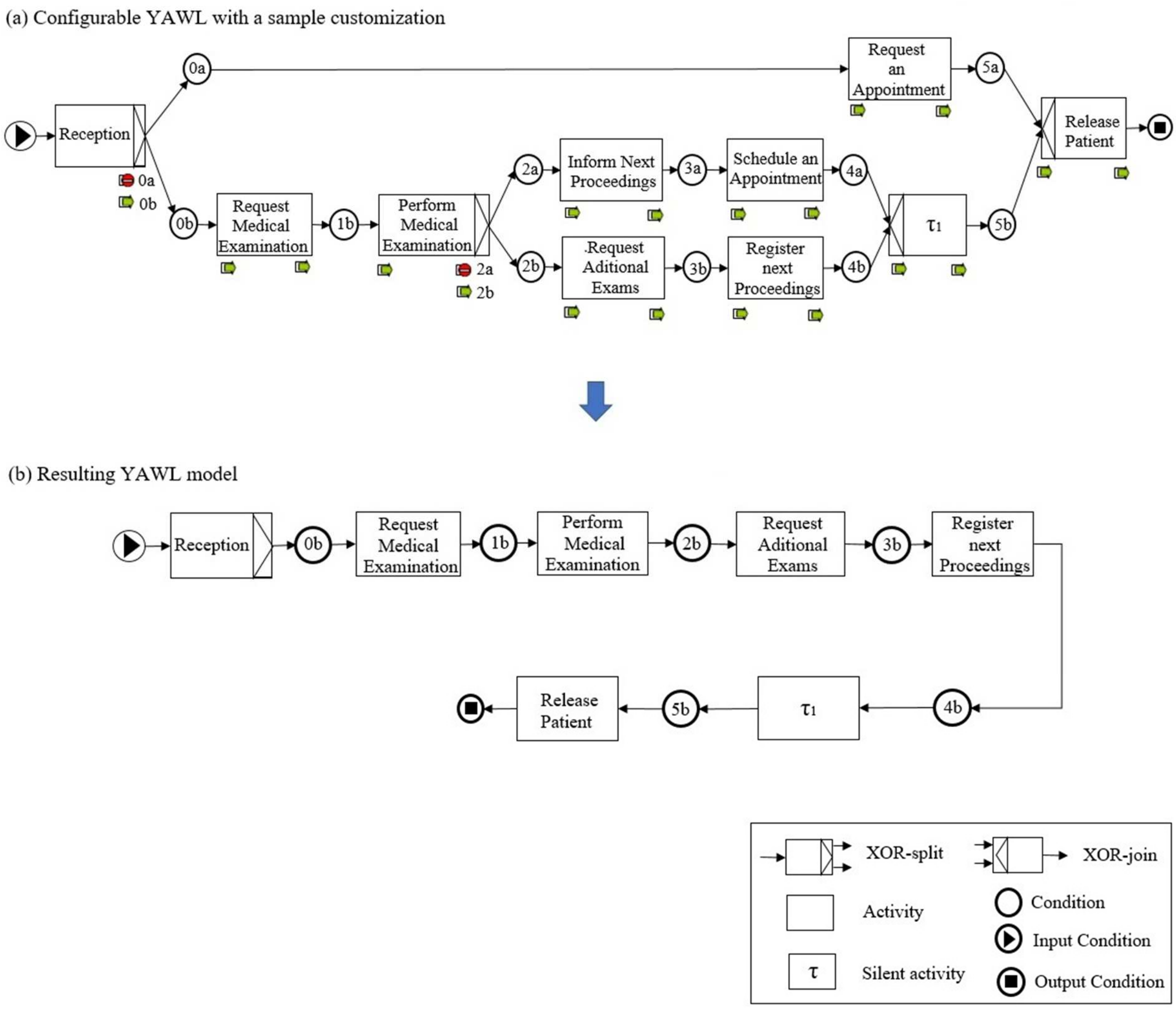
As the port 0b is enabled, this path is triggered. In this path, another activity ('Perform Medical Examination') is used as XOR-split with two output ports: one to trigger the path to condition 2a, and one to trigger the path to condition 2b. The OR-join (τ1), which is a silent activity, is enabled as these tasks are always performed when path 0b is triggered. The silent activity does not include any "action", however it has the same behaviour as the original task.

The process model in C-YAWL represents all the possible behaviours of the application context. Thus, the process model is restricted by hiding or blocking activities through the ports. To obtain the process variant, the blocked elements and their dead successors must be removed from the model and the hidden elements must be replaced by shortcuts.

Based on the configuration of a port, C-YAWL allows to define configuration requirements to restrict the values of other ports. These requirements are expressed as Boolean conditions over the ports configuration. For example, the expression (output), (Perform Medical Examination), {3a}, (Enabled) ⇨ (output), (Reception), {0b}, (Enabled), binds the outgoing ports of the activities 'Perform Medical Examination' and 'Reception' ensuring the execution of the activity 'Inform Next Proceedings'.

The hiding and blocking operators can also be applied to the configuration of elements specific to the YAWL language, such as cancellation regions and composite tasks. Gottschalk et al. (2008) also defined an algorithm for customizing the C-YAWL. The nodes that are blocked or hidden are removed from the input to the output condition, thus ensuring the structural correctness of the model. Figure 14(b) depicts the resulting YAWL model.

Figure 14 - Configurable YAWL for the illustrative process for handling medical examination



To ensure behavioural correctness of the customized models, two alternatives are available, one based on constraints inference and the other on partner synthesis. The authors also implemented the C-YAWL in the YAWL Editor, which allows one to create, customize and transform C-YAWL models into YAWL models. The questionnaire-model approach (Section 3.4.1) can be applied for decision support. The Synergia toolset supports the customization via questionnaire models. The rules to choose a path are not obviously, besides no recommendation about the context of application is provided to the user.

3.3.3. Configurable Integrated EPC (C-iEPC)

Configurable iEPC (LA ROSA et al., 2008; LA ROSA et al., 2011) extended the C-EPC to include the representation of roles and objects in the process model. A role, which can be human or not human, aims to capture a class of organizational resources that is able to perform that task. An object captures an information artefact or a physical artefact of an enterprise that is used or produced by a function.

As in the C-EPC, the main elements in the C-iEPCs are activities, control-flow connectors, and arcs linking these elements. Configurable activities and connectors are represented by thicker border. Besides, configurable gateways can be customized to an equal or more restrictive gateway. Roles and objects are associated directly with activities or through a connector, which specifies a logical conditional for a set of roles or objects. Figure 15 depicts an example of the C-iEPC model.

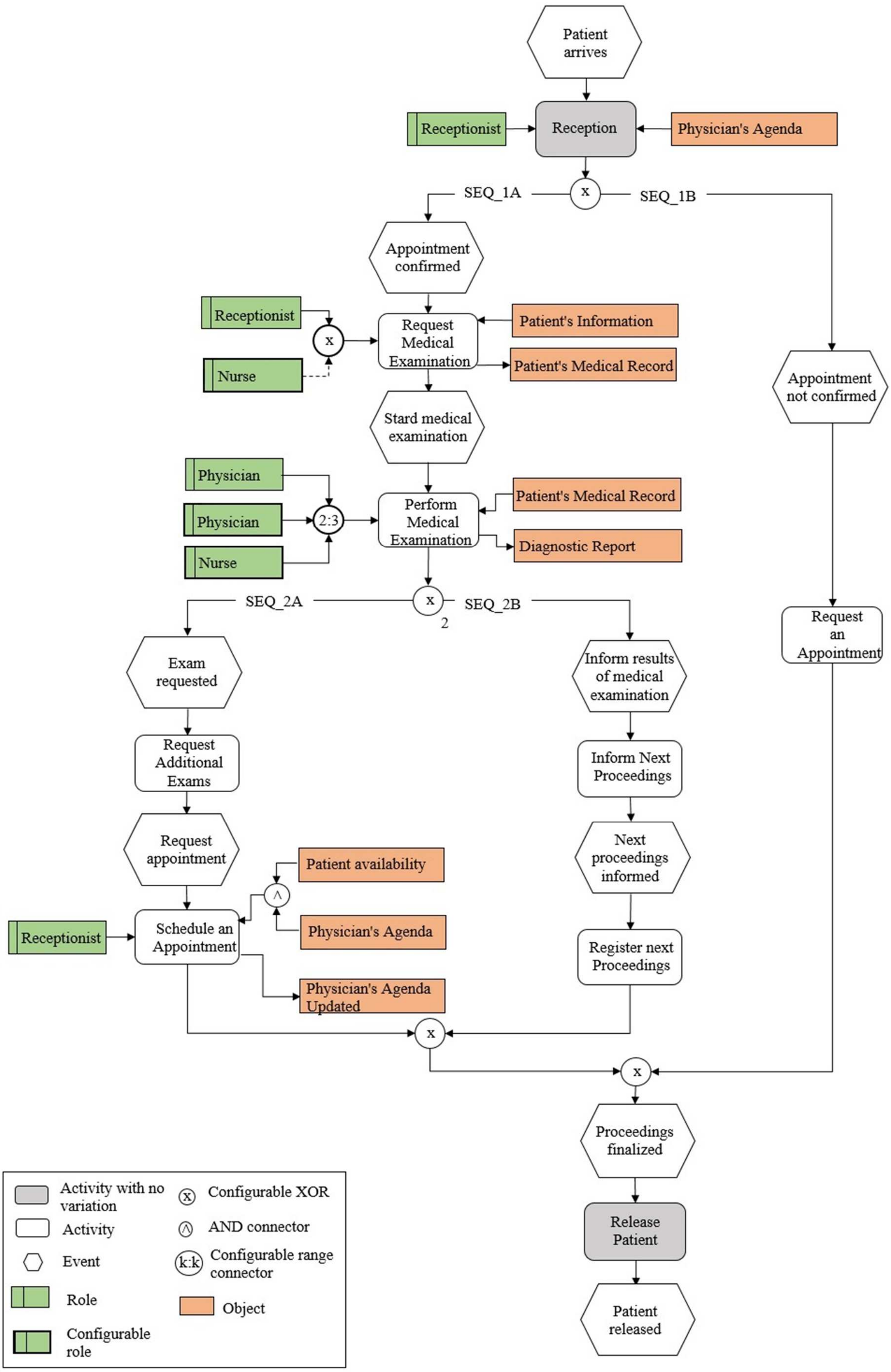
The process for handling medical examination contains three roles: physician, receptionist, and nurse. The receptionist is responsible for schedule appointments and register the next proceedings. The activity 'Request medical examination' can be performed by the nurse or the receptionist. The physician performs the medical examination, request additional exams and inform the patient about the next proceedings. A nurse can help the physician in the execution of these activities.

For simplicity, not all roles and objects are shown in the model. According to Figure 15, roles are shown in the left side of the function, while objects are shown in the right side of the function. In the example, each role is human. However, it also can be a machine or a software system. The activity 'Request Medical Examination' is performed by the receptionist or by the nurse and the activity uses the patient's information to obtain the patient's medical record. Each object in the process model is statically bound to a concrete artefact. Therefore, if two objects in a model have the same label, they are treated as being the same artefact.

The activity 'Perform Medical Examination' can be executed by the physician, the nurse, and the medical resident. These roles are linked together by a range connector. The range connector (k:2) means that the activity must be performed by at least two of the roles. Thus, the connector indicates the lower bound and upper bound for the number of elements (roles or objects) that are required. Range gateways can be used with the three logical types of OR, XOR, and AND, and it allow any combination of the associated objects or roles. They can be optional, which means that all connected elements are also optional.

Thus, roles and objects can be optional (a dashed arc) or mandatory (full arc). However, to ensure the activity's execution at least one mandatory role must be assigned to the activity. According to Figure 15 the role 'Nurse' associated with the activity 'Request Medical Examination' is optional and the role 'Receptionist' is mandatory.

Figure 15 - C-iEPC model for handling medical examinations



Patient arrives
Receptionist
Reception
Physician's Agenda
SEQ_1A
x
SEQ_1B
Appointment confirmed
Receptionist
x
Nurse
Request Medical Examination
Patient's Information
Patient's Medical Record
Appointment not confirmed
Stard medical examination
Physician
Physician
2:3
Nurse
Perform Medical Examination
Patient's Medical Record
Diagnostic Report
SEQ_2A
x
2
SEQ_2B
Request an Appointment
Exam requested
Inform results of medical examination
Request Additional Exams
Inform Next Proceedings
Request appointment
Next proceedings informed
Patient availability
Λ
Receptionist
Schedule an Appointment
Physician's Agenda
Register next Proceedings
Physician's Agenda Updated
x
x
Proceedings finalized
Release Patient
Patient released
Activity with no variation
Activity
Event
Role
Configurable role
Configurable XOR
AND connector
Configurable range connector
Object

An C-iEPC model can have configurable activities, connectors, roles, and objects. Configurable activities can be kept on, switched off or optional. If an activity is switched OFF, it is hidden in the customized model. An optional activity can wait until run-time to be customized. If a resource, object or range gateway is optional, it can be customized to mandatory so that is kept in the customized model, or switched off. If it is mandatory, it can only be switched off. Further, resources and objects can be specialized to a sub-type according to a hierarchy model which complements the C-iEPC model. Configurable input objects that are consumed can be restricted to use, so that they are not destroyed by the activity after use (LA ROSA et al., 2017).

La Rosa et al. (2011) developed an algorithm to guarantee that if the C-iEPC is structurally correct, the customized iEPC is also correct. For the customization, all nodes that are no longer connected to the initial and final events via a path are removed, and the remaining nodes are reconnected. Behavioural correctness is ensured via constraints inference.

C-iEPCs do not provide any execution support. Abstraction and guidance during customization are achieved by means of a questionnaire linked to the configurable nodes of a C-iEPC. The model also does not provide any recommendations for the user and the relationships between variation points are not obvious.

3.3.4. Application-Based Domain Modelling (ADOM)

ADOM-EPC was developed aiming to increase the level of adaptability of EPC models (REINHARTZ-BERGER, SOFFER and STURM, 2010). ADOM has also been applied on UML Activity Diagrams (REINHARTZ-BERGER, SOFFER and STURM, 2008) and BPMN (REINHARTZ-BERGER, SOFFER and STURM, 2009).

In ADOM two types of classifiers, called multiplicity indicators and reference model classifiers are added to all EPC elements (activities, connectors, events, and arcs). Multiplicity indicators, denoted as <min, max>, are attached with model elements. They define the lowest and uppermost numbers of variants that these elements may have in a business process model. The multiplicity indicators capture commonalities and the variability that may exist in the process model.

Common elements are considered mandatory with a multiplicity indicator <1, n>. Optional elements have a multiplicity indicator <0,1>; an activity with a multiplicity indicator <1,1> must be instantiated exactly once, i.e. it is kept as is in the customized model. The default multiplicity, denoted as <0, n> implies no constraints (LA ROSA et al., 2017).

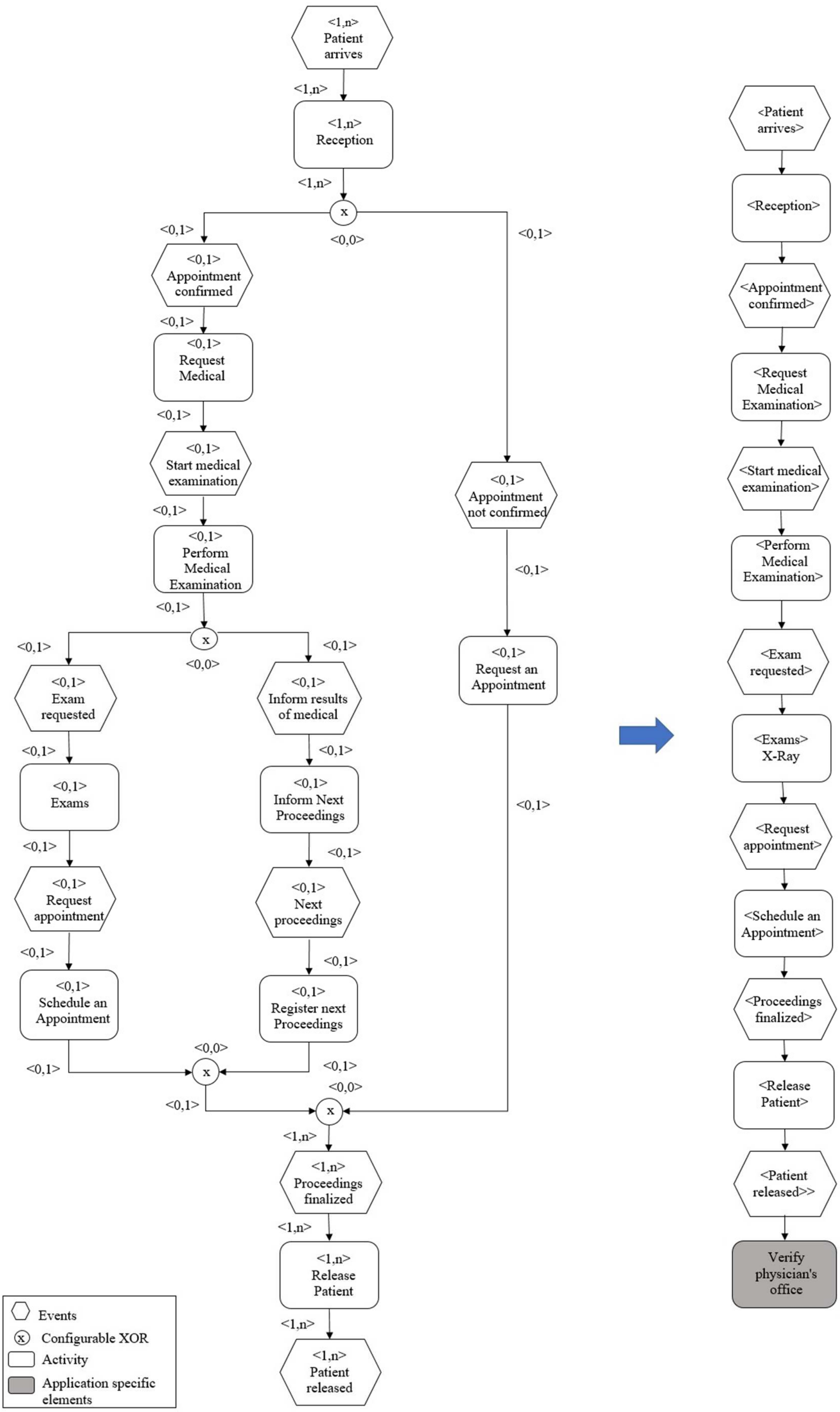
Gateways with cardinality <0,0> are removed from the customized process model. Gateways with cardinality <0,1> are optional. As the C-EPC and C-iEPC, an OR gateway can be restricted to become an AND or XOR. Cardinality can also be assigned to sequence flows, although the customization of an arc need to respect the customization of the configurable nodes connected to the respective arc. For example, an arc with cardinality <0,1> connected with configurable nodes with cardinality <1,1> may derive a process model with two configurable nodes without a sequence flow between them.

The second type of classifier is the reference model classifiers, which represent associations between model elements of the process model with a process variant. The model elements that are related with a process variant are denoted by < > near the specific model element names. Model elements assigned with < > can be specialized, providing more information about the specific situation. For example, the activity ‘Request Additional Exams’ can be specialized according to the type of the exam requested as shown on Figure 16.

Common activities (‘Reception’ and Release Patient’) have a cardinality <1, n>. The two gateways have a cardinality <0, 0>, which means that they can be removed from the customized process model, as well the other optional activities and sequence flows (cardinality <0,1>). ADOM also enables to add model elements (called application-specific elements) that are not present in the customizable process model. In Figure 16, the activity in grey-shaded is added to the customized process model.

The alternation between events and activities are ensured by specific rules that have been defined to bind the customization of an event to that of an activity, though disconnected nodes cannot be avoided (LA ROSA et al., 2017). The approach does not guarantee the behavioural correctness of the customized models. Guidance is not provided. Besides, the ADOM-EPC model has a higher level of abstraction which can raise ambiguities. Also, it is not allowed to constraint the behaviour of the resulting process variant.

Figure 16 - ADOM for handling medical examination



3.4. CONFIGURATION MECHANISMS FOR CUSTOMIZING PROCESS MODEL

As previously mentioned, an approach for process model customization can also be classified as Element Annotation, Activity specialization and Fragment customization. Therefore, this section presents the most relevant approach related to each classification.

3.4.1. Element annotation

Approaches classified as element annotation, customize the process model by annotating a model element with properties of the application domain. In these approaches, the annotate element is the variation point. Thus, by selecting the domain properties the process model is customized. The Configurative Process Modelling is one of the most relevant approaches in this group (LA ROSA et al., 2017).

Configurative Process Modelling (BECKER et al., 2004, BECKER, DELFMANN and KNACKSTEDT, 2007) refers to the use of integrated information models containing all relevant and specific variations from a domain for customize a process model. This approach is based on the idea of meta model projection, i.e., a projection of the process model is created for a specific scenario by fading out the undesired branches. If all models are instances of a formalized meta model, the integrated information models can be automatically transformed into perspective-specific models.

Each application context is represented through adaptations in the process model. There are two types of adaptations: business characteristics and values and perspectives. Business characteristics and values refers to a set of domain properties used to determine the available contexts and drive the customization (LA ROSA et al., 2017). Perspectives represent the requirements of different users applying the information model.

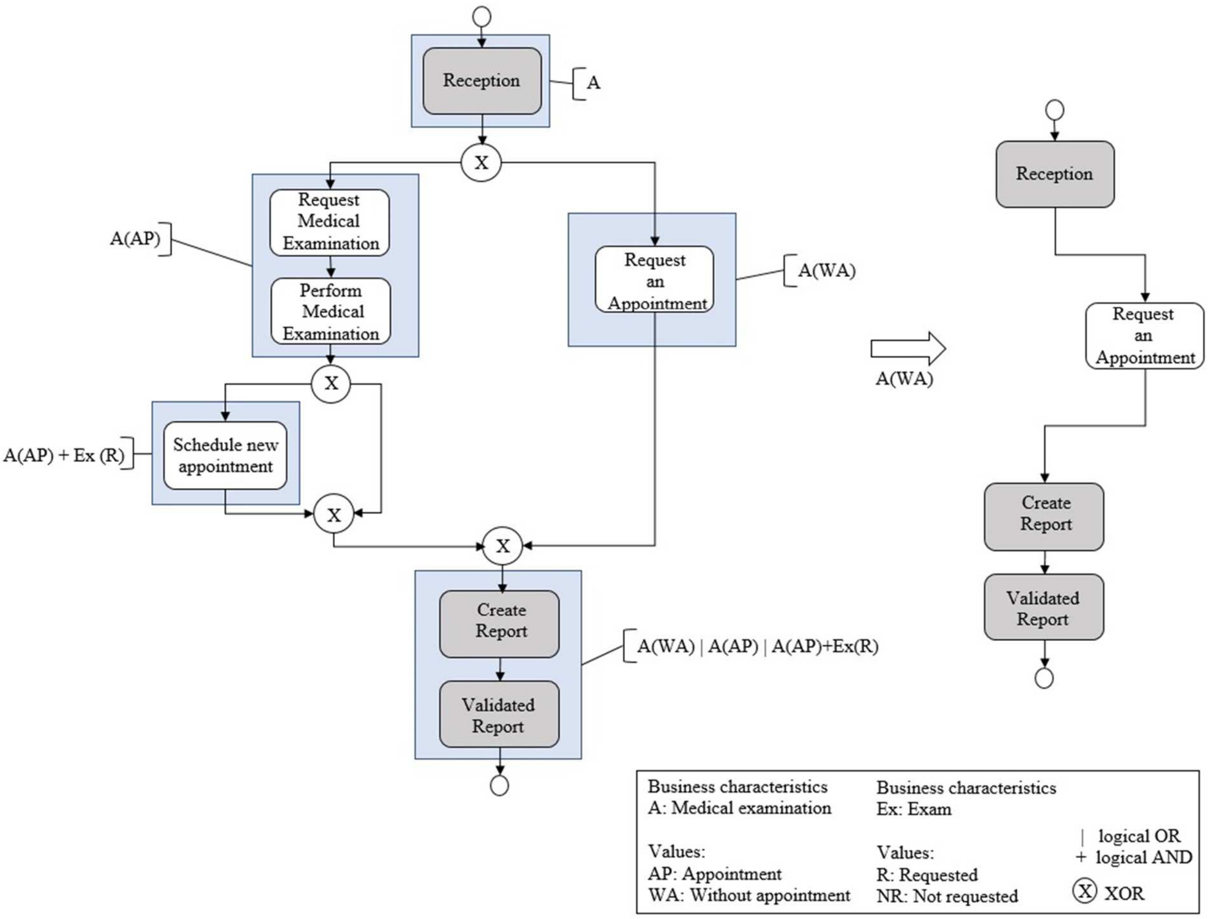
To specify the adaptations concepts, three meta-models are used: model layer, meta-model layer and meta-meta-model layer, which are based on EPCs (Event-Driven Process Chain). EPC consists of interrelated instances of the following process object types: function, event, connector, and several resource types (e.g. document, employee, application).

Both types of adaptations are linked with the process elements by means of configuration parameters defined in the form of simple attributes or logical terms over characteristics (LA ROSA et al., 2017). Regarding the business characteristics, if a parameter evaluates as false, the element is marked as hidden. In this way, the projection of the process model is obtained by removing the hidden elements and reconnecting the remaining nodes.

Configuration parameters also can be applied to remove process modelling perspectives that are not relevant to a specific scenario.

Considering the illustrative process model for handling medical examinations, two business characteristics are identified as shown in Figure 17: the ‘Medical examination’ with values ‘Appointment’ and ‘Without appointment’, and the business characteristic ‘Perform exam’ with values ‘Requested’ and ‘Not requested’. The configuration parameters are defined and linked with the respective elements. For example, configuring the parameter A(WA) as true and the others as false, the process variant is obtained by removing the undesired elements.

Figure 17 – Configurative Process Modelling approach for handling medical examinations



As limitation, this method applies an algorithm to support the individualization, which can fix simple syntactic issues, but cannot ensure that the resulting model is correct in a behavioural and structural way. Besides, the approach suffers a lack of expressiveness since the routing behaviour cannot be restricted (LA ROSA et al., 2017).

No guidance is provided during the evaluation of the configuration parameters. Besides, it is not possible to identify activity subject to variation from the ones that are common to all variants. Also, the approach does not allow to identify the relationships between the variation points.

3.4.2. Activity Specialization

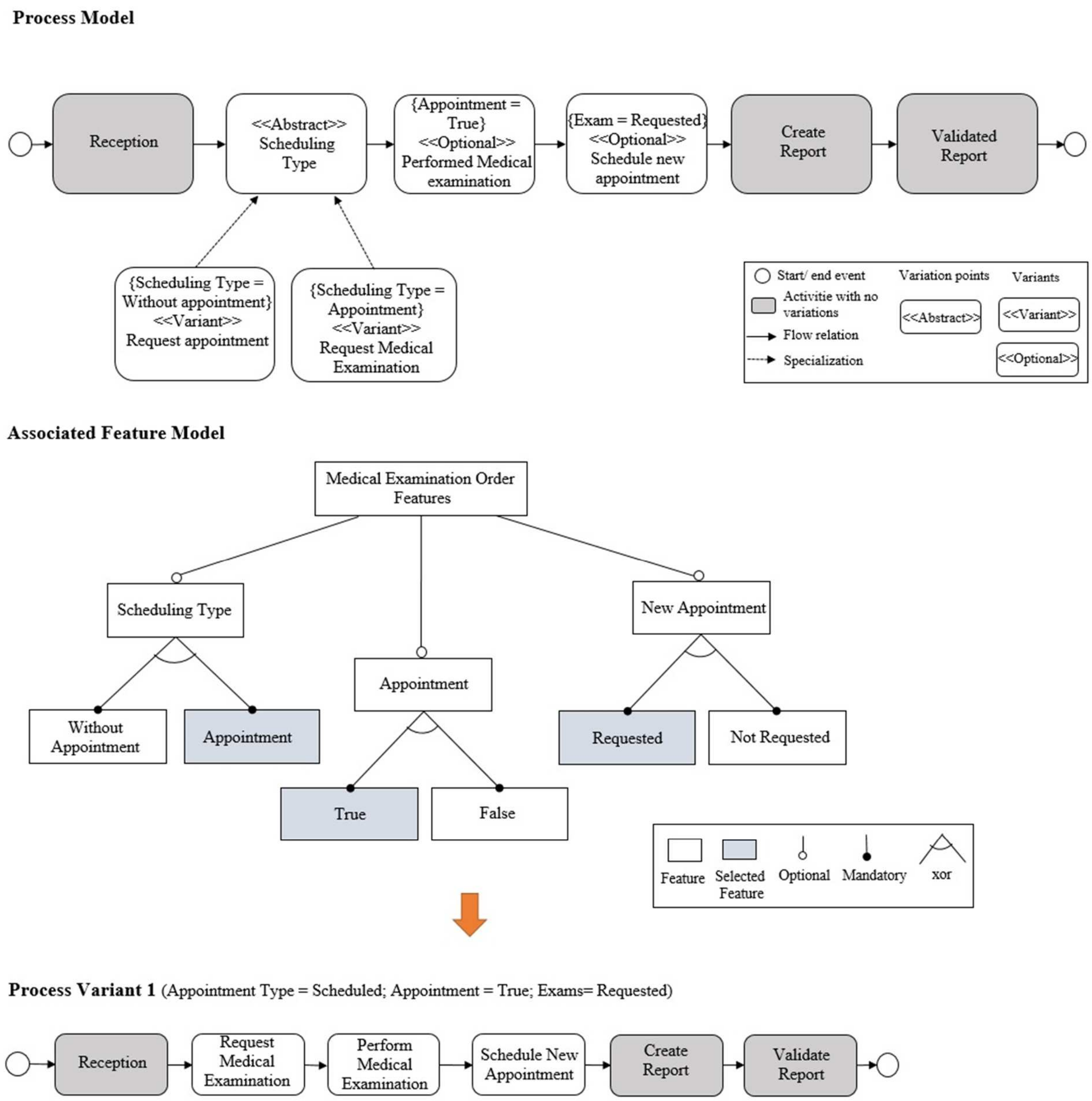
Approaches classified as activity specialization customize the process model by defining some activities as variation points. In this group, one of the most relevant approaches is the Process Family Engineering in Service-Oriented Applications (PESOA). Developed by Puhlmann et al. (2005), this approach aiming the development and the customization of families of process-oriented software. In this approach, activities that are subject to variation are marked with stereotypes. Variants are represented by the stereotype <<Variant>> and variation points are represented by the stereotype called <<VarPoint>>, which can be further specialized.

Feature diagrams (discussed in section 3.5.2) are used to configure the process variant. Thus, a mapping is stablished linking the process variants with the respective features. In this way, when a feature is disabled, the corresponding variant is removed from the process model. Feature diagram also enable defining domain constrains to restrict the possible combinations of process variants.

Figure 18 depicts an example of the PESOA approach related to the process model for handling medical examinations. The variation point ‘Scheduling Type’ is assigned with the stereotype <<Abstract>> since only one activity can be assigned to the variation point (i.e., Request Appointment or Request Medical Examination). Two activities are tagged as <<Optional>> which are modelled as extension point (e.g., activity Schedule New Appointment is only available if additional exams have been requested). Process variants can be derived by selecting features in the associated features model. For example, through the highlighted features, one process variant can be obtained.

PESOA approach applies customization by restriction, i.e., process variants are obtained by removing the undesired variants to obtain a specific process variant. The control flow is partially provided by means of the feature diagram. The approach does not support the structural or behavioural correctness of the related process variants. It also does not consider the relationships that might exist between the different variation points and it is not possible to distinguish whether a variation point is resolved at design or enactment time.

Figure 18 - PESOA approach for handling medical examinations



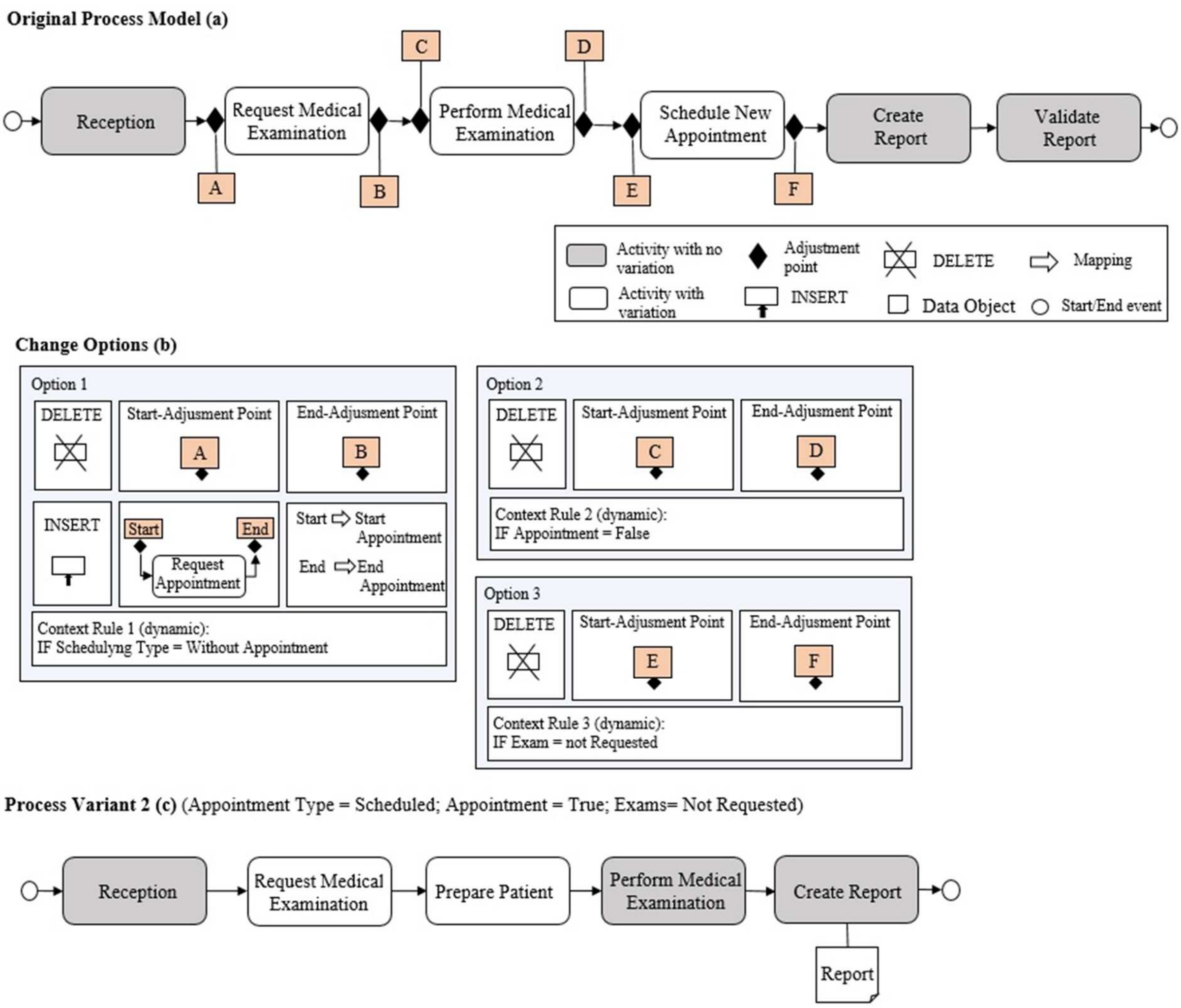
3.4.3. Fragment Customization

Approaches in this group, customize the process model by applying change operations, which enable to insert, move, replace or delete process fragments. In this group, the Process Variant by Options (Provop) is one of the most relevant approaches for process model customization (LA ROSA et al., 2017).

Developed by Hallerbach, Bauer and Reichert (2008), the Provop approach proposes to derive a process variant from a base process model and then adjusting the process variant to a given context. The base process model contains the most common behaviour from which, through adjustments according a specific context, a process variant can be derived.

The adjustments that can be applied to derive a process variant of the base model are expressed in terms of high-level change operations such as INSERT, DELETE, and MOVE process fragments. Furthermore, a MODIFY operation for changing attributes (e.g., actor assignment, activity durations) is provided. Change operations can be grouped into reusable sets, denoted as options, which allows their reuse and enable to configure more complex process. Figure 19 shows an example of the base process model with the adjustment points (a), the change options (b) and the derived process variant (c).

Figure 19 - Provop approach for handling medical examinations



Adjustment points correspond to entries or exits of activities and connector nodes respectively. They also can restrict the parts of the process model that are subject to variation. For example, the option 1 is related to changes that can happen only between the adjustment point A and B. The selection of this option relies on the context rule, which state that if the patient does not have an appointment, the activity (or activities) between these adjustment points must be deleted and the activity Request Appointment must be inserted.

Provop approach customize a process model by restricting or by extending the process model behaviour. Behavioural and structural correctness are not supported by this approach. The correctness must to be ensured after the customization by applying different techniques. Guidance and recommendations during the customization are not provided by the Provop approach.

3.5. DECISION SUPPORT TECHNIQUE

Some approaches for process model customization applies techniques for decision support during customization. The C-iEPC and C-YAWL approaches applies the questionnaire-model approach to provide guidance to the user. Provop applies the feature diagram to customize the process model. Thus, these techniques are explored in more detail in this section.

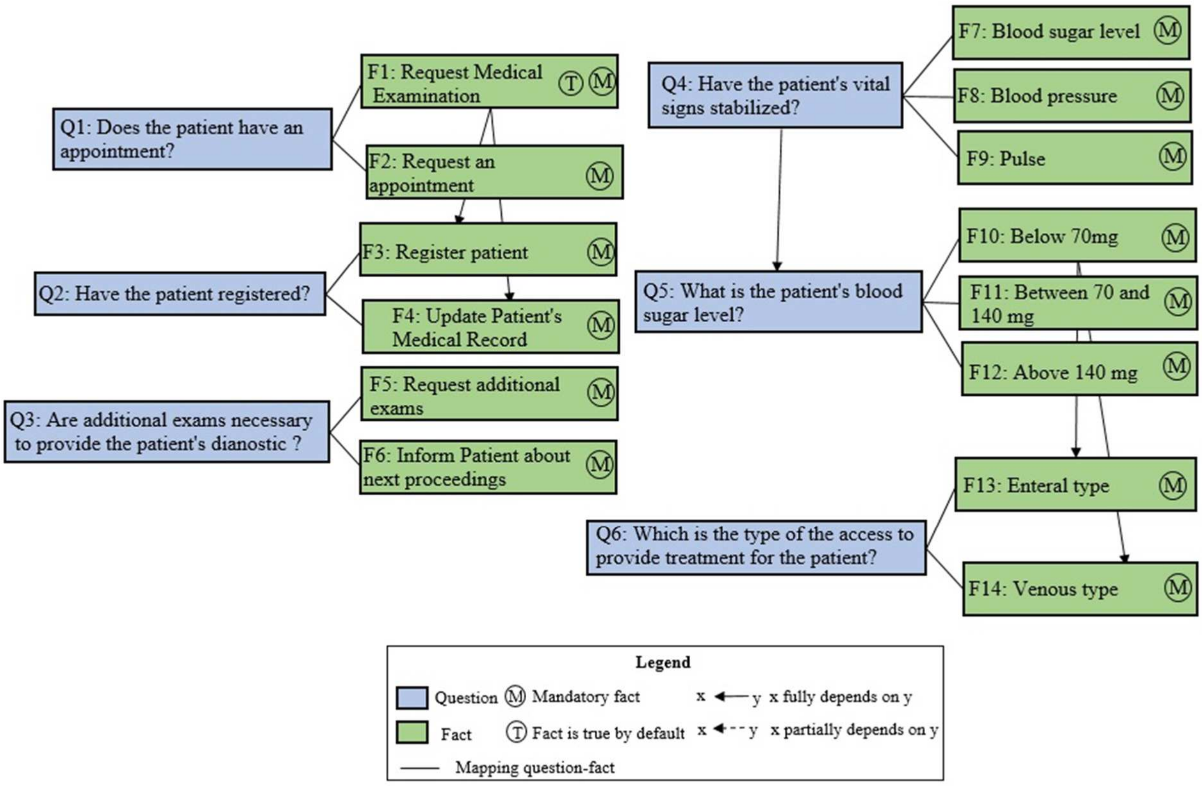
3.5.1. Questionnaire-Model approach

The questionnaire-model approach (LA ROSA et al., 2009) allows a configurable process model to be individualized by applying answers to questions about the respective deployment context. Thus, each question refers to a variation point and each domain fact corresponds to a Boolean variable representing a feature of the domain. Such a feature, in its turn, may either be enabled or disabled depending on the given application context. Thus, the link between configurable process models and questionnaire models is achieved by mapping each process variant to a condition over the values of domain facts, such that when the condition holds, the specific variant is selected (HALLERBACH, BAUER and REICHERT, 2010).

In the questionnaire-model approach some facts are defined as mandatory, which means they must be explicitly set by the user when answering the questionnaire. If a non-mandatory domain fact is left unset, its default value will be used (HALLERBACH. BAUER and REICHERT, 2010).

This approach also allows to specify the order of dependence on facts and questions. Dependence on facts is expressed by associating a set of alternative preconditions with a given fact x, where a precondition is a group of facts that must all be set before x. Only one precondition needs to be satisfied for a dependency to be fulfilled. There are two types of dependencies, a fact partially dependent on another fact if the latter belongs to at least one of its preconditions. On the other hand, a fact fully depends on another one if the latter belongs to all its preconditions. A full dependency subsumes a partial dependency. Figure 20 shows an example of a questionnaire-model. In this example, new information (related to questions Q4, Q5, and Q6) is added in order to explain the methodology.

Figure 20 - Questionnaire-model related with the process for handling medical examination



In Figure 20, default values have been assigned to the facts as true. For example, in the illustrative process model, only patients with an appointment have a medical examination. Thus, this fact is set with a default value. Factors F3 and F4 are fully dependent of F1, since they have one precondition containing only F1. Dependencies of facts affect the order in which questions are posed to users, since questions “inherit” the dependencies defined by their facts (DE MEDEIROS and GÜNTHER, 2005). In the example, since F3 and F4 depend on F1 in Q1, then Q2 automatically depends on Q1.

The dependencies in questions define the order that the questions are posed to users. This is allowed so long as the dependencies defined at question level do not contradict those defined at fact level. Full dependency means that a question must be asked before another one. For example, before asking question Q5, question Q4 must be answered, since answering Q5 depends on the settings of the domain facts produced by Q4. On the other hand, partial dependency means that there is no mandatory order among the questions.

By posing relevant questions for the user, in a consistent order with clearly established dependencies between question answers and facts, this approach can select process variants without violating any domain constraints. However, dependencies do not affect fact values. For example, with a dependency, we cannot capture the restriction on the blood sugar level values, which implies that only one fact among F10, F11 and F12 need to be asserted in Q5. Besides, answer to one question may limit the answers allowed for subsequent questions, and not all combinations of answers may lead to valid fact valuations. For example, if F8 is asserted in Q4, there is no need to know the type of access for the treatment since this information is important only if F7 is asserted in Q4, thus F13 and F14 must be negated in Q6. In this way, the dependencies between facts constitute constraints over the elements and can be modelled as logical expressions (HALLERBACH, BAUER and REICHERT, 2010; LA ROSA, 2009):

C1: $F1 \vee F2$
C2: $(F3 \veebar F4) \Leftrightarrow F1$
C3: $\neg (F3 \vee F4) \Leftrightarrow \neg F1$
C4: $F5 \vee F6$
C5: $(F10 \veebar F11 \veebar F12) \Leftrightarrow F7$
C6: $\neg (F10 \veebar F11 \veebar F12) \Leftrightarrow \neg F7$
C7: $(F13 \veebar F14) \Rightarrow F10$

The first constraint, C1 ensures that at least one fact must be chosen in Q1. C2 and C3 states that only one value in connection with patient's register must be chosen, if and only if, F1 is stated as true in question Q1. In the constraint C4, either fact F5 (Request additional exams) or F6 (Inform next proceedings) may be true, but not both. C5 and C6 states that only one value in connection with blood sugar level must be chosen, if and only if, F7 is stated as true in question Q4. In C7, the information about enteral access (F13) or venous access (F14) is selected if the patient displays a blood sugar level below than 70mg (F10).

In the next step, the link between the questionnaire model and the variation points is established by associating each variation point with a Boolean expression over the domain facts of its corresponding questionnaire-model as shown in the Table 3:

Table 3 - Link between domain facts and configurable activities

Configurable Activity	Configuration Alternative	Boolean Expression Over Facts
Patient's blood sugar level	On	F7
Enteral access	On	F10 ∧ F13
Venous access	On	F10 ∧ F14

Table 3 shows that a configurable activity is set to ON if the Boolean expressions for facts is evaluates as true; e.g., verify blood sugar level is configured to ON if fact F7 is true. In this way, a reference process model can be automatically and correctly configured.

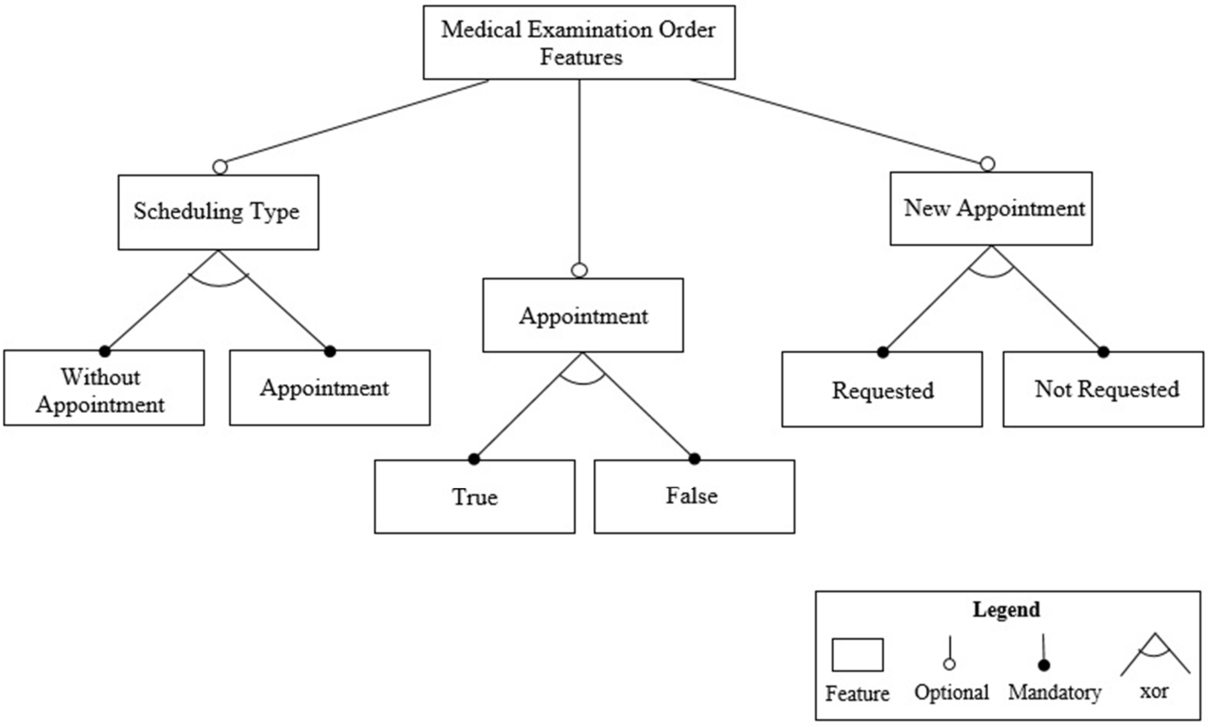
3.5.2. Feature Diagram

The feature diagram was initially proposed by Kang et al. (1990), during the development of their proposal Feature-Oriented Domain Analysis (FODA) aiming to discover and represent commonalities of software systems. Later, the feature diagram was also proposed for process model customization.

The feature diagram is a tree, whose roots represent the different features and sub-features related to the properties of a specific domain (LA ROSA et al, 2017). A feature can be defined as optional and mandatory. An optional feature can be selected or not, but a mandatory feature is always selected. In addition, a feature can be a parent of alternative features. The selection of an alternative feature requires the selection of the parent feature (KANG et al., 1990).

The relations between the features can be defined through constraints, expressed as propositional logical expression and can define the number of sub-features that a feature can have. Thus, if the sub-features are linked by an AND operator, they are all selected. If a XOR operator is the link between sub-features, only one sub-feature is selected. On the other side, if the sub-features are linked by an OR operator, one or more sub-features can be selected (LA ROSA et al., 2017). An example of a feature diagram is presented in Figure 21.

Figure 21 - Example of the feature diagram



The feature diagram illustrates in Figure 21 is composed of three optional features: ‘Scheduling type’, ‘Appointment’ and ‘New appointment’. Each feature contains mandatory sub-features linked by an XOR operation. Thus, in this situation a mandatory sub-feature can be excluded when not selected.

3.6. PROCESS MINING AND CUSTOMIZABLE PROCESS MODEL

Li, Reichert and Wombacher (2008a) developed a heuristic search algorithm for mining a collection of variants aiming to discover a base process model that covers the existing variants best. To derive the merged process model, change operations (e.g., to insert, delete or move activities) are performed such that the average distance between the new process model and the process variants become minimal. The distance between the process model and the variants is measured by the number of change operations.

The results obtained with the mining algorithm are compared with results obtained from traditional process mining such as Alpha and Alpha++ algorithm, Heuristic mining and Genetic mining (LI, REICHERT and WOMBACHER, 2008b). The efforts measured for respective process configurations are the numbers of high-level change operations needed to transform the generic model into the respective model variant. Later, Li, Reichert and Wombacher (2010) developed a clustering algorithm to merge the process variants into a generic process model.

Rozinat, Mans and van der Aalst (2006) analysed the influence of data attributes to the choices made in process models based on past process executions. Process mining is applied to discover the decision points, i.e., the parts of the model that may change. The next step is then to apply decision point analysis to identify the properties followed by cases that followed the same route. The decision point analysis converts every decision point into a classification problem, whereas the classes are the different decisions that can be made. The classification problem is solved by means of decision trees, which enable to infer logical expressions that form the decision rules that restrict the process behaviour.

Buijs, Van Dongen and Van der Aalst (2013) proposed four approaches to discover a configurable process model from a collection of event logs. The first approach applies process discovery on each input event log to obtain the respective process model, then the process models are merged. The second approach proposes to discover a process model that describes the behaviour of all event logs. Then, each event log is individualized from the single process model. In the third approach, a single process model is discovered that describes the behaviour of all event logs. Then, using each individual event log, configurations are discovered for this single process model. In the fourth approach, the discovery of the process model and the configuration is combined.

Buijs and Reijers (2014) proposes an approach that helps to compare how different organizations carry out essentially the same processes. The approach allows to compare the intended and the actual execution of a business process and supporting the comparison of the execution of process variants. Four processes from different companies are analysed considering its commonalities and differences. The comparisons are visualized through a so-called alignment matrix, which provide the connection between the modelled behaviour of a process to its observed behaviour as recorded in the event log. A framework, implemented as a plug-in in the ProM framework, is proposed to facilitate the comparison between the different process models. The framework compares the processes models by analysing the process model and its behaviour by means of three metrics: process model metrics, event log metrics, and comparison metrics.

3.7. ONTOLOGIES AND CUSTOMIZABLE PROCESS MODEL

Huang et al., (2013) proposes a framework to configure a process model by means of SWRL based on business rules. An ontology formalizes the knowledge about the variation points, in which the guidelines of variable points are presented by SWRL rules. Another ontology formalizes a set of domain-specific rules to get the specific rules needed to individualize the process variant according to the users' requirements.

The process is modelled in C-EPC. In the first step of the framework, the domain experts set the business rule ontology regarding a specific domain business context. Then, business programmers set domain variation points ontology, or choose business rule ontology. Some useful intermediate domain-specific fact can be obtained by combining the requirements proposed by users or business context and SWRL-based domain-specific business rules, which can be used as the input of the domain variation point ontology. After the reasoning step, performed by a reasoner engine on the ontology knowledge, the output of the domain variable point ontology is the configured C-EPC. For configuring a process variant, an algorithm is proposed, in which the input is a set of rules and a specific configurable business process. The output is the configured business process.

El Faquih, Sbaï and Fredj (2014) proposed a framework to semantically enrich customizable process models. The framework is composed by three components: CPM (configurable process model) component, which contains an e-health care process which is modelled using Variant Rich BPMN notation; CPM ontology component is used to capitalize the variability concepts of the configurable process model. The ontology is expressed using OWL 2 Web Ontology Language; and Domain ontology component is the ontology related to e-healthcare domain, called E-hospital ontology. The two ontologies are linked via a set of semantic rules which identifies the semantic constraints between the two ontologies. El Faquih, Sbaï and Fredj (2015) focused on the semantic validation of configurable process models by using the CPM ontology. The idea is developing and applying an ontology that capitalizes the CPM variability constraints.

3.8. SYNTHESIS

This chapter discussed some approaches to customize a process model through configuration nodes (or variation points). Seven approaches have been discussed: C-EPC, C-iEPC, C-YAWL, ADOM, Configurative Process Modelling, PESOA and Provop. These approaches have been analysed regarding five criteria: structural and behavioural correctness, guidance, recommendations, and the relation between variation points. Table 4 summarizes the evaluation results indicating to what extent the approach in question covers each evaluation criterion. We used "+" indicate a criterion that is fulfilled, a "-" indicate a criterion that is not fulfilled and a "+/-" to indicate partial fulfilment.

Table 4 - Evaluation results

Approaches / **Evaluation Criteria**	**Customization by Restriction**	**Customization by Extension**	**Structural Correctness**	**Behavioural Correctness**	**Guidance**	**Recommendations**	**Relation between variation points**
C-EPC	+	-	+	+	+	+/-	+/-
C-iEPC	+	-	+	+	+	-	-
C-YAWL	+	-	+	+	+	-	-
ADOM	+	+	+/-	-	-	-	-
Configurative Process Modelling	+	+	+/-	-	-	-	-
PESOA	+	-	-	-	-	-	-
Provop	+	+	-	-	-	-	-

C-EPC (ROSEMANN and VAN DER AALST, 2007) was one of the first approaches developed. The process model is customized by applying constraints to the configurable nodes, thus restricting the process behaviour and ensuring behavioural and structural correctness. Guidance can also be provided, but they are related to each variation point and not with the business contexts. Besides, requirements and guidelines are expressed as logical predicates and in big and complex process models can be complicate the comprehension about the relationships between the many variation points that may exist.

Process models customized by means of C-EPC approach do not represent variability related to the roles and objects. To fulfil this gap, it was developed the C-iEPC approach. The methodology is basically the same of the one provided by C-EPC. Configurable workflows were the third approach discussed, which focuses on executable business process model. To apply the configurable workflow approach, the Configurable YAWL (C-YAWL) was developed. This approach customizes the process model by hiding or blocking the configurable activities. The rules to select the activities are not expressed in the process model, thus is not clear the relationship between the nodes.

Structural and behavioural correctness are ensured in C-iEPC and C-YAWL by means of algorithm. Both approaches enable to apply the questionnaire-model approach in order to provide guidance to the user during customization. The questionnaire-model approach is composed by questions and facts. Each question corresponds to a variation point and each fact corresponds to the alternatives available for the variation point in question. Thus, by selecting a fact, a process variant is selected.

In turn, ADOM was developed to increase the adaptability of the process model. This approach is the only one among the approaches analysed that enable the customization by restriction and by extension. The other three approaches enable only the customization by restriction. ADOM cannot ensure the behavioural and structural correctness. Guidance and recommendations are not provided. The relationships among the variation points cannot be identified. Besides, the ADOM-EPC model has a higher level of abstraction which can raise ambiguities (REINHARTZ-BERGER, SOFFER and STURM, 2010).

Beside the configurable nodes, the mechanisms for customize process model can also be classified as element annotation, activity specialization and fragment customization. Thus, for each classification, an approach is presented. Regarding the element annotation, the Configurative Process Modelling approach is presented. This approach is considered one of the most relevant approaches in this group. The Configurative Process Modelling approach customize the process model by means of the configuration parameters which are attached with the variable parts of the process model (BECKER et al., 2004; BECKER, DELFMANN and KNACKSTEDT, 2007).

PESOA is the approach classified as activity specialization. This approach is characterized by the existence of stereotypes marking the variable activities. The features related with the alternatives for customize the process model. These features are described in a feature diagram, which stablish the link between the process variants and the respective features (PUHLMANN et al., 2005).

The last approach refers to the Provop approach, which is classified as fragment customization. For the customization, adjustment points are defined in the points of the process model that are subject to variation. Thus, in each adjustment point, a change operation is applied, which enable to insert, replace, move or delete process fragments (HALLERBACH, BAUER and REICHERT, 2008).

This chapter also discussed the application of process mining regarding the customization of process models. Process mining have been applied to merge process variants into a single model. However, the studies analysed about the application of process mining in configurable process model show that process mining is not applied to identify process variants.

Thus, this research aims to identify from an event log the process variants and the requirements to select each one of them by means of the discovery of the variation points.

By identifying the process variants through process mining techniques enable to discover deviations or problems that may exist and correct them. The analysis of the process behaviour also enables to capture knowledge about the process that are not explicit, as for example, a relation between two or more activities that need to be considered when configuring the process variants.

Regarding the use of ontologies on customizable process models, Huang et al., (2013) developed an approach to customize a process model by reasoning on two ontologies: one related to the variation points and other related to the rules of the business context. For configuring a process variant an algorithm is proposed, which input is a number of rules and a specific configurable business process. The output is the configurable business process. In this approach, all the requirements to obtain a process variant are provided before to start the customization. Besides, recommendations are not provided and the approach is not user friendly. The user must provide the requirements for a business programmer engineer to obtain a process variant. In addition, external regulations are not considered for customize the process model.

The approaches for process model customization analysed in this chapter shows that behavioural and structural correctness can be provided by means of algorithms. Guidance can be provided by applying the questionnaire-model approach. However, these approaches cannot provide recommendations during the customization. The approaches also cannot provide a view about the relationships among the variation points. For example, by selecting a fact related to a variation point, the user cannot evaluate the impact of the decision in the process model customization.

These issues are essential for process model customization (LA ROSA et al., 2017; AYORA BÜHNE, HALMANS and POHL, 2003, VALENÇA et al., 2013). The approach proposed in this research fulfils the gaps mentioned before. By customizing the process model through ontologies, recommendations related with the choices made during customization can be provided for the user. Since rules define the relationship between the variation points, the impact of the decision about a variation point can be evaluated by the user. By applying the questionnaire model, the customization can respect the user’s requirements and provide guidance. Besides, the customization by means of the questionnaire model is user-friendly.

4. FRAMEWORK FOR CUSTOMIZING PROCESS VARIANTS THROUGH PROCESS MINING AND SEMANTIC REASONING

The previous chapters discussed the needs for customize a process model in a way that the resulting process variant meets the user's goals best. The main requirements to develop a customizable process model is to identify the parts of the model that are common to all process variants, the parts that can change (i.e., the parts that are subject to variation), how they change, and the reasons for changing.

These aspects can be identified through process mining which can provide knowledge about how the process is performed, thus helping on making appropriate decisions to improve it. However, despite the benefits that the event log analysis can provide, many enterprises do not appropriately use such data.

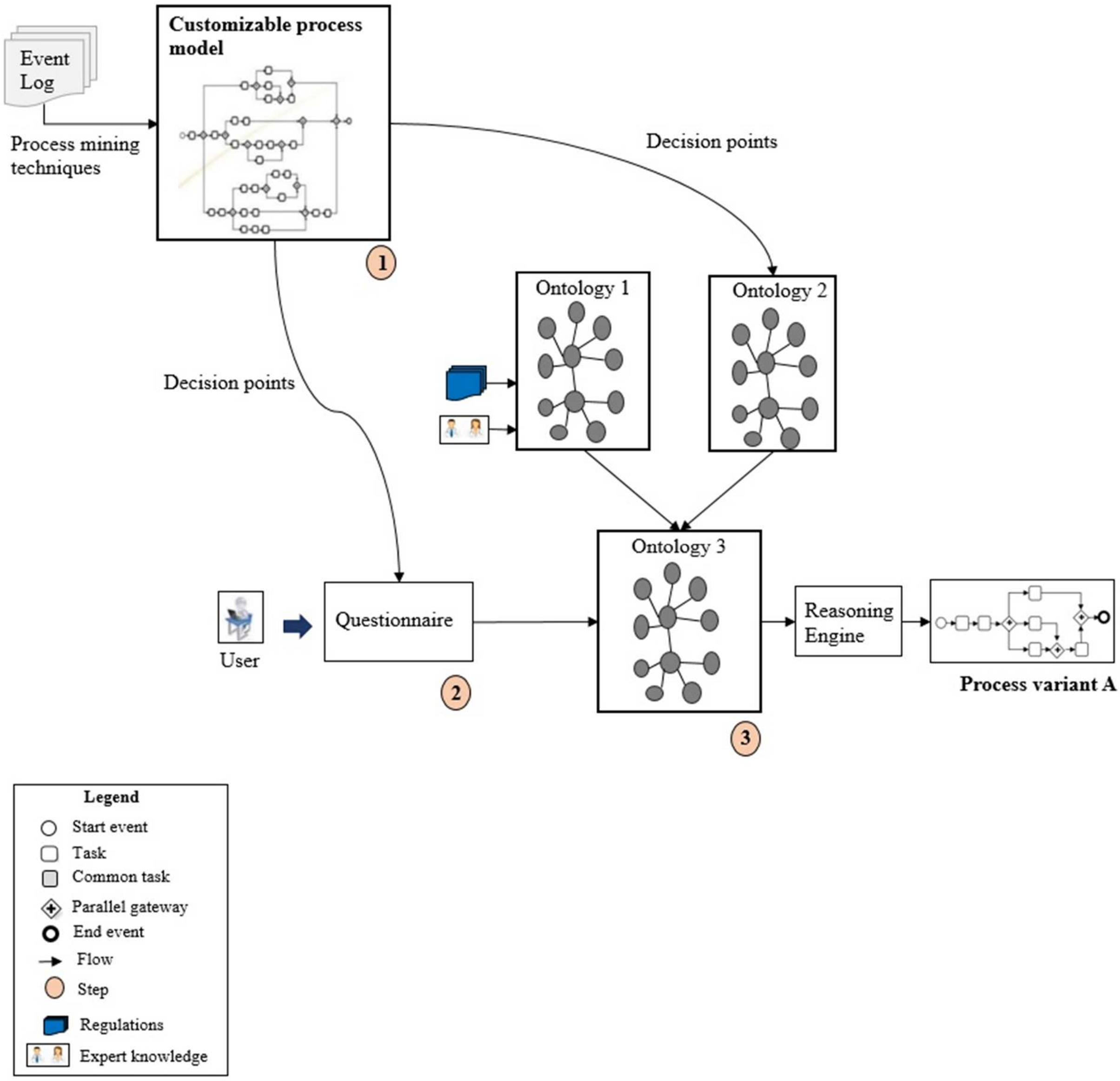
Obtaining the configurable process model by means of the event log enables to improve the process variants by correcting deviations, if they exist, anticipating problems, discovering if the requirements have been followed, etc. Besides, the implicit knowledge can be captured and made explicit, thus enabling to enrich the process variants.

In each variation point, a decision need to be made in order to obtain a process variant. Thus, providing guidance and recommendations for the user during the customization ensure that the resulting process variant is correctly customized. The process variants also need to be correctly in a behavioural and structural way.

The need for recommendations is demonstrate by the analysis previously carried out (Chapter 2). The recommendations provided by the existing approaches are limited to recommendations about the variation points (VALENÇA et al., 2013; LA ROSA et al., 2017; REICHERT and WEBER, 2012). However, recommendations can also be related with the application business context, which can improve the customization. Thus, the recommendations may include the information about the variation points (alternatives and rules) and the information about the business context, including internal and external regulations. Considering this need, ontologies can be applied to support decision making during the process model customization.

The framework proposed in this research, intends to provide a decision-making support during the process model customization. The framework focuses in the aspects such as internal and/or external regulations and expert knowledge to provide recommendations about the business context during the process model individualization. Besides, the knowledge about the actual business process executions is captured in order to improve the process model. Figure 22 depicts the proposed framework.

Figure 22 - Framework for customize process variants



The framework presented in Figure 22 contains three steps. In the first step, process mining techniques are applied to discover the process model from an event log. The event log contains all information about a business process. Thus, by analysing the event log through the process mining techniques, all instances of all paths are obtained. Then the properties of those instances are identified to build a generic model from which all instances may have from (DETRO, et al., 2017).

As an event log can be incomplete or contain noise, an approach to deal with these issues is proposed, resulting in the development of an enriched process model. Thus, an event log is generated in order to analyse the scenarios existing in the developed process model and to discover the relations between the elements in the process model.

For customizing the process model, three aspects need to be identified: the variation points, i.e., the points where the process is split into alternatives branches (OR-split), the alternatives for them, and the rules for choosing the available alternatives. The rules ensure that the configuration of a process variant respects context-specific requirements. Thus, step 1 also aims to discover the rules related to each path and to check its compliance according to the rules of the business application context in order to avoid configuring incorrect process variants. In this step, a decision tree is used to carry out a decision mining analysis, i.e., to find out which properties of a case might lead to taking certain paths in the process.

The process model customization is performed when the user selects an alternative related with each variation point. Thus, in step 2, it is proposed to apply the questionnaire-model approach (LA ROSA et al., 2009) for process variants configuration. The questionnaire-model approach guides the configuration process by posing questions to users whose answers define the process variant selection. The questionnaire is developed based on the knowledge related with the variation points and the rules for selecting each path captured in step 1.

A process variant should be configured respecting the user requirements, but also it should respect the internal and external regulations. Besides, choices in one point of the process model can influence the choices in other points. Thus, step 3 refers to the use of ontologies to formalize all the involved aspects for process model customization and the relations among them in order to individualize the process model according to the user's requirements.

For developing the step related with the knowledge formalization, two sources of knowledge are necessary. Thus, the framework proposes to use two ontologies which are then merged into a new ontology. In this way, one ontology refers to the variation points, the alternative for them and the rules for selecting the alternatives. The second ontology formalizes the knowledge related to the internal and/or external regulations, which is enriched with the expert knowledge.

Thus, by answering the questionnaire and by reasoning on the merged ontology, the process variant is selected according to the regulations and recommendations provided by the user during the customization. The steps related to the proposed framework are discussed in more detail in the next sections.

4.1. DISCOVERING THE PROCESS MODEL

The first step from the proposed approach refers to the discovery of the process model from an event log. As mentioned previously, an event refers to an activity (i.e., a well-defined step in the process) on a particular case (i.e., a process instance). Event logs may store additional information such as the resource (i.e., person or device) executing or initiating an activity, the event's timestamp or data information associated with such event (e.g., the size of an order) (VAN DER AALST, 2012, VAN DER AALST and DUSTDAR, 2012).

However, as mentioned in Section 2.2.2 an event log can be incomplete or may contain noise. Basically, the event log contains noise if it contains rare and infrequent behaviour not representative for the typical behaviour. An event log is incomplete if it contains too few events to be able to discover some of the underlying control-flow structures (VAN DER AALST, 2011).

As a result, the event log may not contain all the activities performed during the process execution, all the sequences and some attributes may be missing. Thus, the event log cannot reflect the correct process model behaviour. In this way, it is proposed to build a process model that can be considered as a reference process model by representing all process variants. This process model can be based on the event log, the internal and/or external regulations, and the expert knowledge.

Building the process model through the event log, the internal and/or external regulations and the expert's knowledge enables to obtain a prescriptive process model, which addresses all relevant issues related to the business context. When the process model is developed, it is necessary to analyse all the scenarios that can be extracted from this process model, what parts are common to the process variants, and the relations between the elements of the process model. Thus, an event log should be obtained in order to simulate these various scenarios.

4.2. OBTAINING AN EVENT LOG

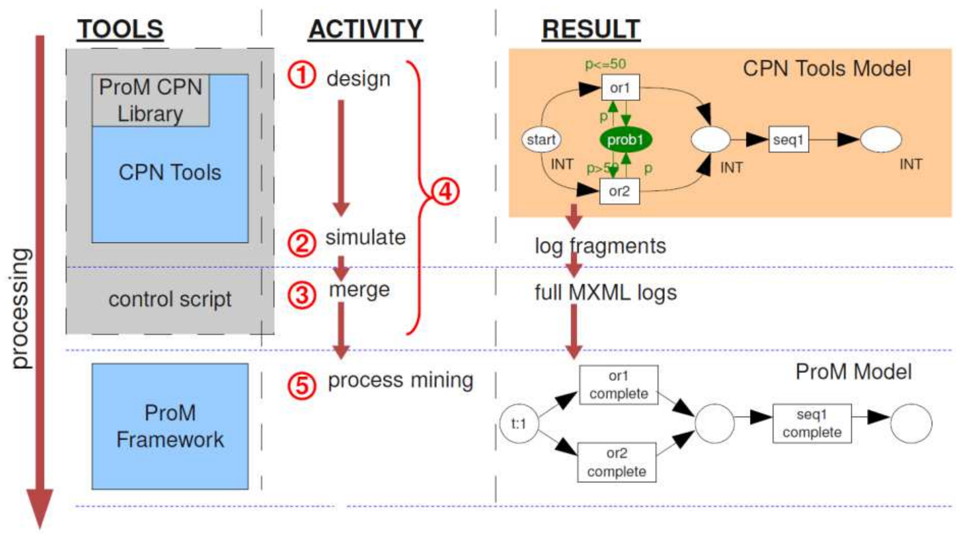
Since the process model was developed considering the regulations, the event log and the expert knowledge, it is necessary to obtain an event log related to this process model, which enable to apply the process mining techniques. By applying the process mining techniques, implicit relations between the elements of the process model can be identified. It also enables to identify the process variants and the rules to select them.

Thus, next step refers to applying process mining techniques in an event log related with the developed process model. Thus, an artificial event log was developed through Coloured Petri Nets (CP-net or CPNs), which are a graphical language for constructing models of concurrent systems and analysing their properties (JENSEN, KRISTENSEN and WELLS, 2007).

An artificial event log can be used when a real-life log is incomplete and/or contain noise (DE MEDEIROS and GÜNTHER, 2005). This approach enables to investigate different scenarios in detail and check whether the expected results are achieved (BARUWA, PIERA and GUASCH, 2016; AIZED, 2009).

According to De Medeiros and Günther (2005), the main idea for obtaining an artificial event log is to create random MXML logs by simulating CP-nets in CPN Tools. Basically, two steps are necessary to create MXML logs using CPN Tools: (1) Modify a CP-net to invoke functions that will create logs for every case executed by the CP-net, (2) Use ProM Import framework (2017) to group the logs for the individual cases into a single MXML log, as shown in the Figure 23.

Figure 23 - Generating logs using CPN Tools

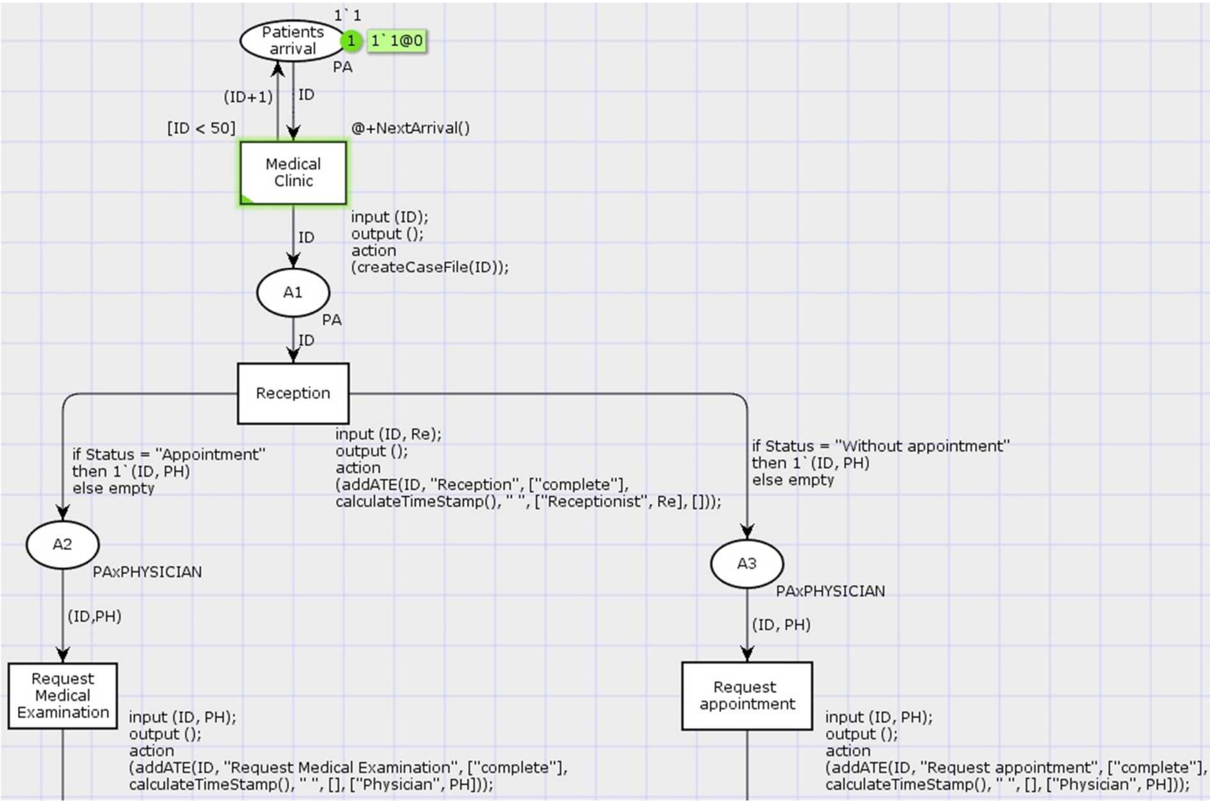


According to Figure 23, activities 1 and 2 are referred to generate records of events logs using CPN Tools. Activity 3 applies tools to group logs for individual cases into a single MXML file. Activity 5 represents the loading of MXML file into ProM (2017) to run the discovery algorithms. The result is the discovery of the process model, which enables extracting of process variants.

4.2.1. Coloured Petri Net

The CPN is composed by places (drawn as ellipses or circles), transitions (drawn as rectangular boxes), arcs and finally some text (inscriptions). Places and transitions are called nodes. Attached to the transitions, there are the code segments consisting of a piece of sequential CPN ML that is executed whenever the corresponding transition occurs in the simulation of the CPN model enabling the generation of the event log (AIZED, 2009; CPN Tools, 2017). A Coloured Petri Net was developed based on the illustrative case for handling medical examinations (Section 3.2), which excerpt is depicted in Figure 24.

Figure 24 - Example of a Coloured Petri Net



The CPN is composed by four transitions, four places, and a number of arcs. As shown in the example, the CPN only allow an arc to connect a transition to a place or a place to a transition. Transitions and places also can be connected by double-headed arcs, such as the place ‘Patient arrival’. A double-headed arc implies that the place is an input place (i.e., places with an arc leading to the transition) and an output place (i.e., arcs coming from the transitions).

Places represent the state of the modelled system. Each place can be marked with one or more tokens, and each token has a data value, called token colour, attached to it. Thus, the state of the system is represented by the number of tokens and the token colours on the individual places (JENSEN, KRISTENSEN and WELLS, 2007). For example, the state of the transition ‘Medical clinic’ is defined by the place ‘Patient arrival’ and the places A1, A2, and A3 define the state of the transitions ‘Reception’, ‘Request medical examination’ and ‘Request appointment’, respectively.

The set of possible token colours is specified by means of a type, and it is called the colour set of the place. In the CPN Tools, colour sets are defined using the CPN ML keyword colset. The colour sets, variables and functions that are used in the CPN are defined as Declarations for the model. Figure 25 shows the declarations for the CPN model.

Figure 25 - Declarations for the CPN model

```
Declarations
    *Standard declarations
    colset Unit = unit;
    colset BOOL = bool;
    colset INT = int;
    *Declarations of the model
    colset PA = INT timed;
    colset AGE = int with 17 ..90;
    colset ID_PATIENT = int with 100 ..150;
    colset RE = subset STRING with ["Receptionist"];
    colset STATUS = subset STRING with ["Appointment", " Without appointment"];
    colset PHYSICIAN = subset STRING with ["1015", "1859"];
    colset PAxPHYSICIAN = product PA * PHYSICIAN;
    *Variables
    var ID:PA;
    var Status: STATUS;
    var PH:PHYSICIAN;
    var ID_Patient: ID_PATIENT;
    var Re:RE;
    var age:AGE;
    *Functions
    fun NextArrival() = discrete(5,60);
    fun OK(ID) =
    if ID<50 then true
    else false;
    *Log declarations
    val FILE = "./logs/logsCPN"
    val FILE_EXTENSION = ".cpnxml"
    use "loggingFunctionsMultipleFiles.sml;
```

The colour set of the place ‘Patient Arrival’ is PA, which is defined as integer type ‘int’ and is used to model the sequence of patients. The remaining places has the colour set ‘PAxPHYSICIAN’. ‘PHYSICIAN’ is defined as text string and is used to model the id of the physician responsible for the patient’s treatment. Thus, the colour set ‘PAxPHYSICIAN’ is defined to be the product of the types ‘PA’ and ‘PHYSICIAN’. This means that the colour set ‘PAxPHYSICIAN’ is used to model the data packets which contain a sequence number and some data.

Places also contain an inscription, written above them, which determines the initial marking of the place. For example, the inscription at the upper right side of the place ‘Patient’s Arrival’ specifies that the initial marking of this place consists of one token with the colour value 1. This indicates the order that the patients arrive in the medical clinic.

Next to the arcs is positioned a textual inscription, called arc expression. When a transition occurs, the arc expressions determine which colour of tokens need to be removed from input places and added to output places. The arc expressions are built from typed variables, constants, operators, and functions. For example, the variable ID is bound to a value of type PA and the variable PH is bound to a value of type Physician.

According to Figure 24, the transition ‘Medical clinic’ is the only transition with a thick green border line, which means that it has an enabled binding. When this transition occurs, the transition ‘Reception’ is enabled. Attached to the transition ‘Medical clinic’, the function ‘@NextArrival() defines the time among the arrival of the patients. According to this function, the time that a patient may arrive vary between 5 and 60 units of time. Another function OK(ID) defines that the event log is composed of 50 cases (i.e., patients).

Two ML functions enable the generation of the data log: createCaseFile and addATE. The function createCaseFile receives an integer as input and it opens the log file for a case. This function should be invoked only once per case and before the function addATE be invoked for this same case. In the example, this function is attached with the transition ‘Medical Clinic’. The function addATE logs the execution of a transition to the log of a case. The parameters of the function addATE are (DE MEDEIROS and GÜNTHER, 2005):

- **caseID –** integer that identifies a case.
- **transitionName –** string that has the name of the transition to log.
- **eventType –** list of strings. If the event type is supported, the list should contain a single element and have the format [name], where name in {“assign”, “withdraw”, “reassign”, “start”, “suspend”, “resume”, “complete”, “autoskip”, “manualskip”, “pi abort”,

“ate_abort”}. If the event type is unknown, this list should have two elements and the format [“unknown", “name”], where name is the unknown event type name.

- **Timestamp –** string that represents the date and time in which the task was executed. The function calculateTimeStamp() is provided to automatically calculate the timestamp field based on the current time (in minutes) of a CP-net.
- **Originator –** string that has the name of the originator (person or system) that executed the transition.
- **Data –** list of strings containing the additional data fields that may be associated to a task.

The parameters timestamp, originator and data are optional. The code segments attached to the transitions ‘Reception’, ‘Request medical examination’ and ‘Request appointment’ are composed by the caseID, transitionName, eventType, calculateTimeStamp() and the resources (i.e., the receptionist and the physician’s code).

The declarations of a CP-net need to be modified to import the ML functions to log transitions. These functions are in the file logging-FunctionsMultipleFiles.sml. The ML functions in this file use two constants: ‘FILE’ and ‘FILE_EXTENSION’. The declaration FILE sets the location and the name prefix of the S-MXML files that the CPN will create for every executed case. The FILE_EXTENSION set the extension that these created files have. For each execution, a new data-log file will be generated in the format ‘.cpnxml’, which after simulation should be joined together in a single file through the use of the software ProM Import Framework (DE MEDEIROS and GÜNTHER, 2005).

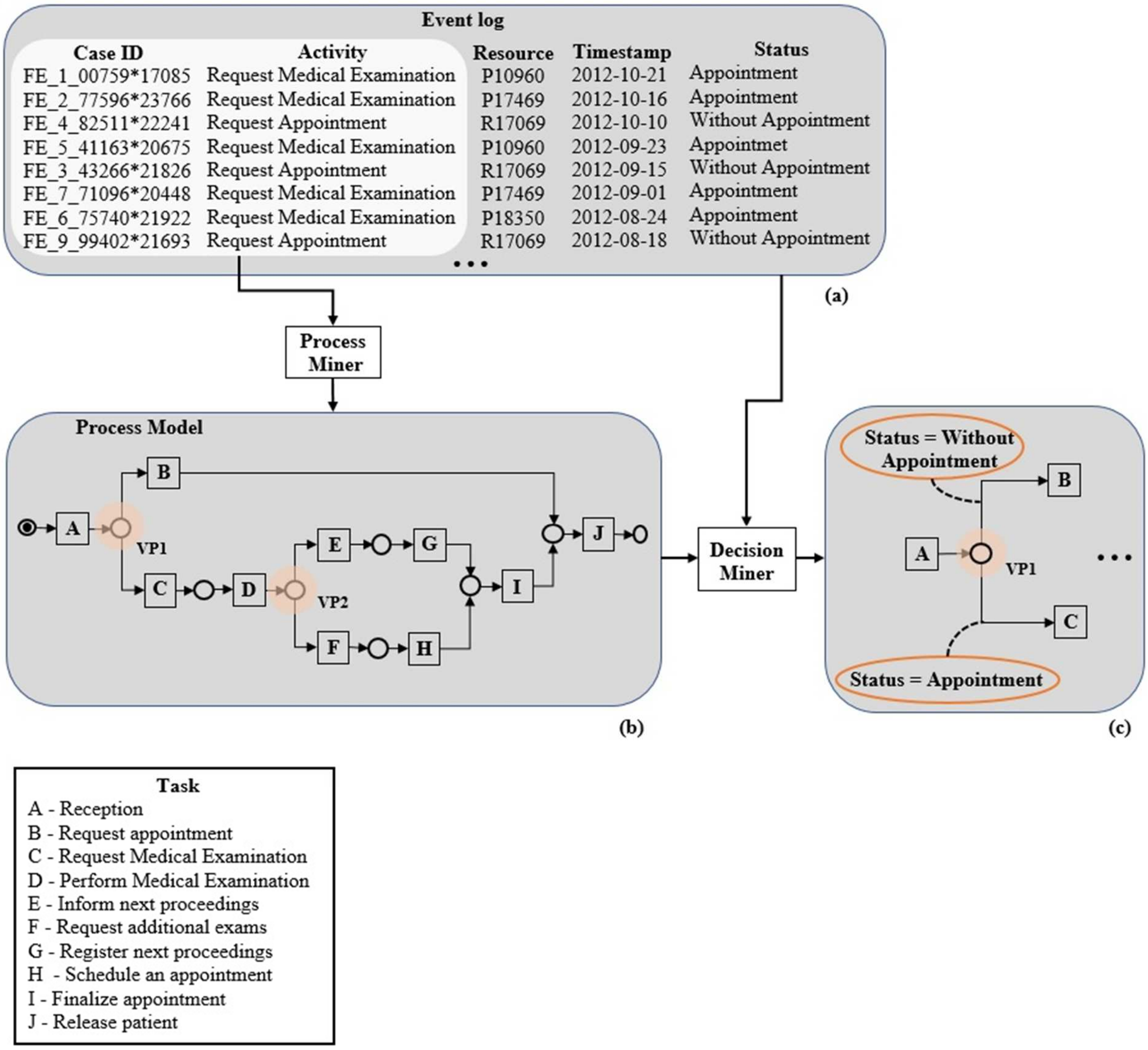
ProM Import Framework has been developed by De Medeiros and Günther, (2005) to serve as a common environment for converting and importing logs from all kinds of information systems, and subsequently creating MXML compliant log files from them (JENSEN, KRISTENSEN and WELLS, 2007; ProM Import, 2007). When the event log is obtained, process mining techniques can be applied to obtain and analyse the process model.

4.2.2. Extracting process variants

The next step refers with the definition of the three aspects needed to build a customizable process model: the variation points also called decision points, i.e., the points where the process is split into alternatives branches (OR-split); the alternatives for each variation point; and the rules for the selection of the available alternatives.

The decision mining analysis is a process mining technique, which aims at the detection of data dependencies that affect the routing of a case (PROCESS MINING, 2017). This technique enables to identify the variation points and based on data attributes associated to the cases in the event log, enable to discover the rules for following one route or the other. Figure 26 shows the decision mining technique.

Figure 26 - Decision mining analysis



Source: Adapted from Rozinat and Van der Aalst, 2006b.

According to Figure 26(a), the event log contains the case (i.e., each patient), the activities, the information about the people executing the activities (e.g., physicians and the receptionist), the timestamp (i.e., date that the activity was performed) and the data involved. By applying process mining techniques, such as the α-algorithm, the process model is discovered (Figure 26b). Classical process mining techniques discover the process model based only in the data related to the case and the activities executed. As Figure 26(b), the process mining identifies the decision points, but no information is provided for the choices. Thus, the decision mining aim to discover the rules for choosing each alternative (Figure 26(c)).

Once the decision points are discovered, it is necessary to investigate if the cases following each route are influenced by case data, i.e., if the groups following the same route share the same properties. Thus, machine learning techniques can be applied to discover the structural patterns in a data base (ROZINAT and VAN DER AALST, 2006b).

In a broadly way, machine learning can be defined as computational methods using experience to improve performance or to make accurate predictions (MOHRI, ROSTAMIZADEH and TALWALKAR, 2012). According to Mitchell (2006) a machine learns with respect to a particular task T, performance metric P, and type of experience E, if the system reliably improves its performance P at task T, following experience E. Depending on how we specify T, P, and E, the learning task might also be called by names such as data mining, autonomous discovery, database updating, programming by example, etc.

Data mining provides techniques for finding and describing structural patterns in data as a tool for helping to explain that data and make predictions from it. In data mining applications, there are four basically different styles of learning (WITTEN and FRANK, 2005):

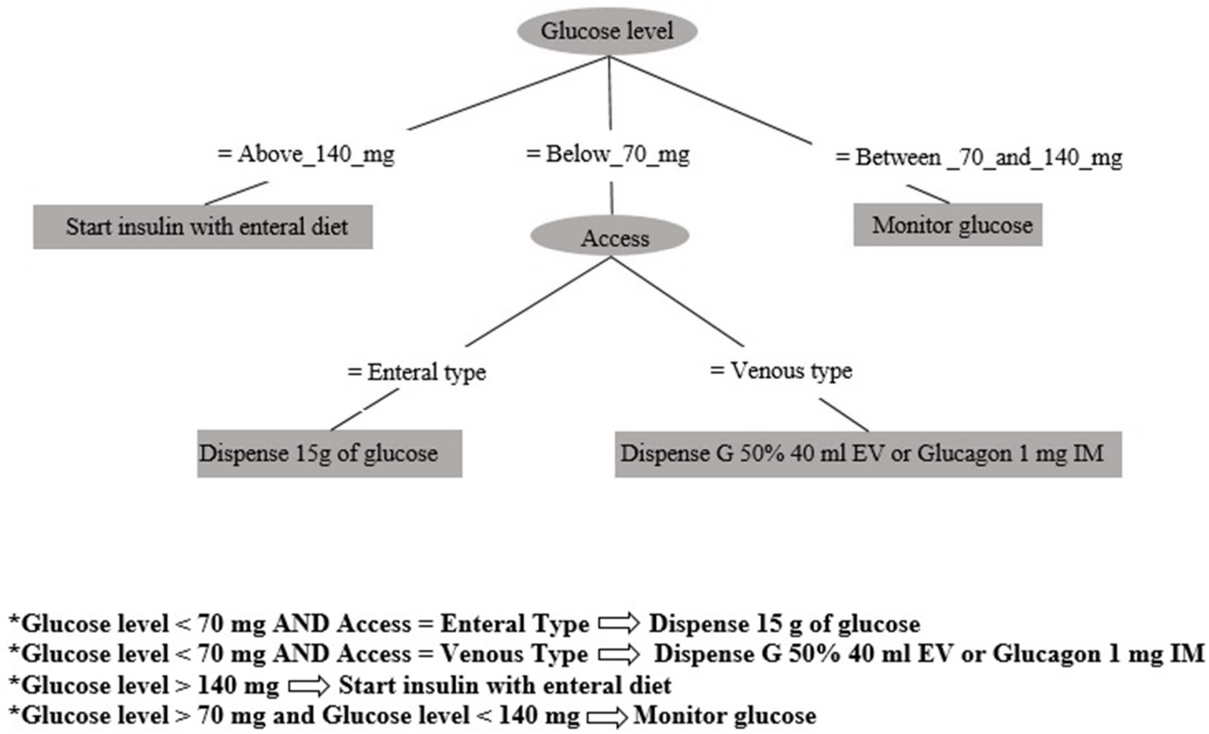
- *Classification learning:* the learning scheme is presented with a set of classified examples from which it is expected to learn a way of classifying unseen examples.
- *Association learning:* any association among features is sought, not just ones that predict a particular class value.
- *Clustering:* groups of examples that belong together are sought.
- *Numeric prediction:* the outcome to be predicted is not a discrete class but a numeric quantity.

Regardless of the type of learning involved, the thing to be learned is called concept and the output produced by a learning scheme is called concept description. Depending of the algorithm applied, a conceptual description may be represented in terms of rules or a decision tree (WITTEN and FRANK, 2005).

The decision mining applies the decision tree algorithm J48 provided by the WEKA software library. This algorithm is part of a set of computer programs, called C4.5, that construct classification models by discovering and analysing patterns found in such records (QUINLAN, 2014).

A decision tree is a flowchart - like tree structure, where each internal node (non-leaf node) denotes a test on an attribute, each branch represents an outcome of the test, and each leaf node holds a class label. The topmost node in a tree is the root node. Thus, each rule represents a unique path from the root to each leaf (AGRAWAL and GUPTA, 2013). An example of a decision tree is depicted in Figure 27.

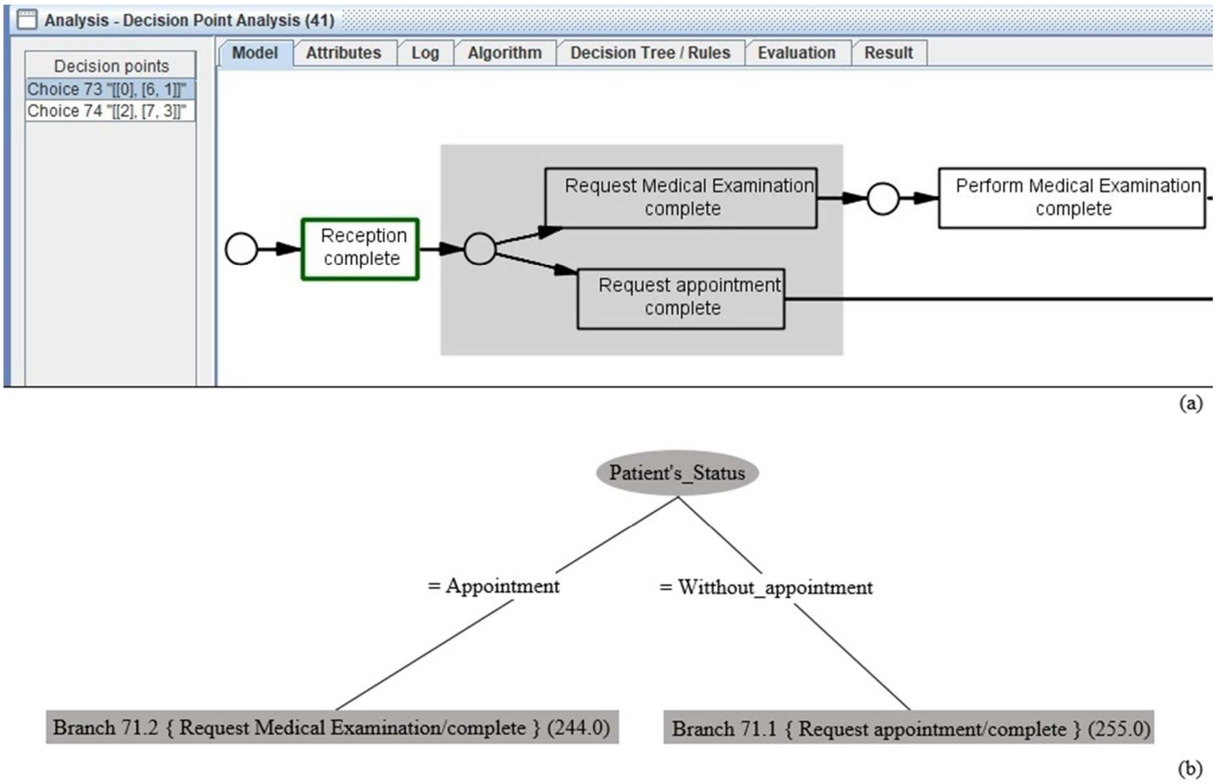
Figure 27 - Example of a decision tree



As shown in Figure 27, the decision tree analysis enables to infer logical expressions which form the decision rules. If an instance is in one of the leaf nodes of a decision tree, it fulfils all the predicates on the way from the root to the leaf, i.e., they are connected by a Boolean AND operator. When a decision class is represented by multiple leaf nodes in the decision tree the leaf expressions are combined via a Boolean OR operator (ROZINAT and VAN DER AALST, 2006b).

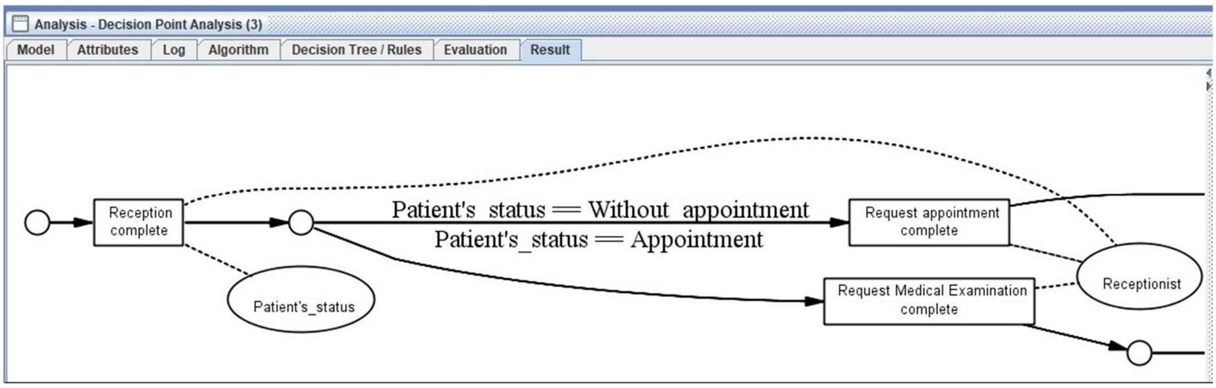
The decision mining analysis was applied in the event log related with the process for handling medical examinations. Figure 28(a) shows an excerpt of the process model, in which two decision points were identified (Choice 73 and Choice 74).

Figure 28 - Decision mining analysis



In Figure 28(a), the first decision point is highlighted (Choice 73). This decision point has two alternatives available: request medical examination and request an appointment. Figure 28(b) shows the decision tree related with this variation point. The decision tree enables to identify the rules for choosing the available alternatives. Decision mining also enable to visualize the rules and the attributes in the context of the process model, as show in the Figure 29.

Figure 29 - Example of the rules in the process model

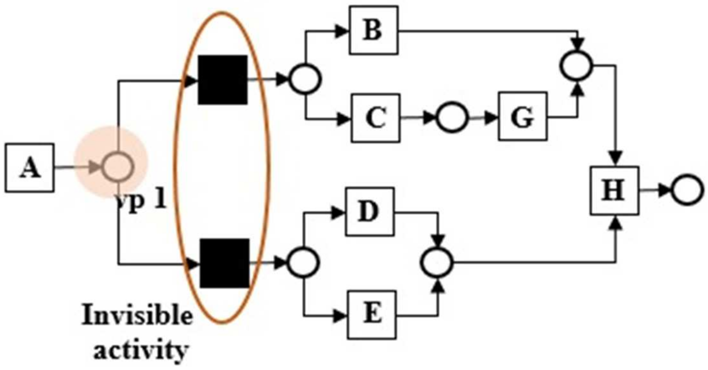


In Figure 29, it is possible to identify that the receptionist is responsible for the activities: 'Reception', 'Request appointment' and 'Request medical examination'. This option of the decision mining enables to visualize all attributes related with the execution of a particular activity. The information about the attributes related with the activities can also be applied to enrich the ontologies developed in the step 3.

According to Rozinat and van der Aalst (2006), there are two challenges that need to be considered when applying the decision mining analysis. The first is related with the quality of the event log (e.g., the existence of noise) and the correct interpretation of their semantics (i.e., the interpretation of the data considering aspects such as the purpose of each activity, whether is relevant, in what quantities it is measured, etc.). The second challenge is related to the correct interpretation of the control-flow semantics of a process model when it comes to classifying the decisions that have been made. This challenge arise problems related with:

Invisible activities: activities that have no correspondence in the log (e.g. they can be added for routing purposes only). Figure 30 shows a fragment of a process model. This fragment contains one variation point with two alternatives, which starts with invisible activities. As these activities have no correspondence in the log, it is not possible to classify the choices (or rules) related with the decision point in question.

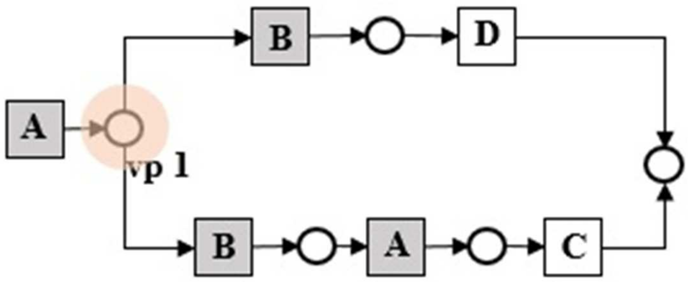
Figure 30 - Example of invisible activities



One way to solve this problem, is to verify the next visible activities that were performed after the invisible activity until the next join construct is encountered. The analysis of these activities can indicate the rules for selecting each alternative.

Duplicate activities: multiple activities that have the same log event associated, which means that their occurrences cannot be distinguished in the log. Duplicate activities (highlighted in grey colour in Figure 31) have an associated log event, however its occurrence cannot be used to classify the possible choices related to a decision point as it could also stem from another activity.

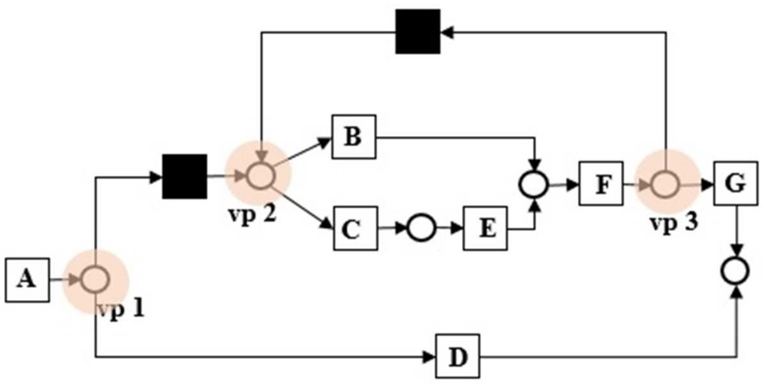
Figure 31 - Example of duplicate activities



The solution for duplicate activities is the same of the invisible activities, i.e., to trace the activities performed after the duplicate activities until an unambiguous activity or a join constructor.

Loops: need to be correctly interpreted. Might be necessary to analyse different event log to understand the occurrence and non-occurrence of activities related with loops. Decision points can be involved with loops in three different ways as shown in Figure 32:

Figure 32 - Example of loops in a process model



- *Decision points contained in a loop (vp2)*: multiple occurrences of a decision related to a decision point may occur per process instance;

- *Decision points containing a loop (vp1)*: although a process instance may contain multiple occurrences of activity B and C, only the first occurrence of either of them indicates a choice related to this decision point;
- *Decision points that are loops (vp3)*: each occurrence of either B or C except the first occurrence must be related to this decision point.

The analysis of the process model, shows two types of decision points: mandatory and optional. According to Bühne, Halmans and Pohl (2003), a variation point is mandatory if minimum one of the related variants is selected. An optional variation point enables the execution of some activities. However, in this case, no process variant is selected in the variation point. Mandatory variation points inherit optional variation points. Thus, a selection in a mandatory variation point enables the selection of the related optional variation points. Figure 33 shows an example of a mandatory and an optional variation point.

Figure 33 - Example of a mandatory and an optional variation point

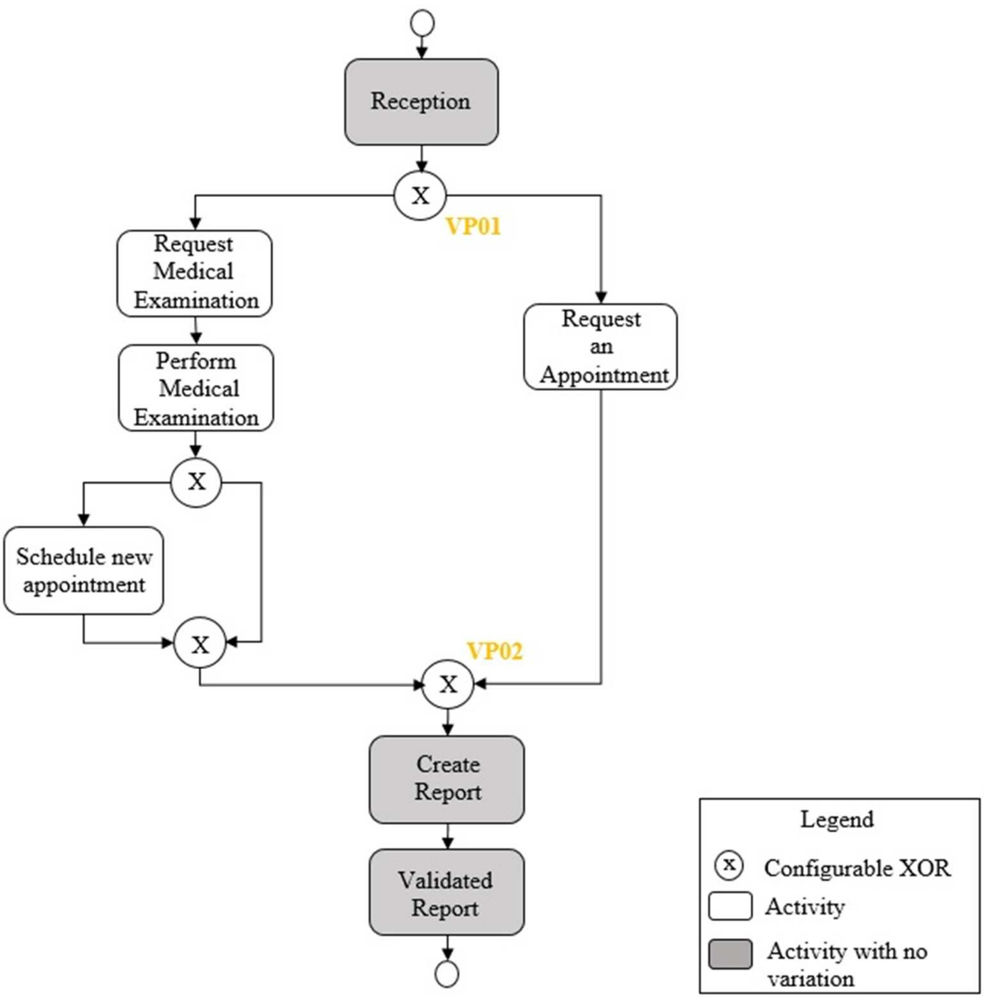


Figure 33 shows that the first variation point (VP01) define the selection of a process variant. However, the second variation point define the execution or not of the activity 'Scheduling new appointment'. The selection on the second variation point relies on the selection of the first variation point. The information about the dependency between the variation points is useful for the development of the next step, which refers to configuring process variants to meet specific end-user requirements. The questionnaire-model approach is applied to support process variant configuration.

4.3. DEVELOPING THE QUESTIONNAIRE-MODEL APPROACH

The decision mining enables to discover the variation points, the alternatives for them and the rules for selecting the process variants. The rules for selecting a process variant are related to the requirements related with the business context and, thus, these requirements can be provided by the user.

The questionnaire-model approach discussed in Section 3.4.1, proposes to individualize a process model according to the answers to questions about the respective deployment context. As mentioned previously, in this approach, each question refers to a variation point, and each fact corresponds to a Boolean variable representing a feature of the domain. Thus, when a feature is selected, i.e., it is enabled, the other alternatives related with the same variation point are disabled.

In this way, the variation points, the alternatives for them, as well the respective rules, are necessary to develop the questionnaire. Thus, the results obtained through decision mining analysis guide the development of the questionnaire, as shown in the Figure 34.

Figure 34 - Development of the questionnaire based on the decision tree

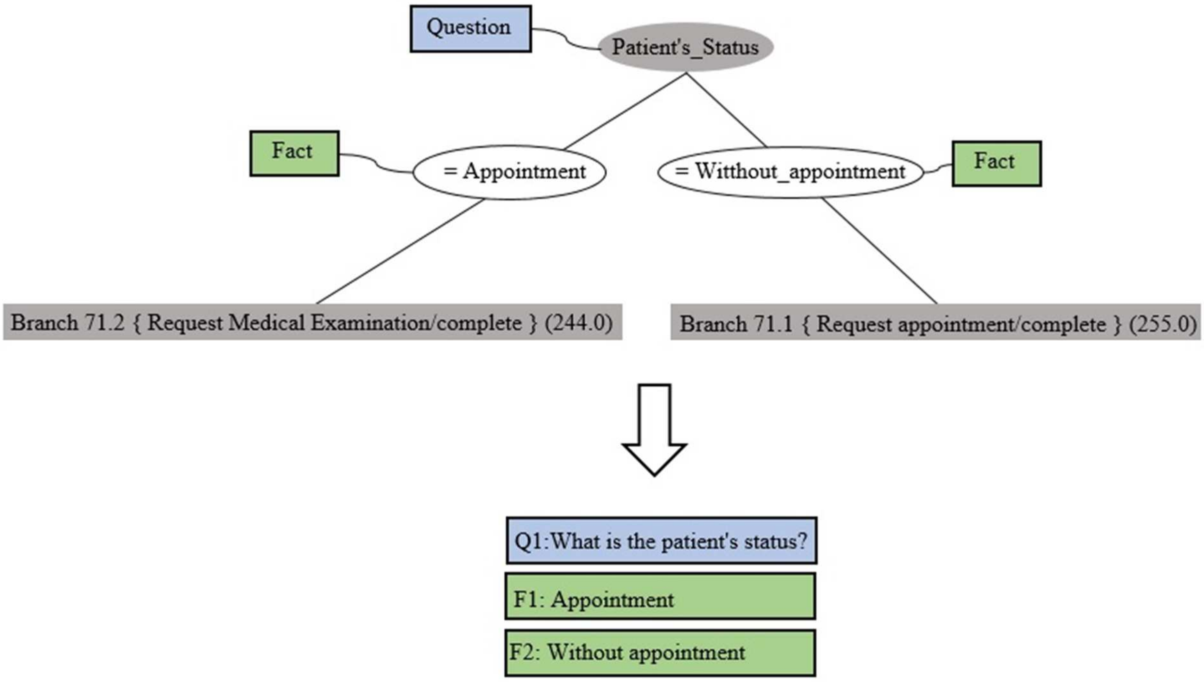


Figure 34 shows that the root from a decision tree is related with a question in the questionnaire and the branches of the decision tree are related with the domain facts. Besides, the decision tree also shows the activity that must be performed according to each branch. Thus, the decision tree also can be used to develop the next step of the proposed framework, which is related with the development of the ontologies.

4.4. USING ONTOLOGIES FOR PROCESS MODEL CUSTOMIZATION

It is worth mentioning that building ontologies is needed but is not the focus of this research. The focus is using ontologies, whose were built by experts in ontology engineering. According to the framework proposed in Section 4.1, one ontology is based on the internal and external regulations and the expert knowledge about the business context. Another ontology is related to the variation points of the process model. The ontologies are merged into one ontology.

Based on the questionnaire developed in step 2, the user selects a choice in the first variation point, then by reasoning on the ontology the next alternative is selected. Besides, based on the information provided by the user, recommendations about the process model and recommendations about the business context can be provided for the user. Thus, as the user select choices in the questionnaire and reason on the ontology, the process model is customized.

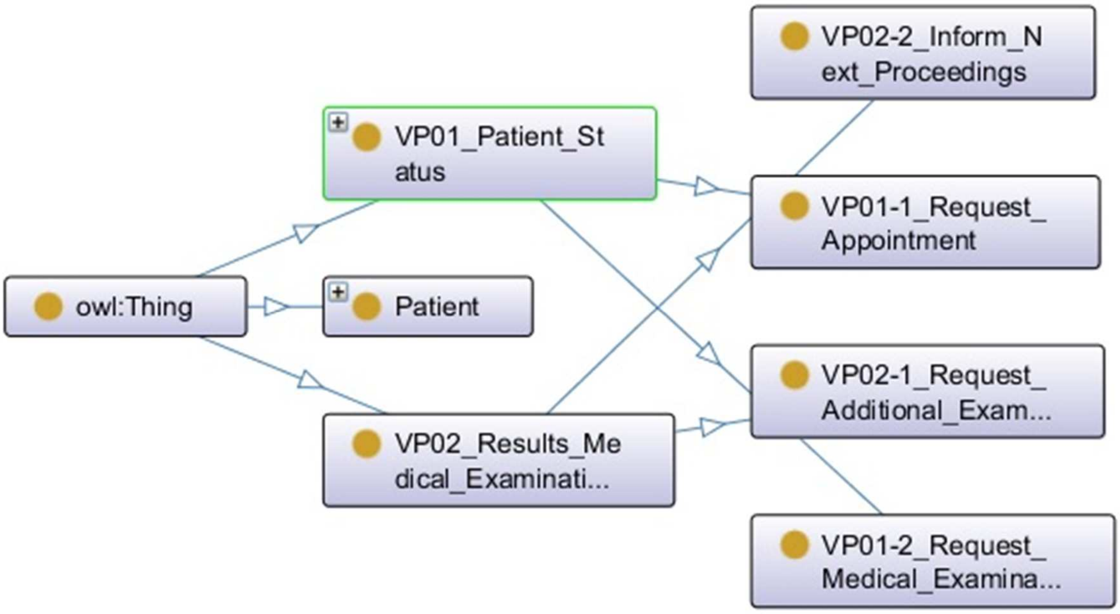
The decision trees are used for the development of the ontology related with the variation points in the following way:

Table 5 - Relation between a decision tree and an ontology

Elements in the Decision Tree	Elements in the Ontology
Branch	Data properties
Leaf node	Classes

According to Table 5, each leaf node is a class in the ontology and the branches are defined as data properties, whose values need to be set by the user. Figure 35 shows the ontology related with the two variation points of the process model for handling medical examinations.

Figure 35 - Ontology based in the variation points



The ontologies were developed in the software Protégé, which is a free, open source ontology editor, developed by the Stanford Center for Biomedical Informatics Research at the Stanford University School of Medicine. Protégé fully support the OWL 2 Web Ontology Language and RDF specifications from the World Wide Web Consortium (PROTÉGÉ, 2017).

Figure 35 shows the concepts in the ontology which are related to the variation points (VP01_Patient_Status and VP02_Results_Medical_Examination). These concepts are the root in the decision trees. The subsumed concepts are the lead nodes of the decision tree, i.e., the alternatives available for each variation point. The branches in the decision tree are defined as the data properties in the ontology. Data properties describe the relationships between the individuals and data values.

The concepts and properties are related with the different proceedings performed in order to provide medical treatment for the patients. Thus, the patients are the individuals in the ontology, i.e., they are the objects in the domain that we are interested. For example, the data property related with the first variation point ‘VP01_Patient_Status is ‘Appointment’, which can be set as true or false, i.e., the patient can have an appointment or can request one.

As can be noted, the data properties correspond to the facts in the questionnaire. Thus, answering the questionnaire, corresponds to the definition of the data properties in the ontology. The relationships between the different elements in the ontology are defined in terms of the Semantic Web Rule Language (SWRL). SWRL is based on a combination of the OWL DL and OWL Lite sublanguages of the OWL Web Ontology Language. It allows users to write Horn-like rules to reason about OWL individuals and to infer new knowledge about those individuals (HORROCKS et al., 2004).

Horn-like rule is a definite Horn clause in which all the prepositions are in the form of RDF triples and only allow variables in subject position and object position (WU et al., 2014).

$$HLR = \{\{\text{antecedent } 1, \ldots \text{ antecedent } n\}, \{\text{consequence}\}\}$$

The proposed rules are of the form of an implication between an antecedent (body) and consequent (head). The intended meaning can be read as: whenever the conditions specified in the antecedent hold, then the conditions specified in the consequent must also hold. SWRL does not support negated atoms or disjunction. On the other side, it supports built-in, binding, sameAs and differentFrom clauses, OWL restrictions, among others (HORROCKS et al., 2004). Regarding the ontology in Figure 36, the following SWRL rules were defined:

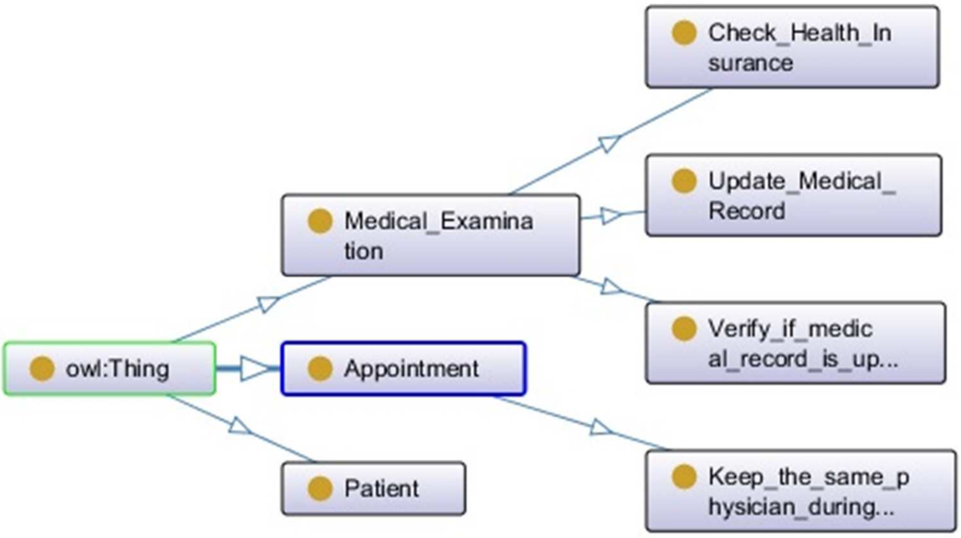
Figure 36 - SWRL rules

Name	Rule
S1	Patient(?s) ^ VP01_Patient_Status(?p) -> hasStatus(?s, ?p)
S2	Patient(?s) ^ hasStatus(?s, ?p) ^ VP01_Patient_Status(?p) ^ Appointment(?p, true) -> VP01-2_Request_Medical_Examination(?p)
S3	Patient(?s) ^ hasStatus(?s, ?p) ^ VP01_Patient_Status(?p) ^ Appointment(?p, false) -> VP01-1_Request_Appointment(?p)
S4	Patient(?s) ^ VP01-2_Request_Medical_Examination(?p) -> hasResults(?s, ?p)
S5	Patient(?s) ^ hasResults(?s, ?p) ^ VP01-2_Request_Medical_Examination(?p) ^ AdditionalExams(?p, true) -> VP02-1_Request_Additional_Exams(?p)
S6	Patient(?s) ^ hasResults(?s, ?p) ^ VP01-2_Request_Medical_Examination(?p) ^ Diagnostic(?p, true) -> VP02-2_Inform_Next_Proceedings(?p)

The rules in Figure 36, shows three elements defined as data properties: 'Appointment', which can be set as true or false; 'Additional exams' and 'Diagnostic'. For example, Rule S2, states that if a patient has the status 'Appointment' set as true, then the next activity to be performed is VP01-2_Request_Medical_Examination. Otherwise, i.e., 'Appointment is set as false, the rule S3 states that the activity performed is VP01-1_Request_Appointment.

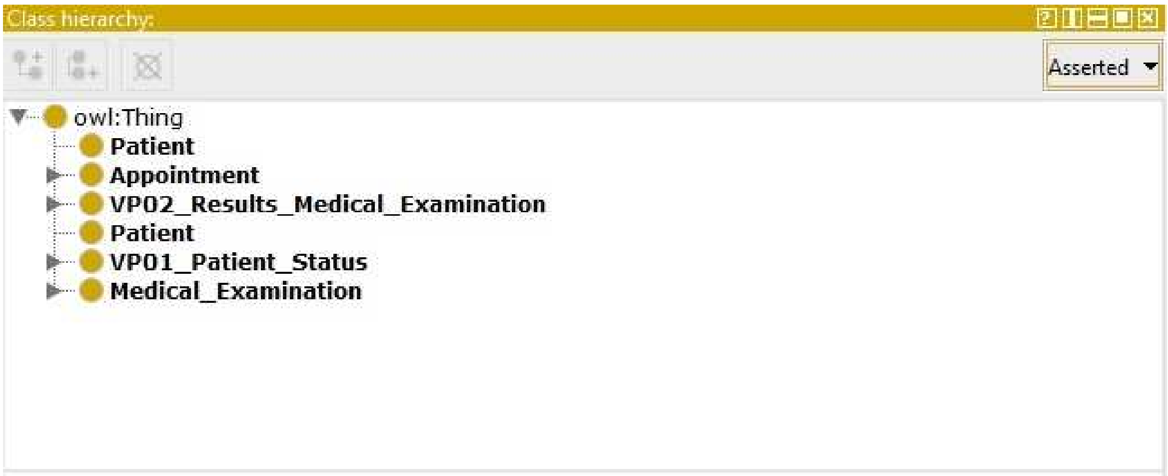
The second ontology is based on the regulations of the business context. Figure 37 shows an ontology created based on four recommendations:

Figure 37 - Ontology based on the regulations



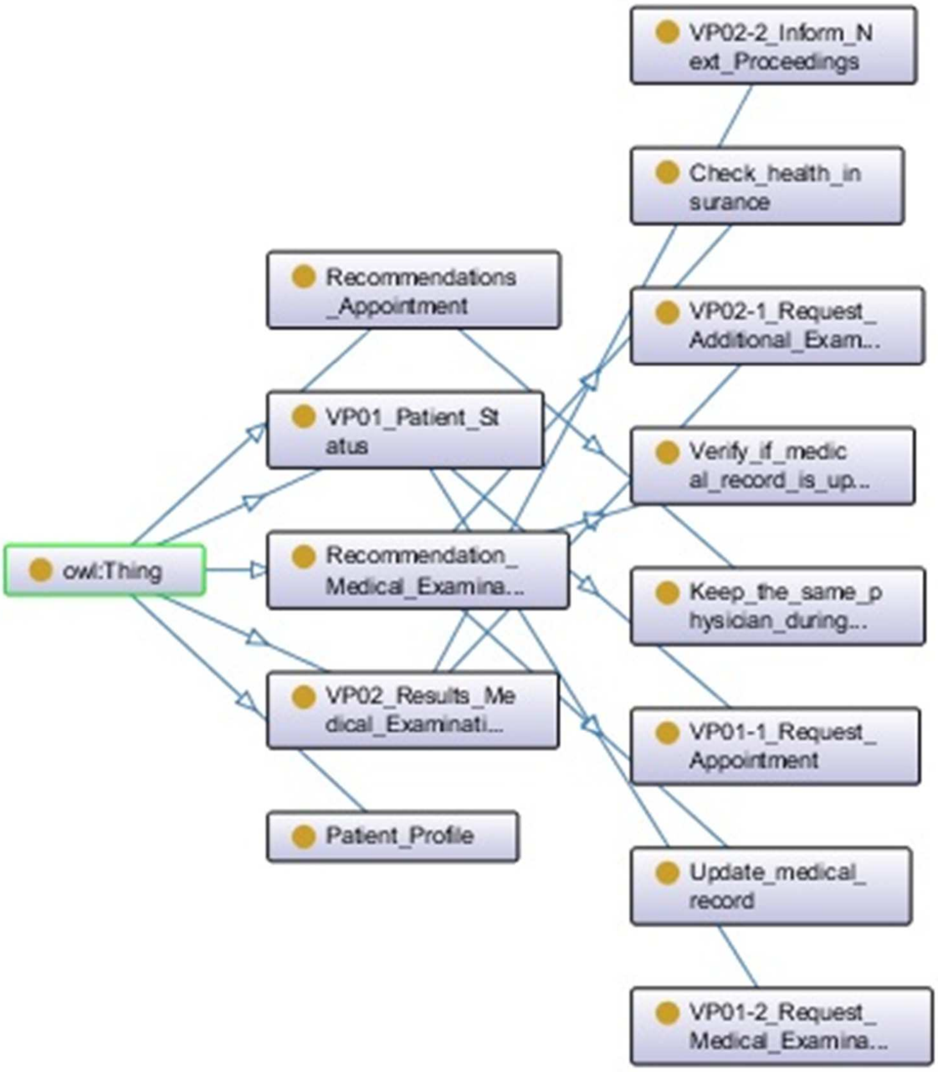
Then, both ontologies were merged in Protégé. As mentioned in Section 2.3.4, merging ontologies is related to building a new ontology by unifying knowledge from other ontologies. Protégé enables to merge ontologies in a new ontology or in an existing ontology. After merging the ontologies, it is necessary to check for inconsistencies. Figure 38 depicted the concepts in the new ontology.

Figure 38 - Concepts in the merged ontology



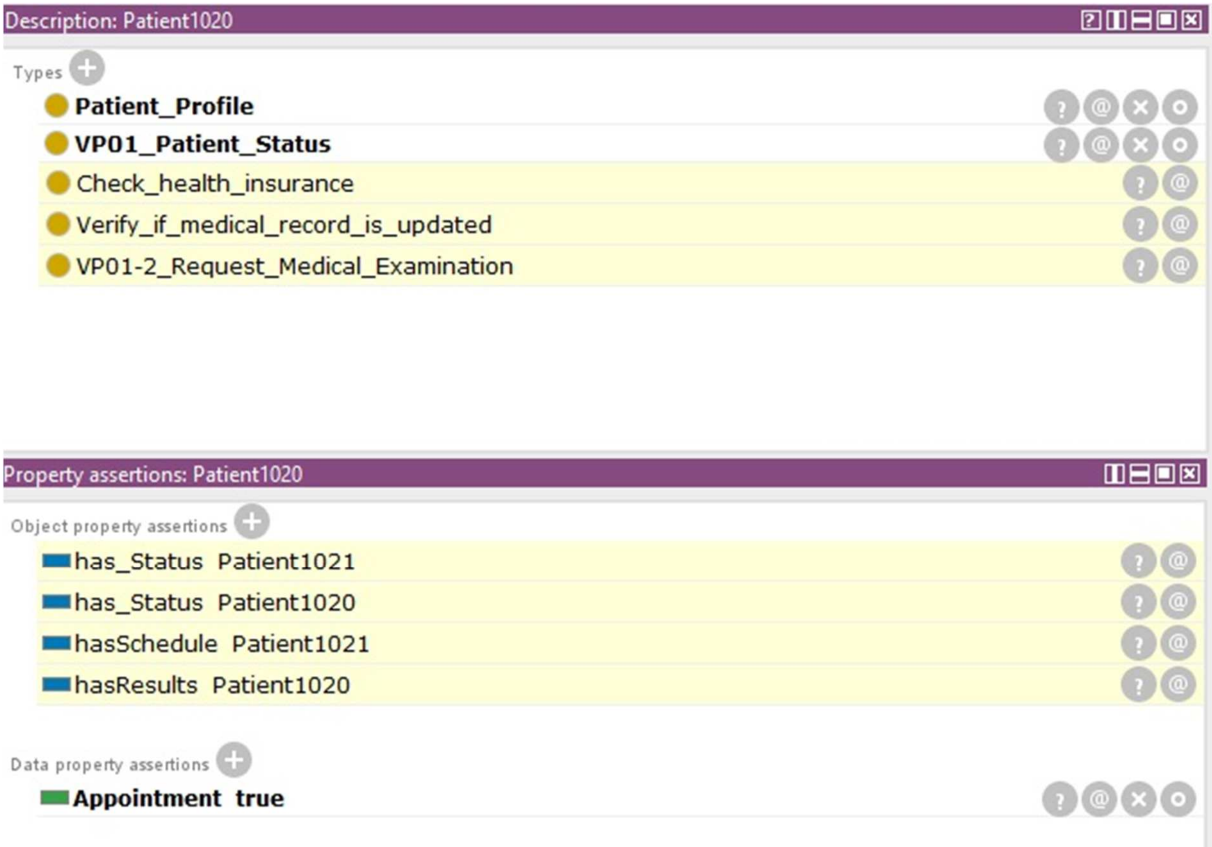
As shown in Figure 38, both ontologies have a concept named Patient. As result, the unique name assumption is violated. In addition, one ontology contains a concept named ‘Appointment’, and the other ontology contains a data property named ‘Appointment’, which is also a violation. As the merged ontologies are developed in the same domain, it may exist similarities or intersections between the concepts. Thus, it is necessary to analyse the concepts in order to eliminate the inconsistencies of the ontology. The concept ‘Appointment’ was renamed as ‘Recommendation_Appointment’. The concept ‘Patient’ can be changed to one concept, which is renamed to ‘Patient_Profile’. The object property hasStatus is also present in both ontologies, and then was renamed to has_Status. The new ontology is depicted in Figure 39.

Figure 39 - Merged ontology



Then, some rules SWRL were changed in order to connect the concepts related with the variation points and the recommendations. In this way, when an activity is selected, the recommendations about the path correspondent with the activity in question are also selected. Then, a reasoner is used for analysing the logical conditions and creating the inferences. We used the Pellet reasoner, which is a complete OWL-DL reasoner. Pellet is written in Java, is open source and is a Description Logic reasoner based in tableaux algorithms (SIRIN et al., 2007). An example of the result obtained from the reasoning step is depicted in Figure 40.

Figure 40 - Result of the reasoning step



Thus, based on the information provided by the user ('Appointment' set as true) and after reasoning on the ontology, the next activity to be performed is selected (VP01-2_Request_Medical_Examination). In addition, two recommendations are provided (Check_health_insurance and Verify_if_medical_record_is_updated).

4.5. SYNTHESIS

The framework proposed in this research aims to support the decision-making during the process model customization by providing recommendations about the business context and the activities in the process model. The recommendations are based on the knowledge obtained from internal and/or external regulations, expert knowledge, and the knowledge captured from the process model executions. The framework also provides guidance by means of a questionnaire.

The framework is composed of three steps. In the first step, an event log is analysed by means of process mining techniques. As the event log can be incomplete or contain noise, an approach is proposed to develop a process model based on the event log, the expert knowledge, and the internal and external regulations.

In order to analyse the various scenarios that can be extracted from the developed process model, an event log is created to simulate these scenarios. By applying classical process mining techniques, such as α-algorithm or the heuristic miner, the process model is obtained from the generated event log. These approaches enable to identify the variation points. However, they do not provide any information about the rules for choosing the alternatives for each variation point. In order to discover the data dependencies that affect the routing of a case is applied the decision mining analysis, a process mining technique. In this step, the decision tree concept is used to carry out a decision point analysis, i.e., to find out which properties of a case might lead to taking certain paths in the process.

Based on the decision trees obtained through the decision mining analysis, the questionnaire-model approach is developed in the Step 2. The questionnaire is applied to guide users in providing the information needed for process variant selection. In this approach, each variation point refers to a question, thus the selection of an alternative for a question refers to the selection of the paths available in relation to the respective variation point.

Variations point can be defined as optional or mandatory. The variation points defined as mandatory are related to the selection of process variants, thus they inherit the optional variation points. This definition of the variation points enables to understand the interdependencies between the variation points. In this way, when a selection is made in the mandatory variation points, the related set of optional variation points is enabled. This knowledge is helpful during the definition of the order of dependence on facts and questions in the questionnaire-model approach.

Step 3 refers to the development of the ontologies for process model customization. One ontology formalizes the knowledge related with the variation points. This ontology is developed based on the decision tree obtained through the decision mining analysis. The leaf nodes are defined as concepts in the ontology, which correspond with the alternatives for the variation points. The branches are defined as data properties in the ontology and they corresponds with the facts in the questionnaire.

Other ontology formalizes the knowledge about the internal and/or external regulations and expert knowledge. Both ontologies are merged into one ontology. Thus, the resulting ontology contain the knowledge about the business context and the process model. SWRL rules define the relations between the various elements in the ontology. Thus, when a fact is selected, the corresponding data property is enabled, then by reasoning on the ontology, the alternatives related with the business process and the recommendations about the business context are provided for the user.

5. CASE STUDY

This chapter presents the application of the proposed approach for customizing process variants related to the treatment of patients diagnosed with acute ischemic stroke. Thus, the first section presents the symptoms and the treatments provided for patients diagnosed with ischemic stroke. The second section presents the development of each step of the approach for process model customization. The last section discusses the application of the proposed approach.

5.1. CASE DESCRIPTION

The case study proposed to evaluate the framework is related with the treatment provided for patients diagnosed with acute ischemic stroke. A stroke happens when the blood supplied to the part of the brain is cut off. Without blood, brain cells can be damaged or die. A stroke caused by lack of blood reaching part of the brain is called an ischemic stroke (WORLD STROKE ORGANIZATION, 2017).

According to the Global Health Observatory (2017), among the 56.4 million deaths worldwide in 2015, strokes are responsible for approximately 6.24 million of the deaths. Ischaemic heart disease and stroke are the world’s biggest killers, accounting for a combined 15 million deaths in 2015. These diseases have remained the leading causes of death globally in the last 15 years.

Despite being responsible for a large share of the world's mortality, many patients survive to stroke, but the resulting sequelae impacts on functional capacity and quality of life, causing great impact on health systems. However, there are ways to significantly reduce its impact such as recognizing the signs of stroke early, treating it as a medical emergency with admission to a specialized stroke unit. Accessing to the best professional care can substantially improve outcomes (WORLD STROKE ORGANIZATION, 2017).

There are some types of treatments that can be provided for patients diagnosed with acute ischemic stroke according to several criteria. These criteria include the onset of symptoms, the patient’s age, the patient medical record (e.g. surgery, previously stroke, medication used regularly for the patient, among others). According to Martins et al., 2012:

“One of the treatments is the administration of intravenous recombinant tissue plasminogen activator (rt-PA), which is a clot busting drug that only can be provided within 4 hours and 30 minutes after the stroke. Another treatment is the intra-arterial approach which may provide some advantages, such as increased concentration of the thrombolytic agent at the site of occlusion, the greater time required to begin the intra-arterial procedure, among others. The protocol for combined (intravenous and intra-arterial) thrombolysis is the treatment combining the ease of administration and speed of intravenous thrombolytic therapy and the higher recanalization rates and potentially superior outcomes of its intra-arterial counterpart. Finally, there is the mechanical thrombolysis treatment which is related to removing the clot with a stent".

5.2. CONDUCTING THE CASE STUDY

For the evaluation of the framework, we obtained an event log related to the treatment of patients diagnosed with acute ischemic stroke from a Brazilian hospital. The hospital only provides the thrombolysis therapy treatment. If this treatment is not appropriate, the patient is monitored or transported to another hospital. The event log is composed by several information as shown in Figure 41.

Figure 41 - Information obtained in the event log related with the treatment of patients diagnosed with acute ischemic stroke

Event log's information

* Patient's ID
* Onset of symptoms
* Time the patient receive help
* Arrival at hospital
* Patient's gender
* Patient's age
* Exams results: cholesterol, creatinine, etc.
* Patient medical record: previously stroke, diabetes, smoker, etc.
* Stroke scales: Banford, NIH, Rankin, Barthel.

An ID is provided to ensure the patient's anonymity. The event log contains the patient's age, gender and the timestamp related with the onset of symptoms, the time that the patient received help and the time that the patient arrived at the hospital. Information about the exams performed by the patient are also available. These exams are related to: cholesterol levels, creatinine, blood sugar levels, creatinine, uric acid, electrocardiogram, among others.

Information about the patient's medical record are also available in the event log such as previously stroke, surgery in the last months, medication used by the patient, among others.

The event log also shows if the patient performed or not the thrombolytic treatment. Finally, the event log contains four stroke scales:

- *Bamford Scale*: developed during the Oxfordshire Community Stroke Project (OCSP), this classification defines four subtypes of cerebral infarction: TACS, indicates total anterior circulation syndrome; PACS, partial anterior circulation syndrome; LACS, lacunar syndrome; and POCS, posterior circulation syndrome (LINDLEY et al., 1993).
- *Modified Rankin Scale*: consists of six levels of classification that describe the degree of disability in stroke survivors: 0, no symptoms; 1, no significant disability; 2, slight disability; 3, moderate disability; 4, moderately severe disability; 5, severe disability and 6, death (FISH, 2011).
- *Barthel Scale*: measures disability or dependence in activities of daily living in stroke patients. The items can be divided into a group that is related to self-care (feeding, grooming, bathing, dressing, bowel and bladder care, and toilet use) and a group related to mobility (mobility, transfers, and stair climbing). The maximal score is 100, if 5-point increments are used, indicating that the patient is fully independent in physical functioning. The lowest score is 0, representing a totally dependent bedridden state (SULTER, STEEN and DE KEYSER, 1999).
- *National Institutes of Health Stroke Scale (NIHSS or NIH):* is a systematic assessment tool that provides a quantitative measure of stroke-related neurologic deficit. The NIHSS is used as a clinical assessment tool to evaluate acuity of stroke patients, determine appropriate treatment, and predict patient outcome (NIH STROKE SCALE, 2017). The NIHSS contains 15 items, including level of consciousness, eye movement, visual field deficit, and motor and sensory involvement. Scale items are scored by degree of severity using weighted scores (LYDEN et al., 1999).

Many of the information contained in the event log (e.g., onset of symptoms, previously stroke or surgery in the last three months, among others) are necessary for the selection of the appropriate treatment. However, the event log does not contain all the activities performed during the patient’s treatment. In fact, the activities are only related with the exams performed during the treatment, and the treatment provided for the patient, but not the activities performed during the selected treatment.

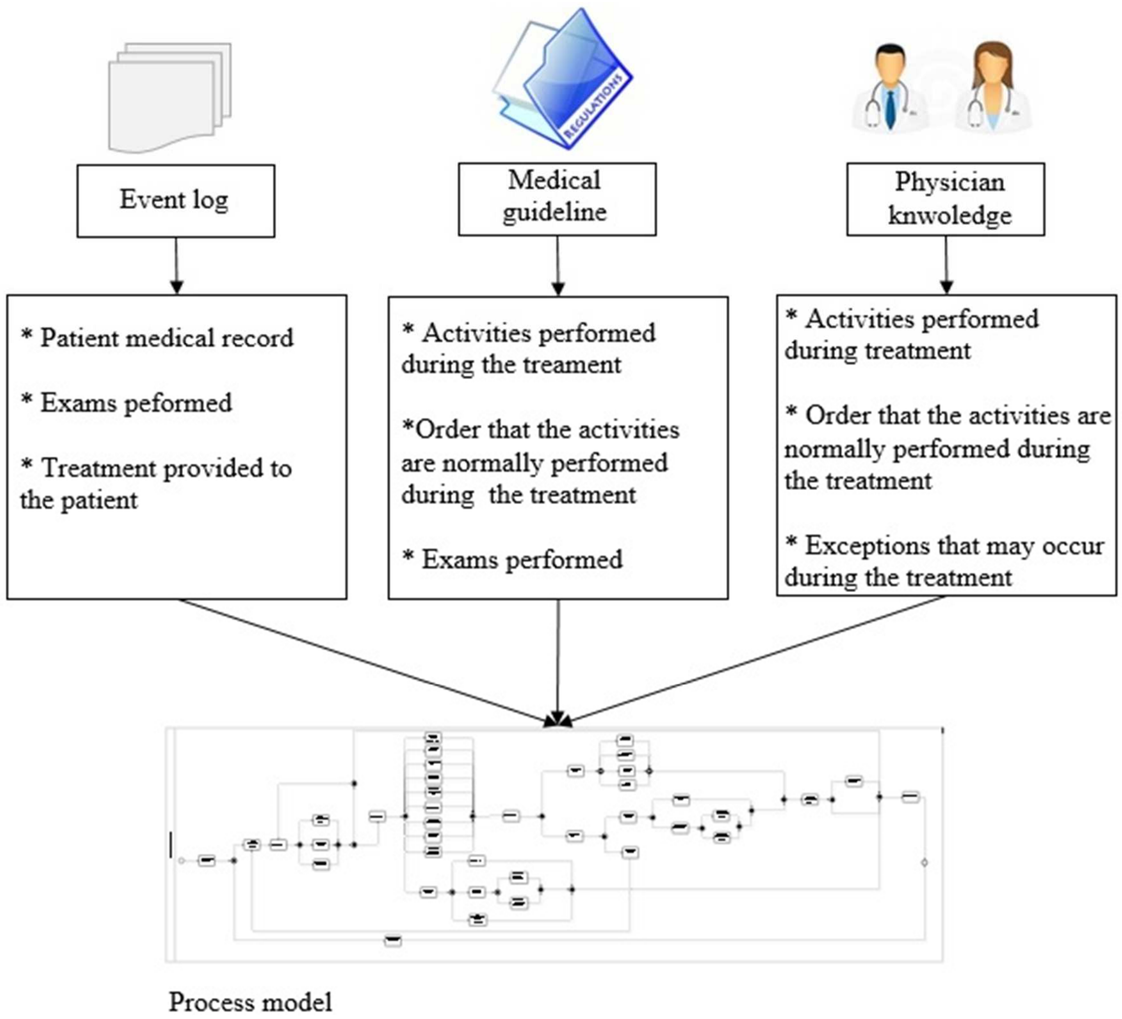
As result, is not possible to analyse the process model directly from the log. Thus, as proposed in the framework for process model configuration, a process model should be developed regarding the event log, the external knowledge such as the Brazilian guideline for the ischemic stroke (OLIVEIRA et al., 2012; MARTINS et al., 2012) and the physician knowledge.

Clinical guidelines are statements that include recommendations intended to optimize patient care, improve the quality of care, limit unjustified practice variations and reduce healthcare costs (KAYMAK et al., 2012). The guideline for acute ischemic stroke was developed in 2012 through several meetings of the Brazilian Stroke Society, which represents the Scientific Department in cerebrovascular diseases of the Brazilian Academy of Neurology, responsible for technical opinions and educational projects related to cerebrovascular diseases. The developed guideline aims to guide specialists and non-specialists in stroke care in managing patients with acute ischemic stroke (OLIVEIRA et al., 2012; MARTINS et al., 2012).

However, no guideline can represent all the situations that may happen during the treatment. Usually the guideline represents only the treatment for an ‘average’ patient (QUAGLINI, 2008). Thus, the expert knowledge enriches the knowledge about the treatment, aiming to represent situations that may not be present in the guideline.

Figure 42 shows the information captured from the event log, the clinical guideline, and the physician knowledge for the development of the process model.

Figure 42 - Information for developing the process model



Due the complexity of the treatment for ischemic stroke, the case study focused in the intravenous protocol treatment. Thus, from the event log is obtained the information about the patient medical record, the exams performed during the treatment, and if the thrombolytic treatment was provided for the patient. The event log can show how the symptoms presented by the patient are related with the treatment provided.

From the medical guideline, we captured the activities and the exams that need to be performed during the treatment. Thus, the medical guideline helps to understand how the symptoms presented by the patient are related with the exams and the activities performed during the treatment. The expert knowledge helps to understand the proceedings realized in the hospital.

The event log analysis and the clinical guideline help to identify the differences and commonalities between them. Thus, possible improvements in the activities performed in the hospital can be identified. The expert knowledge also helps to understand the order that the activities are performed during the treatment. Figure 43 shows the process model developed based on the event log, expert knowledge, and the clinical guidelines.

Figure 43 - Process model developed based on the event log, expert knowledge and the clinical guideline

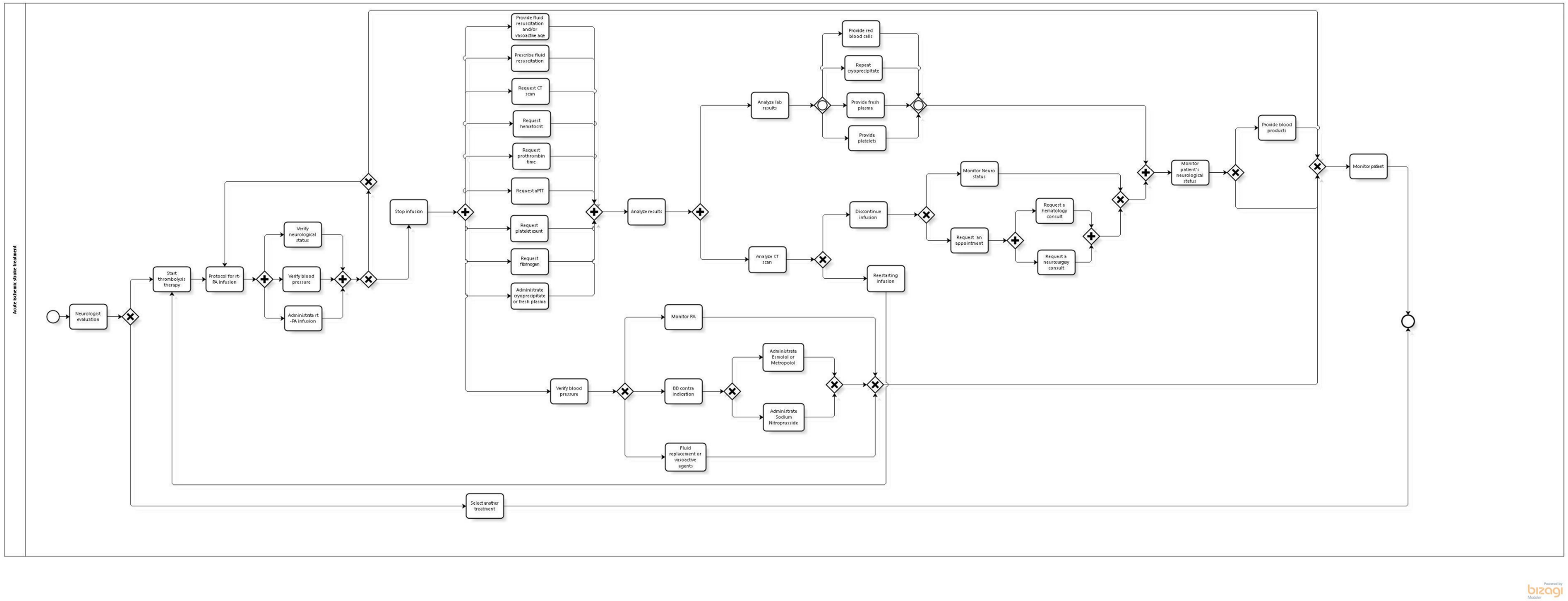
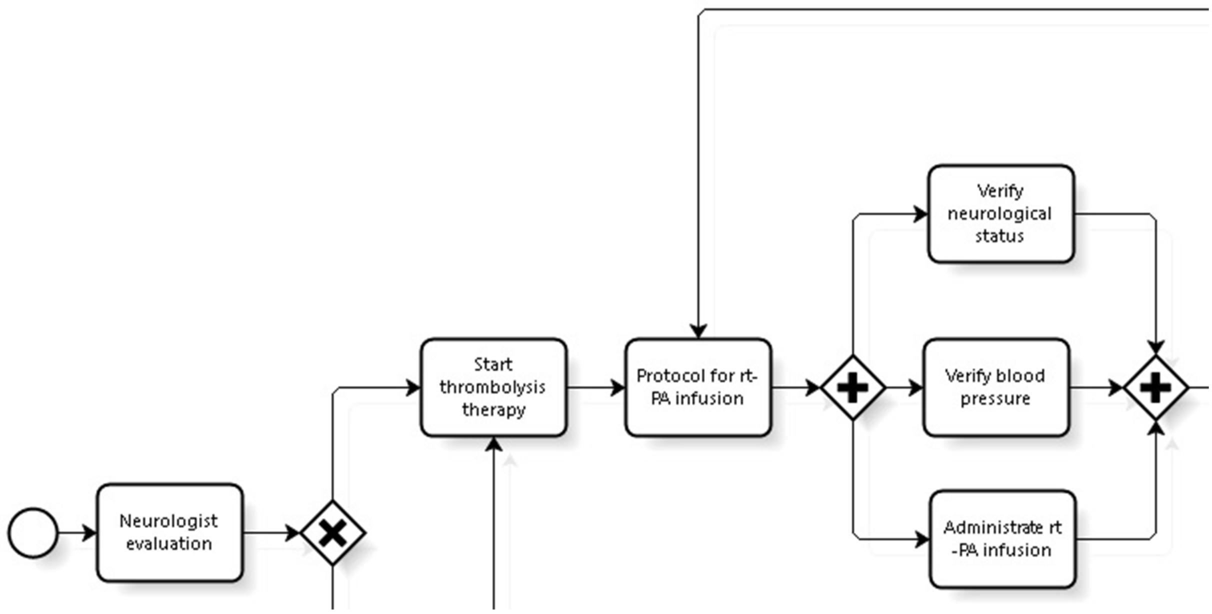


Figure 44 shows an excerpt of the process model generated by the framework. According to the clinical guideline, a neurologist should evaluate the appropriate treatment according to the symptoms presented by the patient and the patient's medical record. The selection of the thrombolysis treatment relies on 4 inclusion criteria and 16 exclusion criteria. These criteria are related with different aspects, such as the symptoms presented by the patient, medication used by the patient, medical record, among others. Thus, the thrombolysis treatment is indicated only if the patient presents all the inclusion criteria and no exclusion criteria.

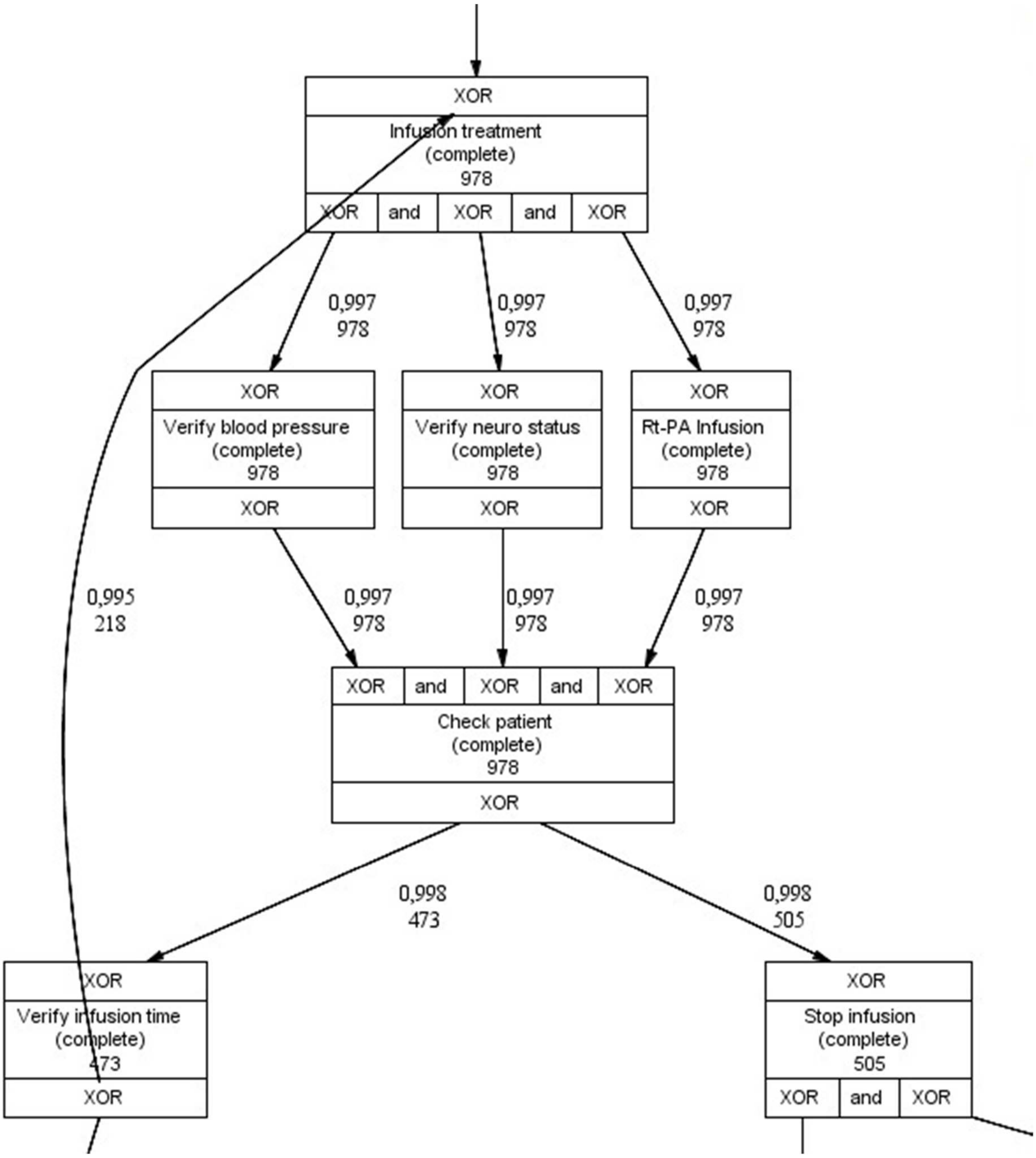
Figure 44 - Excerpt of the process model developed based on the event log, expert knowledge, and clinical guideline



Once the protocol for rt-PA infusion starts, three activities are performed in parallel, the administration of the rt-PA infusion, the verification of the neurological status, and the blood pressure presented by the patient, which should be verified every 15 minutes. If the patient presents a symptom of haemorrhagic complications, the infusion is interrupted and many other activities, including medication, lab exams, neuroimaging techniques are performed to stabilize the patient. Otherwise, the infusion continues to be administrated for approximately 90 minutes and then the infusion is finalized. If the patient presents haemorrhagic complications, but he is stabilized, the infusion can restart. However, if the patient is not stabilized or if the patient shows evidence of intracranial haemorrhage the infusion should be discontinued.

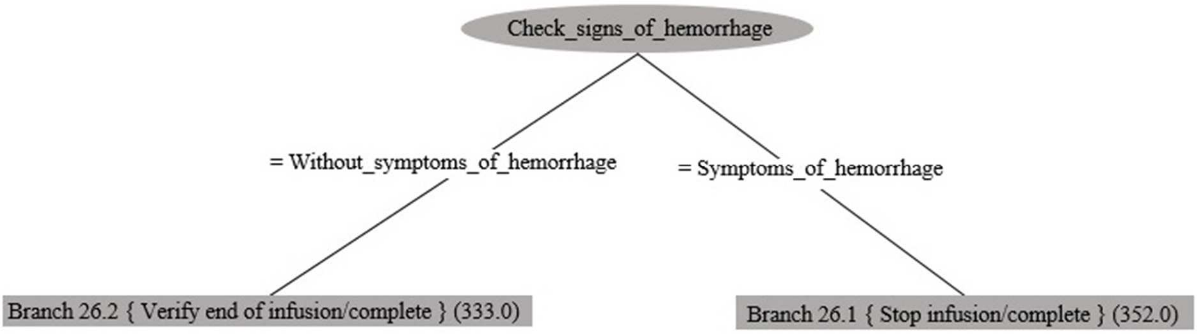
As proposed in the framework, based on the process model developed, a Coloured Petri Net (CPN) model is built, resulting from an event log, which was analysed with ProM. The simulation was made considering 1000 patients. The excerpt of the heuristic miner analysis is depicted in Figure 45.

Figure 45 - Process model obtained by applying the heuristic miner algorithm



As mentioned previously, techniques such as heuristic miner enable to identify the decision points (i.e., XOR-splits) and the alternatives for them. Figure 45 shows that the activity ‘Check patient’ is related to a decision point, where one of two activities can be selected: ‘Stop Infusion’ or ‘Verify infusion time’. However, the heuristic mining cannot show the rules for choosing the available alternatives. Thus, for discovering the rules, the decision mining is applied and the decision tree related to the activity ‘Check patient’ is depicted in Figure 46.

Figure 46 - Decision tree obtained by applying the decision mining analysis



The decision tree shows that if the patient present signs of haemorrhagic complications the infusion is interrupted. Otherwise, it is necessary to verify how long the infusion started. By applying the decision mining analysis, nine decision points were discovered, the alternatives for them and the rules for each alternative are summarized in Table 6.

Table 6 - Variation points, alternatives, and the rules

Decision points	Alternatives	Rules
Select treatment	Start thrombolysis therapy	Only inclusion criteria
	Start another treatment	Exclusion and/or inclusion criteria
Check patient	Verify end of infusion	Haemorrhagic complications
	Stop infusion	Without haemorrhagic complications
Verify end of infusion	Finalize rt-Pa infusion	Time of infusion >= 90 minutes
	Infusion treatment	Time of infusion < 90 minutes
Manage BP (blood pressure)	Monitor changes in blood pressure	Normal blood pressure
	Verify if patient has contra-indication of bb (beta blocker)	Patient with hypertension
	Provide fluid replacement or vasoactive agents	Patient with hypotension
Verify if patient has contra-indication of bb	Administrate Metoprolol or Esmolol	Patient has contra indication of bb
	Administrate Sodium Nitroprusside	Patient has not contra indication of bb

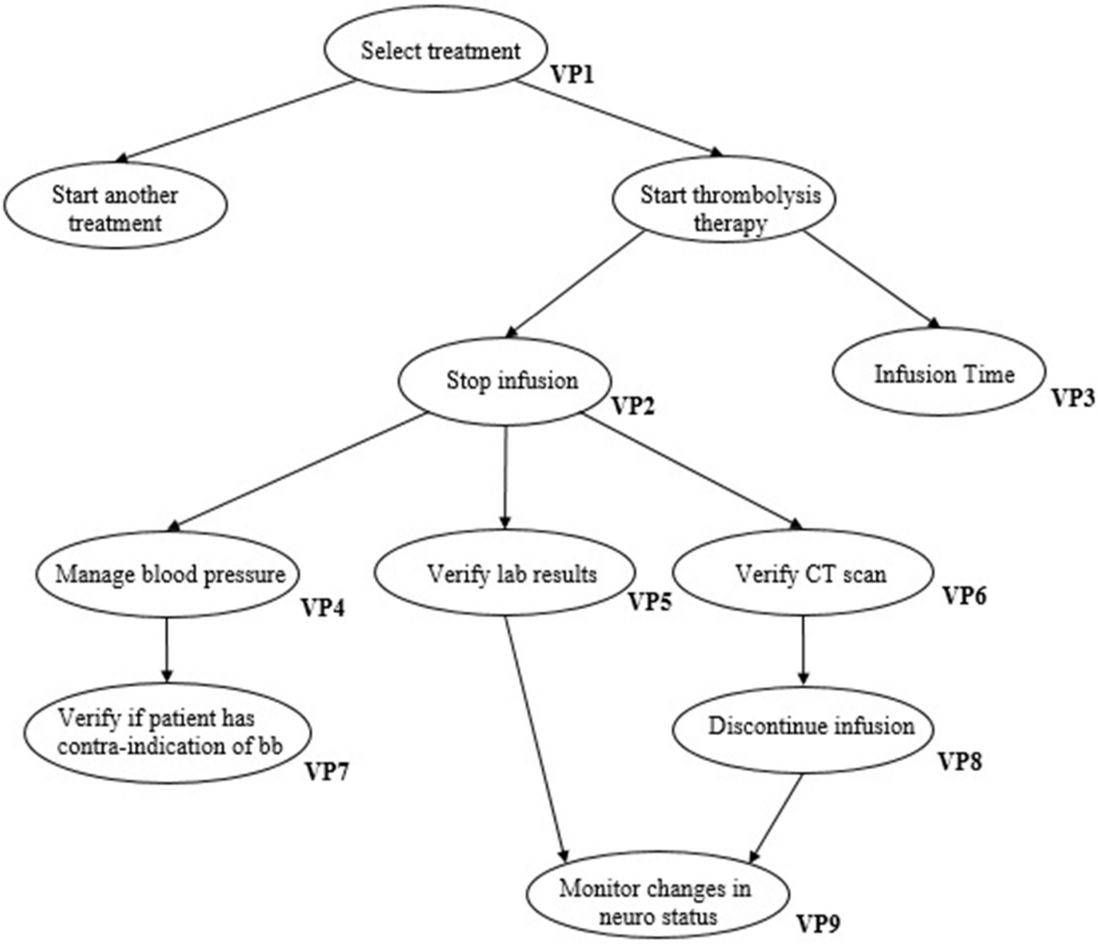
Verify CT scan results	Restarting infusion	No intracranial haemorrhage
	Discontinue infusion	Intracranial haemorrhage
Verify lab results	Provide red cell	Abnormal haematocrit
	Provide platelets	Low platelet count
	Repeat cryo	Low fibrinogen
	Provide fresh plasma	Abnormal PT or aPTT
Discontinue infusion	Monitor neurological status	No evidence of bleeding in the central nervous system
	Request a haematology and a neurosurgery appointment	Evidence of bleeding in the central nervous system
Monitor changes in neuro status	Provide blood products	Clinical status deterioration after 4 to 6 hours
	Monitor patient	No deterioration of the clinical status

Based on table 6, for each alternative related to a variation point a question or a set of questions was formulated. For example, the first variation point is related with the selection of the treatment provided for the patient. As previously mentioned, the selection of the treatment relies on several aspects related to the patient. In the process model, the rules attached with both alternatives related with the treatment selection are simplified to facilitate the understanding of the process model.

However, for the development of the questionnaire, a set of questions were defined based on the inclusion and exclusion criteria. In this way, the questionnaire was developed based on the variation points, the alternatives for them and the rules for choosing each alternative. Thus, for selecting the treatment in the first variation point, a set of questions related to the inclusion and exclusion criteria need to be answered. The clinical guideline is also used for the development of the questionnaire by defining the rules for select a treatment.

Through the decision mining analysis, we can also identify the dependency between the variation points. For example, the activities 'Manage blood pressure', 'Verify lab results', 'Verify CT scan' are dependent of the variation point 'Stop Infusion'. This means that, considering the decision tree (Figure 46), and the dependency among the variation points (Figure 47), it can be concluded that if the patient present haemorrhagic complications, then the activity 'Stop Infusion' is enabled, which also enable the activities 'Manage blood pressure', 'Verify lab results', and 'Verify CT scan'. Figure 47 shows the dependency between the variation points.

Figure 47 - Relation between variation points



The dependency between the variation points and the information from Table 6 are necessary for the development of the ontology based on the variation points. Both guide the development of the rules for configuring the process variants. The concepts and the subsumed concepts in the ontology correspond with the variation points and the alternatives for them respectively as depicted in Figure 48.

Figure 48 - Concepts of the ontology of the variation points

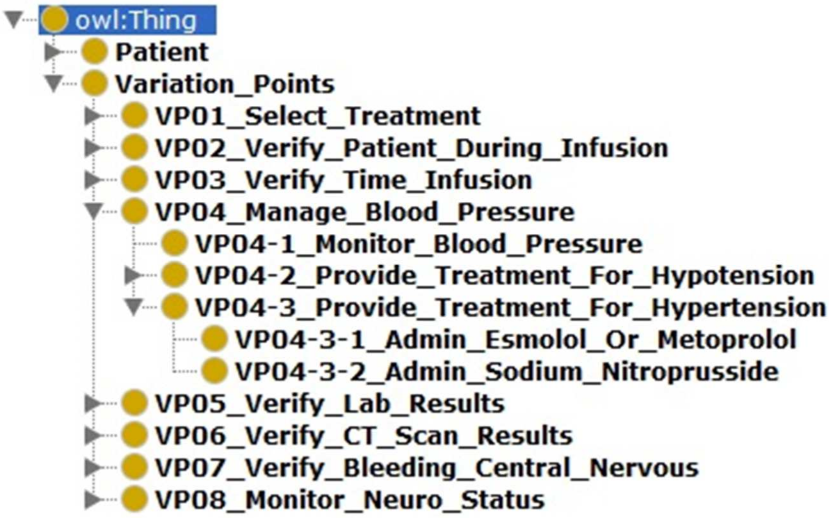


Figure 48 shows the variation points related with the ischemic stroke treatment. Each concept corresponds with a variation point and the subsumed concept 'VP04-3_Provide_Treatment_For_ Hypertension' is a variation point related with the treatment provided to the patient with hypertension. Sodium Nitroprusside is prescribed in case of patients with asthma, heart failure or severe abnormalities in heart function. Otherwise, Esmolol or Metoprolol can be prescribed for the patient. The rules for choosing the available alternatives are developed as SWRL rules as shown in Figure 49. The SWRL rules are also developed considering the dependency between the variation points. Thus, when a variation point is selected, the related variation points can be displayed.

Figure 49 - SWRL rules related with the variation points

SWRL Rules

Patient(?p) ^ VP01_Select_Treatment(?s) -> hasStrokeTreatment(?p, ?s)

Patient(?p) ^ hasStrokeTreatment(?p, ?s) ^ VP01_Select_Treatment(?s) ^ hasInclusionCriteria(?s, true) -> VP01-2_Start_Thrombolysis_Therapy(?p)

Patient(?p) ^ hasStrokeTreatment(?p, ?s) ^ VP01_Select_Treatment(?s) ^ hasExclusionCriteria(?s, true) -> VP01-1_Start_Another_Treatment(?s)

Patient(?p) ^ VP01-2_Start_Thrombolysis_Therapy(?s) -> hasRt-PA_Infusion(?p, ?s)

Patient(?p) ^ hasRt-PA_Infusion(?p, ?s) ^ VP01-2_Start_Thrombolysis_Therapy(?s) -> VP02_Verify_Patient_During_Infusion(?s)

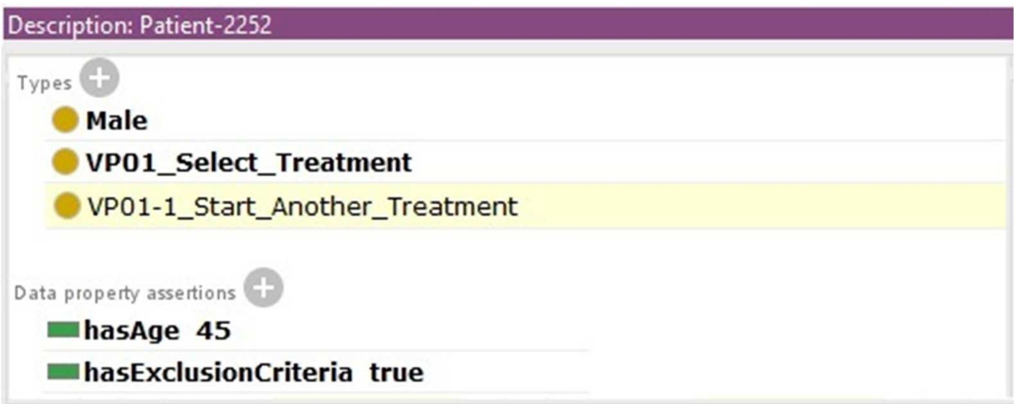
Patient(?p) ^ VP02_Verify_Patient_During_Infusion(?s) -> hasPatientSymptomsVerified(?p, ?s)

Patient(?p) ^ hasPatientSymptomsVerified(?p, ?s) ^ VP02_Verify_Patient_During_Infusion(?s) ^ hasNoBleedingSymptoms(?s, true) -> VP03_Verify_Time_Infusion(?s)

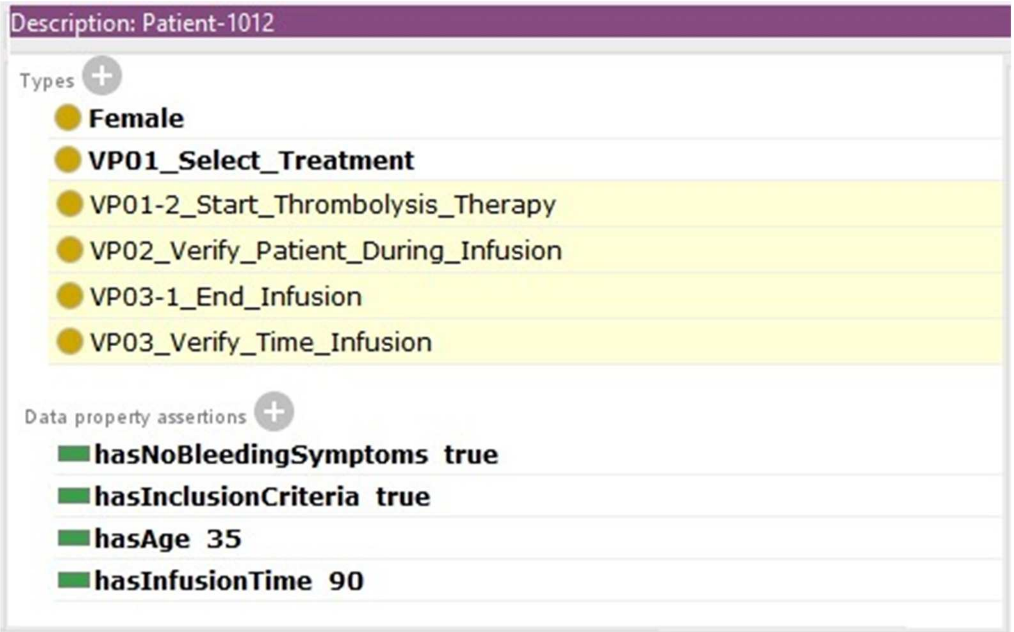
Patient(?p) ^ hasPatientSymptomsVerified(?p, ?s) ^ VP02_Verify_Patient_During_Infusion(?s) ^ hasBleedingSymptoms(?s, true) -> VP02-2_stop_Infusion(?s)

The SWRL rules in Figure 49 is related with two variation points. The first three lines states that if the data property 'hasInclusionCriteria' is set as true, then the rt-PA infusion is selected. However, if the data property 'hasExclusionCriteria' is set as true, another treatment is selected as shown in Figure 50.

Figure 50 - Variation points selection after the reasoning step



(a)

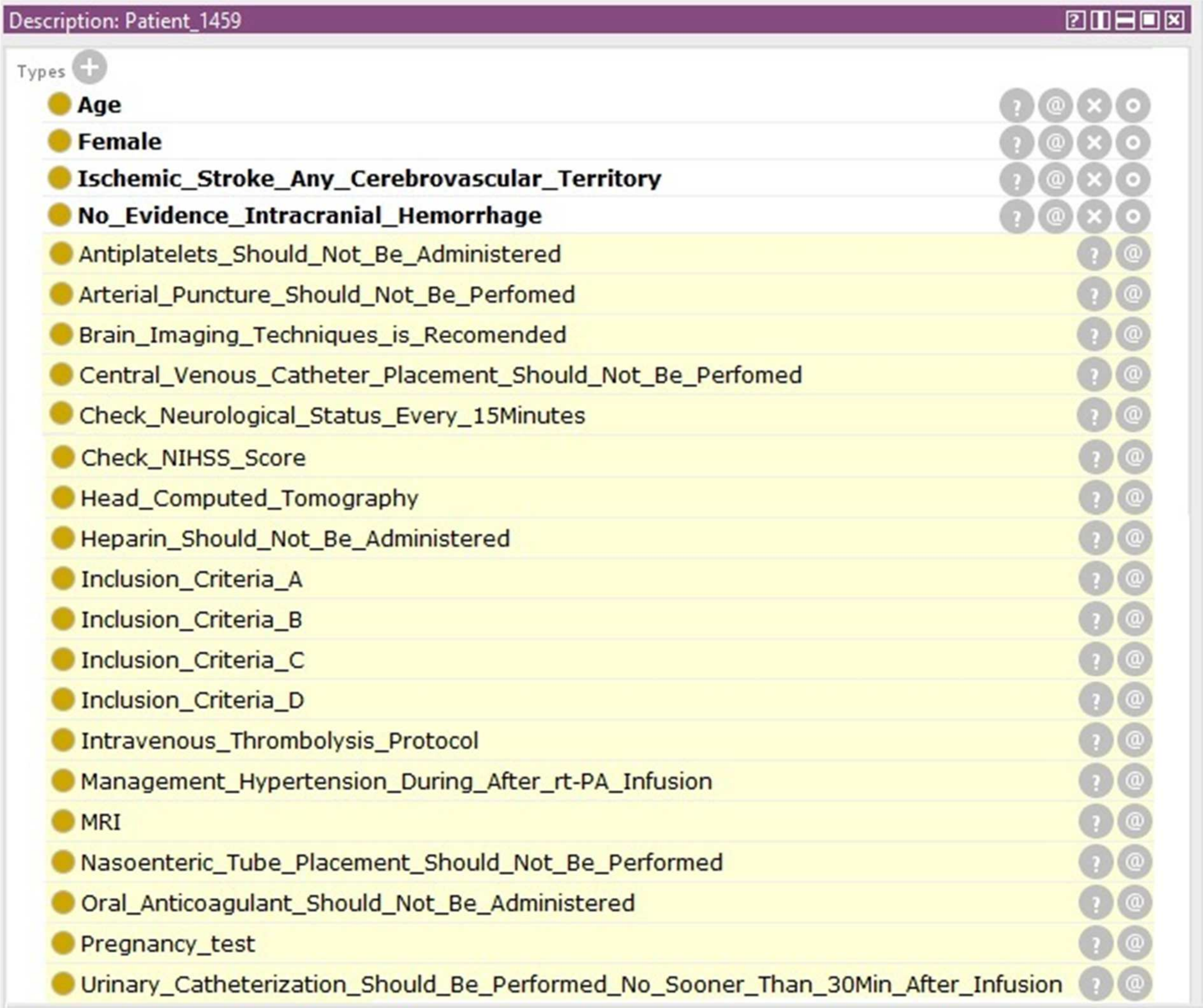


(b)

As depicted in Figure 50(a), patient with ID 2252 present some of the exclusion criteria, thus the data property hasExclusionCriteria is set as true and another treatment is selected. The Figure 50(b) shows that the patient with ID 1012 present only the inclusion criteria. As result, the data property hasInclusionCriteria is set as true and the thrombolysis therapy is selected. Then another variation point is selected 'VP02_Verify_Patient_During_Infusion'. In this variation point, if the patient present signs of bleeding, it is necessary to verify the time of infusion, otherwise, the infusion is interrupted. If the time of infusion is equal or greater than 90 minutes, then the infusion occurred without problems and is finalized.

Another ontology is developed based on the clinical guideline for acute ischemic stroke. As mentioned previously, this ontology is based on the recommendations for the thrombolysis therapy. The Figure 51 shows some recommendations provided by answering questions about the patient's age, gender and symptoms.

Figure 51 - Recommendations obtained after the reasoning step

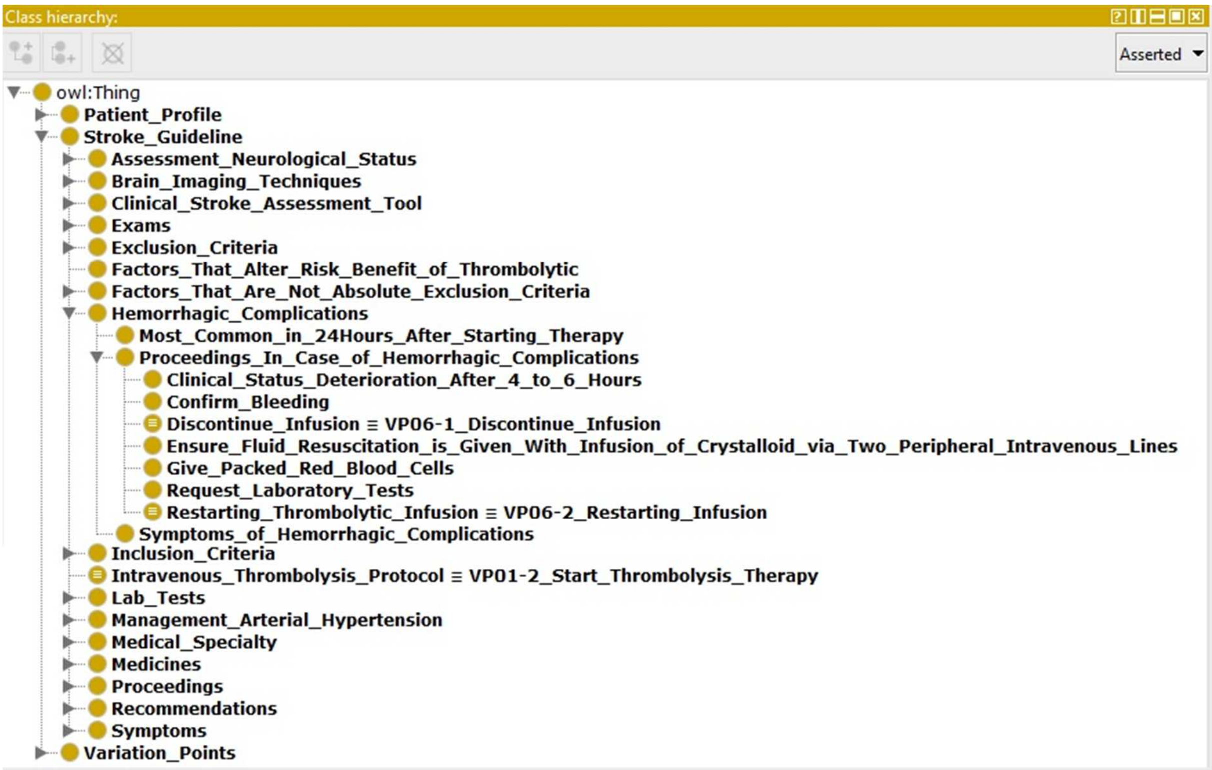


Then, according proposed in the framework for process model customization, both ontologies are merged into one ontology, as shown in Figure 52. In this ontology, we can identify that both ontologies contain the concept 'Patient'. We can also identify concepts that are equivalent such as Intravenous Thrombolysis Protocol and VP01-2_Start_Thrombolysis_Therapy.

Figure 52 - Concepts in the merged ontology

Thus, the obtained ontology needs some adjustments. Concepts with the same label in the merged ontology, may be or may be not equivalent. When concepts with the same label have the same meaning, new classes were defined, such as Patient_Profile, Patient_Age and Patient_Gender, P_Female, P_Male, among others. Besides, classes with different labels, but with the same meaning such as Intravenous_Thrombolysis_Protocol and VP01-2_Start_Thrombolysis_Therapy are defined as equivalent. Some of these adjustments are shown in Figure 53.

Figure 53 - Adjusts performed in the merged ontology



The ontology in Figure 53, shows the concept ‘Patient_Profile’ replacing the concept ‘Patient’. In addition, the concept ‘Intravenous_Thrombolysis_Protocol’ and ‘VP01-2_Start_Thrombolysis_Therapy’ are defined as equivalent, as well the concept ‘Discontinue_Infusion’ and ‘VP06-1_Discontinue_Infusion’ and ‘Restarting_Thrombolytic_Infusion’ and ‘VP06-2_Restarting_Infusion. This ontology is depicted in Appendix B.

In addition, it is needing to verify the SWRL rules because some rules from both ontologies can be integrated. For example, the ontology related to the variation points contains a rule defining that the variation point VP02-2_Stop_Infusion is only selected with the patient present any symptom of haemorrhage. However, as illustrated in Figure 54, we cannot identify which are these symptoms.

Figure 54 - SWRL rule of the ontology related to the variation points

SWRL Rule related to the variation point Stop Infusion
* Patient(?p) ^ hasPatientSymptomsVerified(?p, ?s) ^ VP02_Verify_Patient_During_Infusion(?s) ^ hasBleedingSymptoms(?s, true) -> VP02-2_Stop_Infusion(?s)

On the other side, the ontology related to the guidelines define the symptoms of Haemorrhagic complications such as severe headache, decrease level of consciousness, sudden increase of blood pressure, among others. Figure 55 present these rules.

Figure 55 - SWRL rule of the ontology related to the clinical guideline about the ischemic stroke

SWRL Rules related with symptoms of Haemorrhagic complications

* Patient(?p) ^ hasRt-PA_Infusion(?p, ?s) ^ Intravenous_Thrombolysis_Protocol(?s) ^ hasDiagnostic(?s, ?t) ^ Evidence_Significant_Bleeding(?t) -> Symptoms_of_Haemorrhagic_Complications(?t)

* Patient(?p) ^ hasRt-PA_Infusion(?p, ?s) ^ Intravenous_Thrombolysis_Protocol(?s) ^ hasDiagnostic(?s, ?t) ^ Severe_Headache(?t) -> Symptoms_of_Haemorrhagic_Complications(?t)

* Patient(?p) ^ hasRt-PA_Infusion(?p, ?s) ^ Intravenous_Thrombolysis_Protocol(?s) ^ hasDiagnostic(?s, ?t) ^ Nausea(?t) -> Symptoms_of_Haemorrhagic_Complications(?t)

* Patient(?p) ^ hasRt-PA_Infusion(?p, ?s) ^ Intravenous_Thrombolysis_Protocol(?s) ^ hasDiagnostic(?s, ?t) ^ Vomiting(?t) -> Symptoms_of_Haemorrhagic_Complications(?t)

* Patient(?p) ^ hasRt-PA_Infusion(?p, ?s) ^ Intravenous_Thrombolysis_Protocol(?s) ^ hasDiagnostic(?s, ?t) ^ Decrease_Level_Consciousness(?t) -> Symptoms_of_Haemorrhagic_Complications(?t)

* Patient(?p) ^ hasRt-PA_Infusion(?p, ?s) ^ Intravenous_Thrombolysis_Protocol(?s) ^ hasDiagnostic(?s, ?t) ^ Sudden_Increase_Blood_Pressure(?t) -> Symptoms_of_Haemorrhagic_Complications(?t)

Based on these rules, it is necessary to define new rules that integrated the variation point ‘Stop infusion’ (Figure 54) with the symptoms of haemorrhagic symptoms (Figure 55). Thus, the variation point VP02-2_Stop_Infusion is selected if the patient presents any symptom of haemorrhagic complications. In this way, the recommendations can be provided considering specifically the symptoms presented by the patient. The new rules are shown in Figure 56.

Figure 56 - SWRL rules for selecting the variation point related to haemorrhagic symptoms

SWRL Rules of the merged ontology

* Patient_Profile(?p) ^ hasPatientSymptomsVerified(?p, ?s) ^ VP02_Verify_Patient_During_Infusion(?s) ^ hasDiagnostic(?s, ?t) ^ Severe_Headache(?t) -> VP02-2_Stop_Infusion(?t)

* Patient_Profile(?p) ^ hasPatientSymptomsVerified(?p, ?s) ^ VP02_Verify_Patient_During_Infusion(?s) ^ hasDiagnostic(?s, ?t) ^ Vomiting(?t) -> VP02-2_Stop_Infusion(?t)

* Patient_Profile(?p) ^ hasPatientSymptomsVerified(?p, ?s) ^ VP02_Verify_Patient_During_Infusion(?s) ^ hasDiagnostic(?s, ?t) ^ Decrease_Level_Consciousness(?t) -> VP02-2_Stop_Infusion(?t)

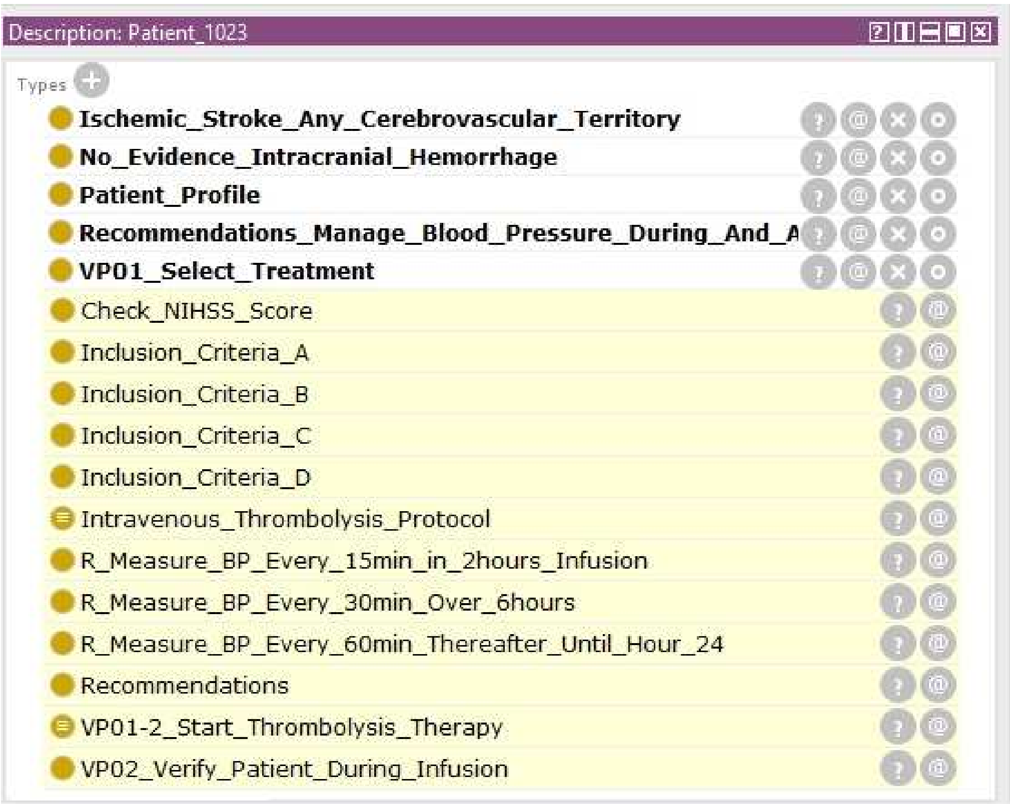
* Patient_Profile(?p) ^ hasPatientSymptomsVerified(?p, ?s) ^ VP02_Verify_Patient_During_Infusion(?s) ^ hasDiagnostic(?s, ?t) ^ Deterioration_Neurological_Status(?t) -> VP02-2_Stop_Infusion(?t)

* Patient_Profile(?p) ^ hasPatientSymptomsVerified(?p, ?s) ^ VP02_Verify_Patient_During_Infusion(?s) ^ hasDiagnostic(?s, ?t) ^ Sudden_Increase_Blood_Pressure(?t) -> VP02-2_Stop_Infusion(?t)

* Patient_Profile(?p) ^ hasPatientSymptomsVerified(?p, ?s) ^ VP02_Verify_Patient_During_Infusion(?s) ^ hasDiagnostic(?s, ?t) ^ Nausea(?t) -> VP02-2_Stop_Infusion(?t)

For example, the patient with ID 1023 presents all inclusion criteria and no exclusion criteria. By reasoning on the ontology, the next activities are displayed as shown in Figure 57.

Figure 57 - Results obtained after the reasoning step



According to Figure 57, the next activity is 'Verify patient during infusion' and 'Check NIHSS score' is the recommendation provided. The user can also receive recommendations about the process context. For example, to verify the rules about the blood pressure management is necessary to select 'Recommendations about the management of blood pressure during and after the infusion'. Then, by answering questions related to the patient's symptoms during infusion the next activities and recommendations are displayed. These questions are presented in Figure 58.

Figure 58 - Questions related to the patient's symptoms

Questionnaire

32) Has the NIH presented by the patient had an increase of 4 points or more during infusion?
(If yes, select hasAdded_4_Points_of_More_in_NIHSS_Score = true)

33) The patient present any of the symptoms below: (If yes, select the corresponding Type)

(a) Headache
(b) Nausea
(c) Decrease level of Consciousness
(d) Deterioration neurological status
(e) Sudden increase of blood pressure
(f) Vomiting;

If, VP02-1_Continue_Infusion is selected answer the question 34:

Select Types: VP03_Verify_Time_Infusion

34) How long the infusion started? (Data property: hasInfusionTime – in minutes)

To access the Recommendations for the first 24 hours after infusion select the concept.

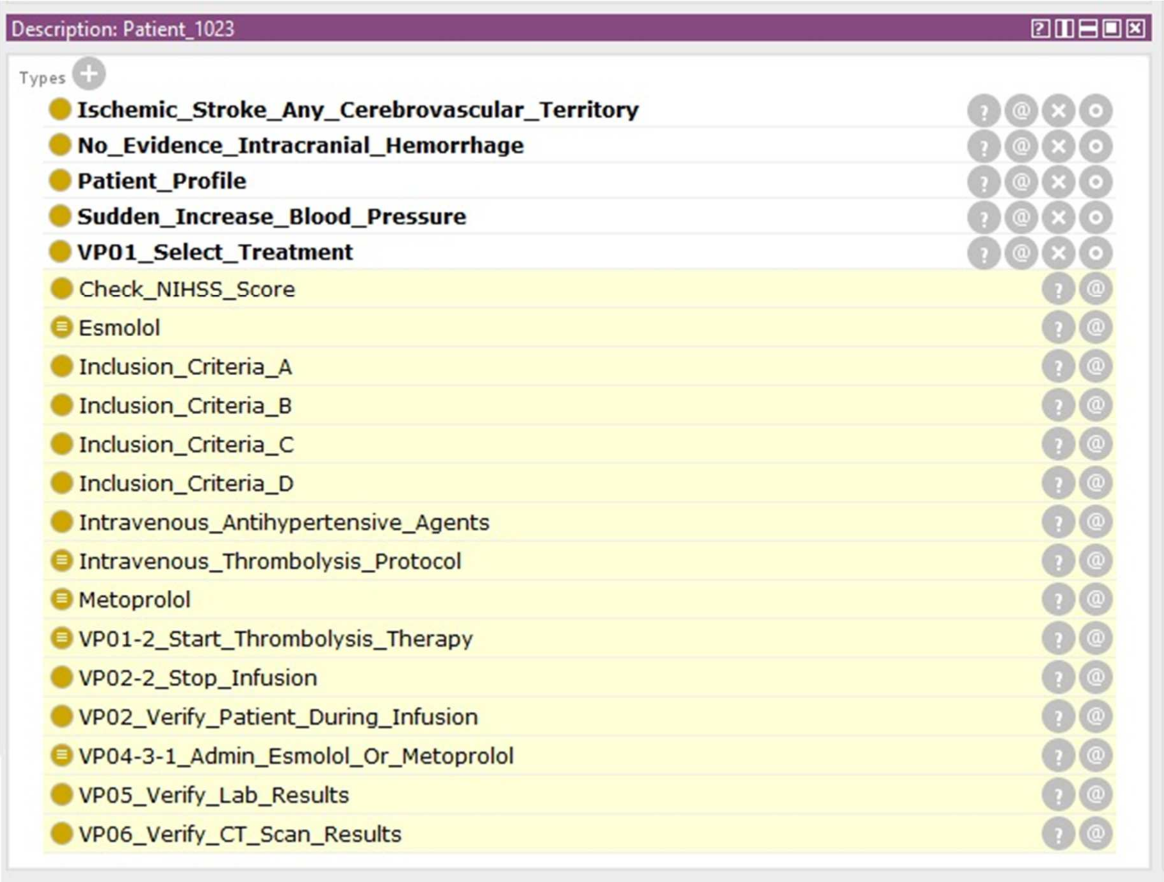
35) What is the blood pressure presented by the patient?
(Select Data properties: DBP_DuringInfusion and SBP_DuringInfusion)

36) In case of hypertension, verify if patient present one of the symptoms below:
(If yes, select the corresponding Type, otherwise select hasContraIndication = false)

Asthma
(a) Heart Failures
(b) Severe abnormalities in heart function that would contraindicate administration of beta-blockers
(c) Uncontrolled hypertension

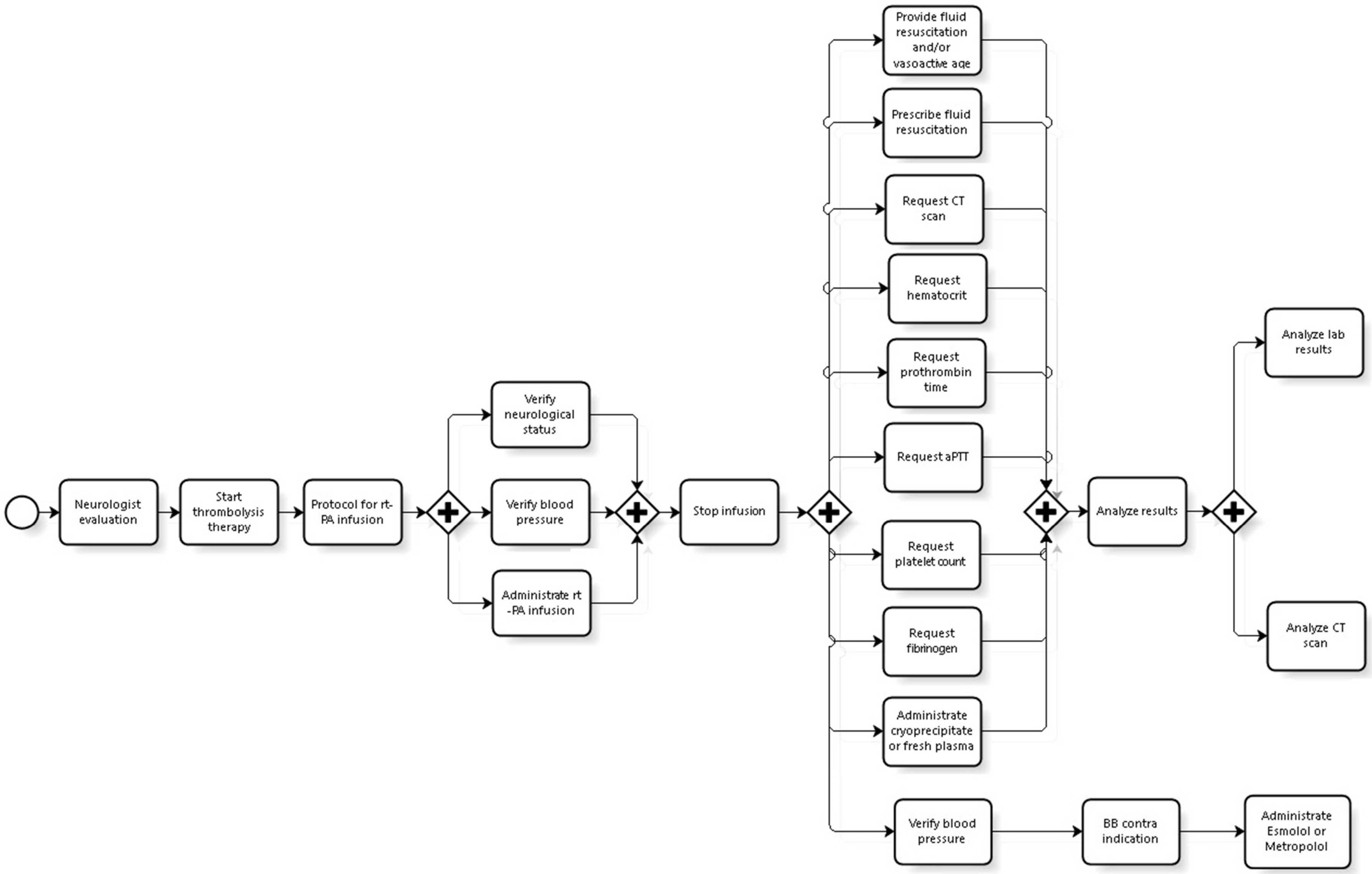
By answering the questions related to the symptoms presented by the patient during infusion (question 34), the level of systolic and diastolic blood pressure during infusion (questions 35 and 36) and by using the reasoning, the next variation points and recommendations are provided according to Figure 59.

Figure 59 - Results of the reasoning step for customizing a process model



According to Figure 59, if the patient presents a sudden increasing of blood pressure, the infusion is interrupted. Besides, if the patient presents a systolic blood pressure (SBP) and/or a diastolic blood pressure (DBP) above the levels stablished in the clinical guideline (DBP≥105 mmHg or SBP≥180 mmHg), the patient should be treat with antihypertensive agents. If the administration of beta-blockers is not contraindicated for the patient, the treatment should be performed with Metoprolol or Esmolol. Next variation points are also displayed. By answering the questionnaire and reasoning on the ontology, the process variant can be derived. The excerpt in Figure 60, shows a fragment of the process built according to the questions answered.

Figure 60 - Fragment of the process variant customized through the merged ontology



5.3. DISCUSSION OF THE RESULTS

This chapter discussed the application of the framework proposed for customizing a process model related with the treatment of patients diagnosed with acute ischemic stroke. There are four treatments that can be provided for treating patients with ischemic stroke: intravenous thrombolysis, intra-arterial protocol, a combination of intravenous and intra-arterial thrombolysis, and mechanical treatment. For the selection of the appropriate treatment, several criteria need to be considered such as the onset of symptoms, patient's age, the symptoms presented by the patient, and the patient's medical record.

An event log related with patients diagnosed with acute ischemic stroke was obtained from a Brazilian hospital. The hospital only provides treatment related with the thrombolysis therapy. If the patient cannot be treat through this treatment, the patient is monitored or transported to another hospital. Thus, the case study is based on the thrombolysis treatment.

The event log contains several information about the patients such as age, symptoms, exams results, patient's medical record, among others. However, the event log does not contain any other information about the activities performed after the selection of the treatment. Thus, as proposed in the framework, a process model was developed based on the event log, the clinical guideline for ischemic stroke, and expert knowledge. The developed process model contains all the scenarios and respect the rules from the clinical guideline. The expert knowledge adapts the process according to the proceedings performed in the Brazilian hospital. The development of the process model helps to understand the commonalities and differences between the treatment provided in the hospital and the clinical guideline.

Based on the process model, a coloured petri net was developed. Thus, an event log was obtained and analysed through process mining techniques. These techniques can simulate several scenarios, and show relations between activities and variation points that are not explicitly in the process model. The heuristic miner was applied in the event log, which enable to identify the variation points and the alternatives for them. To identify the rules for selecting each alternative was applied the decision mining analysis.

Based on the information about the variation points, the alternatives for them and the rules to select each path, the questionnaire was developed. Thus, each question or a set of questions refers to a variation point. And the fact for the questions are related with the alternatives for the variation points. The information related to the variation points are also necessary to build the ontology related with the variation points. In this ontology, the variation points and the alternatives are concepts and subsumed concepts and the SWRL rules define the requirements to select each alternative. The dependency between the variation points are also defined as SWRL rules. Thus, when a variation point is selected, the related variation points are also selected.

Another ontology was developed based in the clinical guideline for ischemic stroke. SWRL rules define the relations between the recommendations and rules of the clinical guideline. Then, both ontologies are merged into a new ontology. Thus, by answering the questionnaire and by reasoning on the ontology, the process model can be individualized. Besides, as a variation point is selected, the related variation points can be visualized and recommendations about the context of the process model are also provided for the user. In this way, the developed approach provides recommendations during the process model customization. These recommendations provide guidance during the customization of the process model. In addition, the recommendations can improve the treatment provide to the patient.

6. DISCUSSION AND CONCLUSIONS

This chapter discusses the conclusions of this research and a perspective for future work. This research proposes an approach for customizing a process model according to the user's requirements. By applying process mining techniques, the framework ensures that the individualized process variants follow the rules of the application environment. Ontologies are used to support the user during the process model customization by providing recommendations about the configuration process and the business context.

Based on the research goal, four questions have been raised. In the following, we present answers for these questions by summarizing the main findings.

RQ1: How to customize a process model in order to obtain a process variant that correctly represent a business context?

This question was decomposed in two sub-questions aiming to demonstrate the aspects that need to be considered when building a customizable process model and how the existing approaches propose to customize a process model. Through the literature review (Chapter 2), two types of customization were identified: by extension and by restriction. When the customizable process model represents all the behaviour into a single model, the customization is performed by restricting its behaviour. On the other side, when the customizable process model represents the most common behaviour, the customization is performed by extending its behaviour.

Therefore, customization is obtained through transformations in the customizable process model. These transformations are performed in the points of the customizable process model that are subject to variation, called variation points. In each variation point a choice need to be made in relation to the available alternatives. This choice is made based on the rules attached with each alternative.

Thus, in each variation point a decision need to be made in order to individualize a process model. A decision can be made at design time (can be made before process execution) or at run-time (can only be made when the information is available). Besides, a decision in one point can have a direct implication on the other variation points.

In this way, when building a customizable process model is essential to identify the aspects related with the variation points. Another aspect is related to the behavioural and structural correctness of the individualized process model. The structural correctness ensures that all nodes in the customized process model are connected and the behavioural correctness aims to avoid deadlocks or livelocks. In addition, guidance and recommendations during process model customization ensure that the customized process model is correctly customized. Recommendations may be related with the process model customization or with the application context.

The analysis of the related work (Chapter 3) provides an understanding about the existing approaches for process model customization enabling to identify the drawbacks in this area. The approaches for customizing a process model can be classified as configurable nodes, element annotation, activity specialization, and fragment customization.

Configurable nodes are points in the process subject to variation, in which options are assigned. A configurable node can be defined as activities, events, gateways, resources and objects associated with activities. In the element annotation, a model element (activities, events, gateways, sequence flows, resources and objects) is a variation point, which graphically annotate a model with properties of the application domain. In the activity specialization, the variation points are activities defined as abstract and optional. Fragment customization is related to the customization of the process model through change operations, which enable to delete, insert, move or modify a process fragment.

This research focused on the configurable node approaches. Thus, four approaches (C-EPC, E-iEPC, ADOM, C-YAWL.) have been analysed regarding the type of customization (by restriction and/or by extension), structural and behavioural correctness of the process variants, guidance and recommendations during customization, and the relationships between the variation points. The analysis of these approaches showed that support decision making during customization is not addressed by the existing approaches.

C-EPC ensures behavioural and structural correctness. Guidelines can also be provided, but they are related to each variation point and not with the business contexts. Besides, is not easy to identify the relationships between the variation points. C-iEPC extend the C-EPC by representing variability related to the roles and objects. Configurable YAWL (C-YAWL) customize the process model by hiding or blocking the configurable activities. The rules to select the activities are not expressed in the process model, thus is not clear the relationship between the nodes.

Structural and behavioural correctness are ensured in C-iEPC and C-YAWL by means of algorithm. Both approaches enable to apply the questionnaire-model approach in order to provide guidance to the user during customization. The questionnaire-model approach is composed of questions and facts. Each question corresponds to a variation point and each fact corresponds to the alternatives available for the variation point in question. Thus, by selecting a fact, a process variant is selected. However, these approaches do not provide recommendations for the user and the relationships between the variation points are not obvious.

ADOM customize the process model by restricting and by extended its behaviour. ADOM cannot ensure the behavioural and structural correctness. Guidance and recommendations are not provided. The relationships among the variation points cannot be identified.

The chapter also presented the most relevant approaches for customize a process model regarding the other types of classification: element annotation, activity specialization and fragment customization. Thus, regarding these classifications, the Configurative Process Modelling, PESOA approach, and Provop approach are discussed. The PESOA approach and the Provop approach have been used as base for the development of many other approaches.

Thus, the analysis of these approaches for process model customization showed that an approach capable of providing recommendations related to the business process customization and the business context, also showing the impact of a decision on the process model are still missing. The discovery of these needs help to answer the next questions.

RQ2: What are the theoretical and practical arguments motivating the application of process mining to discover customizable process models?

This question was decomposed in two questions aiming to identify the process mining techniques that can be applied to identify the aspects related with the process variants enabling to improve the customizable process model, and how improve the customizable process model to consider several scenarios.

Process mining techniques analyse the data about the business process execution, enabling to discover, monitor, and improve business process. Several algorithms have been developed focusing in different aspects. Thus, when analysing the process mining techniques, we focused on the algorithms that could be applied for identify the elements for developing a customizable process model as identified in the RQ1, i.e., the variation points, the alternatives for them and the rules for selecting each alternative.

As result, heuristic miner and decision miner were applied. When constructing a process model, heuristic miner considers the frequencies of events and sequences. This technique enables to discover the variation points and the alternatives. However, it cannot be used to discover the rules for selecting the available alternatives. Thus, decision miner was applied to discover these rules. Decision miner discover the points where the process is split into alternatives paths (i.e., the decision points) and the rules for select each alternative.

The discovery of process variants from the log enables improving process variants by correcting deviations, anticipating problems, etc. In addition, implicit knowledge can be captured, thus enabling enriching the process variants. Besides, by analysing the variation points two types of variation points were identified: mandatory and optional. The variation points defined as mandatory are related to the selection of process variants, thus they inherit the optional variation points. This definition of the variation points enables to understand the interdependencies between the variation points. In this way, when a selection is made in the mandatory variation points, the related set of optional variation points is enabled.

The analysis provided by the heuristic miner and the decision miner may show deviations that might exist in the business process. Besides, an event log may not contain all the activities performed during the process execution. Thus, the process model obtained from the event log analysis can be improved with expert knowledge and the internal and external regulations.

RQ3: What are the theoretical and practical arguments motivating the use of ontologies for process model customization?

This question was decomposed in one question aiming to identify if ontologies can be applied for providing decision-making support during the process model customization. Ontology is like a vocabulary structuring the concepts from a domain, its attributes, and the relationships between them, enabling to represent, (re)use, share, and exchange the common understanding about these elements. Ontologies also enable to use a reasoner engine, which take a collection of axioms written in OWL enabling to deduce new knowledge. A reasoner is also used for classification, query, to check the ontology consistency, among others.

A process model is customized by selecting a choice for each variation point that might exist in the customizable process model. As mentioned, the selection of an available choice relies on the rules attached to them. These rules are defined based on the business rules, including the internal and external regulations. As result, the process model customization may rely in a large amount of information. Besides, the selection in one variation point may influence the selection of other variation points. These relations can also be defined in the ontology. Thus, ontologies can be applied to formalize the knowledge for customizing process models in a specific business context.

RQ4: How process mining and ontologies can be applied to customize a process model according to all the requirements related to a particular business context?

By answering the previous questions, it was possible to identify how ontologies and process mining can be applied for customize process models. The use of both techniques allows to fulfil the existing gaps related to the process model customization. As mentioned, there is a need for supporting the decision-making during customization through recommendations about the business process customization and the business context. In addition, the understanding about the relationships between the variation points enable to evaluate the impact of the decision in the process model customization.

Based on this, a framework for customizing process variants was developed. The framework is composed of three steps. In the first step, an event log is obtained and analysed through the decision mining techniques. If the event log is incomplete, the event log can be improved with the expert knowledge and the internal and external regulations. In this case, a process model is developed. In order to analyse the developed process model, an event log can be generated and analysed by applying process mining techniques. To identify the process variants, the decision miner is applied to discover the variation points, the alternatives for them, and the related rules. The decision miner also enables to understand the dependencies between the variation points.

A process model is customized according to the user's requirements. In this way, the questionnaire-model approach can be applied to guide users in providing the information needed for process variant selection. In this approach, each variation point refers to a question, and the alternatives available for the variation points refers to facts for answer the questions. Thus, by selecting a fact related to a question, the related alternative is selected in the process model. In the proposed framework, the questionnaire-model is developed based on the information provided by the decision point analysis.

The third step refers to apply ontologies for customizing a process model. It was proposed to use two ontologies, one based on the variation points and other related to the internal and external regulations. The relation between the variation points is specified based on the analysis provided by the decision miner. Then, the ontologies are merged. The resulting ontology contains all the knowledge need for customize a process variant. In this way, by answering the questionnaire, and then, by using a reasoner engine, the ontology displays the recommendations about the next variation points and the recommendations about the business context in relation with the alternative selected.

Thus, the framework proposes to customize a process model by restricting its behaviour. By customizing the process model through ontologies ensures supporting decision making during customization and the behavioural correctness. As drawback, the present work does not automatically select the process variants. In this way, the questionnaire is also not connected with the variation points. Besides, other perspectives, such as the organizational structure are not considered in the framework. In addition, the framework should contain all types of treatments that can be provided for patients with ischemic stroke.

Considering these drawbacks, as future work, the process model can be annotated with the concepts in the ontology enabling the automation of the modelling and configuration tasks. Thus, an interface can be developed for the user to customize the process model. The system should link the questionnaire, the process model and the ontologies. In this way, the interface displays the questions for the user and according its selection, the process fragment till the process variant is complete. Besides, only the relevant questions should be displayed for the user. In addition, according the user' selection, recommendations about the process should be provided.

APPENDIX A – QUESTIONNAIRE FOR PROCESS VARIANT SELECTION

1. What is the patient's age? (Data property: hasAge)

2. What is the patient's gender? (Data property: hasGender)

3. How long has the patient displayed signs of ischemic stroke? (Data property: hasSympOnset – e.g., 2.3 equal to 2h30m)

4. Does the patient present ischemic stroke in any cerebrovascular territory? (If yes, select Types: Ischemic_Stroke_Any_Cerebrovascular_Territory)

5. Does the patient present evidence of intracranial haemorrhage in brain imaging techniques? (If no, select Types: No_Evidence_Intracranial_Hemorrhage)

6. Does the patient use oral anticoagulant? (If yes, select Types: Oral_Anticoagulant)

7. What is the Prothrombin time presented by the patient? (Data property: hasINR and hasPT) e.g., INR = 1.7, PT = 15 seconds.

8. Has the patient used heparin? If yes, how many hours ago did the patient take this medication? (Data property: hasHeparinUseHours = hours)

9. Does the patient present prolonged aPTT? (If yes, select Types: Prolonged_aPPT)

10. Has the patient had ischemic stroke in the last three months? (If yes, select Data property: hasIschemicStroke = true)

11. Has the patient had severe head trauma in the last three months? (If yes, select Data property: hasHeadTrauma)

12. Has the patient history of intracranial hemorrhage and/or cerebrovascular malformation? (If yes select Types: History_Intracranial_Hemorrhage and/or Cerebrovascular_malformation)

13. Does the patient present hypodensity of more than one-third of the middle cerebral artery territory in the head CT? (If yes select Type: Hypodensity_More_One_Third_Middle_Cerebral_Artery_Territory)

14. What is the systolic and the diastolic blood pressure presented by the patient? (Select Data property: SBP_1, SBP_2, SBP_3, DPB_1, DPB_2, DPB_3).

If necessary, (hypertension (SBP >= 185 mmHg, DBP >= 110mmHg) answer the next question (15):

15. Does the patient have asthma, heart failure, severe abnormalities in heart function that would contra-indicate administration of beta-blockers or uncontrolled hypertension? (If yes,

select Types: Asthma, Heart_Failure, Severe_Abnormalities_Heart_Function, Uncontrolled_Hypertension)

16. Has the signs and symptoms had a rapid and complete resolution before thrombolytic agent administration? (If yes, select Types: Rapid_and_Complete_Resolution_Signs_Symptoms_Before_Thrombolytic)

17. Does the patient present mild neurological deficit (with no significant functional deterioration)? (If yes, select Types: Mild_Neurological_Deficit_No_Significant_Functional_Deterioration)

18. Has the patient had major surgery in the last two weeks? (If yes, select hasMajorSurgery = true)

19. Has the patient had an invasive procedure in the last two weeks? (If yes, select hasInvasiveProcedure = true)

20. Has the patient history of genitourinary bleeding in the last 3 weeks? (If yes, select hasGenitourinaryBleeding = true)

21. Has the patient history of gastrointestinal bleeding in the last 3 weeks? (If yes, select hasGastrointestinalBleeding = true)

22. Has the patient history of esophageal varices (If yes, select Esophageal_Varices)?

23. Has the patient history of an arterial puncture at a no compressible site within the last 7 days? (hasArterialPuncture = true)

24. Does the patient present coagulopathy (INR > 1.7) (Select Types: Coagulopathy and Data property: hasINR)? Prolonged aPTT (Select Types: Prolonged_aPTT)?

25. What is the platelet count presented by the patient? (Data property: hasPlateletCount, <100.000/mm3)

26. What is the blood sugar level (<50mg/dL) presented by the patient? (Select Data property: hasBlood_Sugar)

27. Does the patient present signs of endocarditis (Select Types: Endocartitis)? Septic embolus (Select Types: Septic_Embolus)? Or pregnancy (If yes, select Data property: isPregnant = true)?

28. Has the patient history of myocardial infarction in the last 3 months? (If yes, select Data property: hasMyocardial_Infarction = true)

29. There is clinical suspicion of subarachnoid haemorrhage (Types: subarachnoid haemorrhage) or acute aortic dissection (Types: Acute_Aortic_Dissection)?

30. Has the patient had epileptic seizure at the onset neurological deficit (Select Types: Epiletic_Seizure_at_the_Onset_Neurological_Deficit) or cerebral aneurism (Select Types: Cerebral_Aneurism)

31. What is the NIH presented by the patient? (Select Data properties: hasNIHSS)

For verifying the factors that are not absolute exclusion criteria select Factors_That_Are_Not_Absolute_Exclusion_Criteria.

Select hasNoExclusionCriteria = true, if:

- If the answer for questions 6 to 29 is No;
- If the treating neurologist is convinced that the factors that are not an absolute exclusion criteria are unrelated to the patient's acute neurological deficit;
- If the factors that alter the risk/the benefit ratio of thrombolytic therapy does not constitute a contraindication to its use;

Otherwise, select hasInclusionCriteria = true.

After answer the previously questions, Select VP01_Select_Treatment. If the treatment selected is VP01-2_Start_Thrombolysis_Therapy:

To access recommendations about neurological assessment during infusion, select: Types: Neurological_Assessment_Recommendations

To access recommendations about the management of blood pressure during infusion, select Types: Recommendations_Manage_Blood_Pressure_During_And_After_Infusion

32. Check the NIH. Has the NIH presented by the patient had an increase of 4 points or more during infusion? (If yes, select hasAdded_4_Points_of_More_in_NIHSS_Score = true)

33. The patient present any of the symptoms below: (If yes select the corresponding Type)

- Vomiting;
- Severe Headache
- Nausea
- Decrease level of Consciousness
- Deterioration neurological status
- Sudden increase of blood pressure

If, VP02-1_Continue_Infusion is selected answer the question 34:

Select Types: VP03_Verify_Time_Infusion

34. How long the infusion started? (Data property: hasInfusionTime – in minutes)

To access the Recommendations for the first 24 hours after infusion select the concept.

If, VP02-2_Stop_Infusion is selected answer the following questions:

35. What is the blood pressure presented by the patient? (Select Data properties: DBP_DuringInfusion and SBP_DuringInfusion)

In case of hypertension, verify if patient present one of the symptoms below: (If yes, select the corresponding Type)

- Asthma
- Heart Failures
- Severe abnormalities in heart function that would contraindicate administration of beta-blockers
- Uncontrolled hypertension

VP05_Verify_Lab_Results or VP06_Verify_CT_Scan_Results. It is not necessary follow a specific order:

If the laboratories results are available, answer the following questions:

- Does the patient present a low fibrinogen? (Data property: hasLowFibrinogen = true)
- Does the patient present abnormal PT or aPTT? (Data property: hasAbnormalPT and/or has Abnormal_aPPT)
- Does the patient present low haematocrit or high? (Data property: hasLowHematocrit = true or hasHighHematocrit = true)
- Does the patient present a low platelet count? (Data property: hasLowPlateletCount)

If the CT scan results are available, answer the following questions:

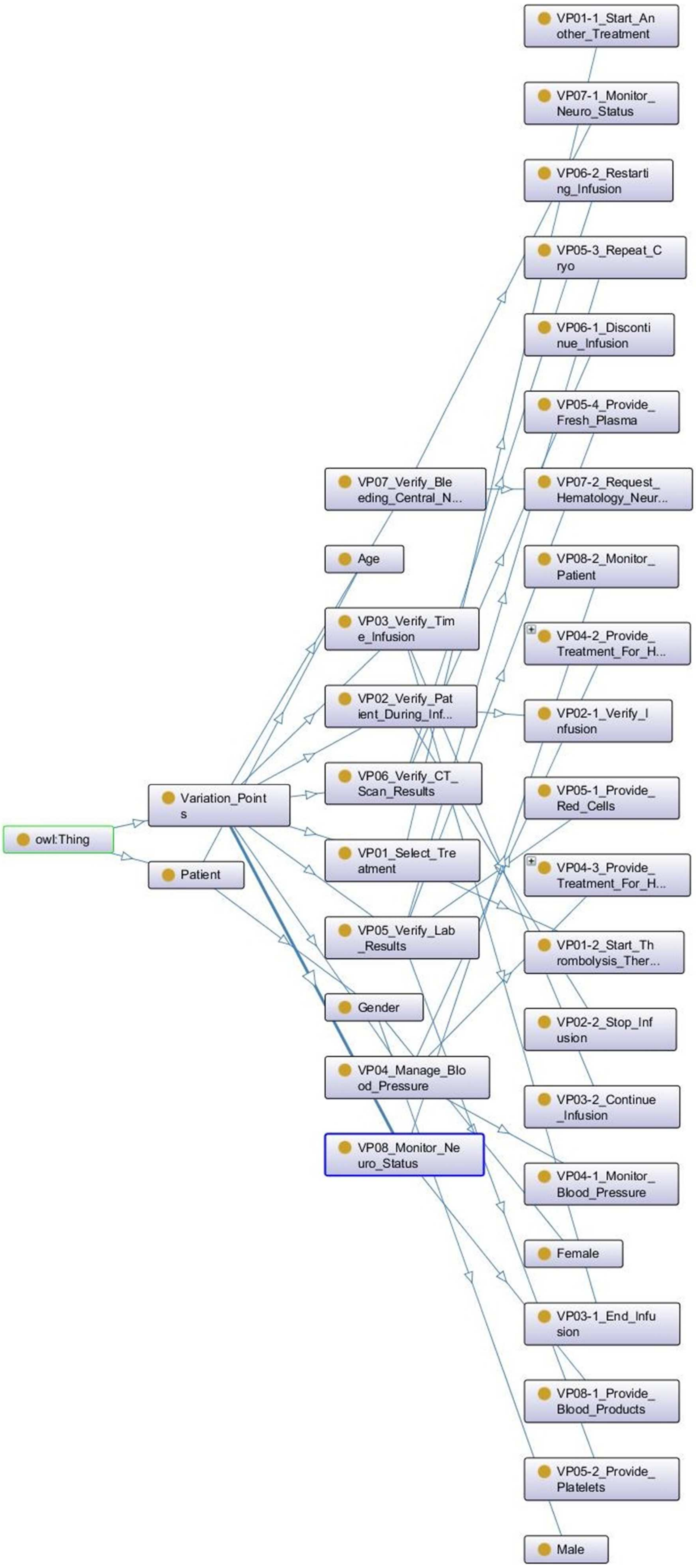
- Does the CT scan show signs of intracranial haemorrhage? (If yes, select hasIntracranialHemorrhage = true otherwise hasIntracranialHemorrhage = false)

If VP07_Verify_Bleeding_Central_Nervous. Answer the following question:

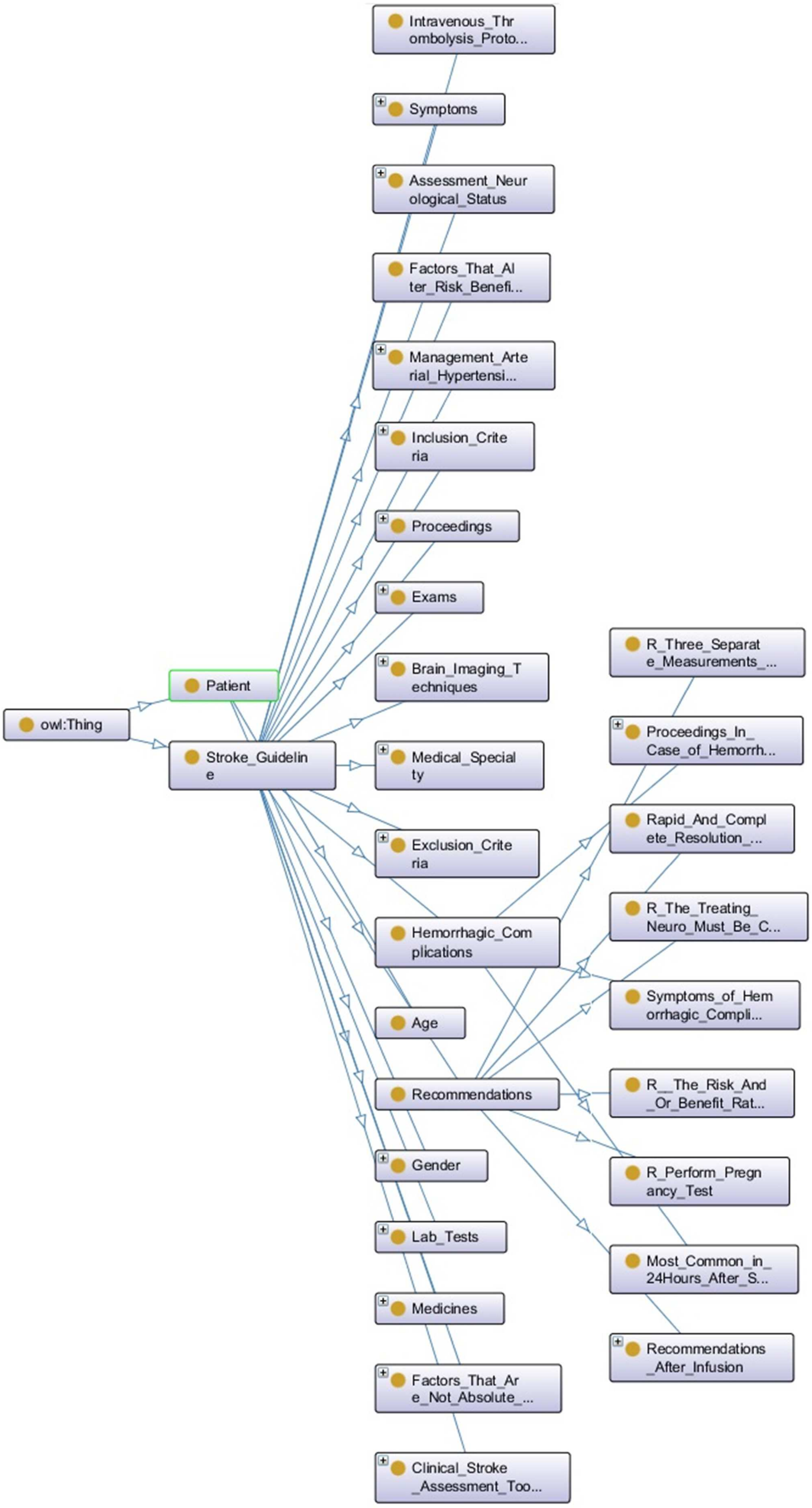
- Does the patient present signs of bleeding in the central nervous system? (If yes, select hasCentralNervousBleeding = true, otherwise select hasCentralNervousBleeding = false

36. Does the patient’s clinical status continue to deteriorate after four to six hours? (If yes, select hasDeterioationNeuroStatus = true, otherwise select hasDeterioationNeuroStatus = false).

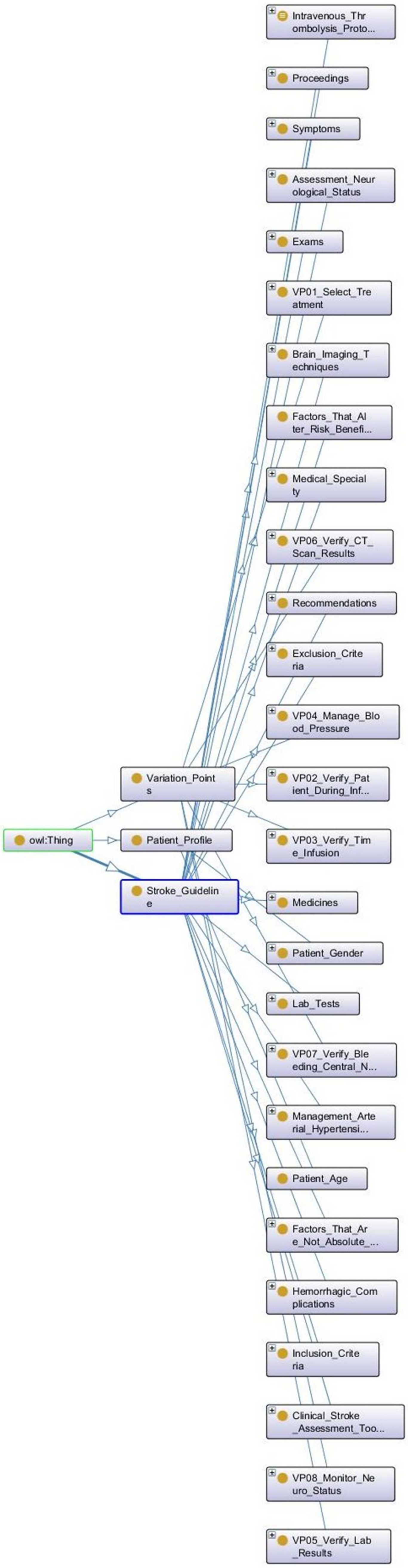
APPENDIX B.1 – ONTOLOGY RELATED WITH THE VARIATION POINTS



APPENDIX B.2 – ONTOLOGY RELATED WITH THE CLINICAL GUIDELINE FOR THE TREATMENT OF ACUTE ISCHEMIC STROKE



APPENDIX B.3 – MERGED ONTOLOGY



7. REFERENCE

ABBURU, S. A survey on ontology reasoners and comparison. **International Journal of Computer Applications**, 57(17), 2012.

AGRAWAL, G.L. and GUPTA, H. Optimization of C4. A decision tree algorithm for data mining application. **International Journal of Emerging Technology and Advanced Engineering**, *3*(3), pp.341-345, 2013.

AIZED, T. Modelling and performance maximization of an integrated automated guided vehicle system using coloured Petri net and response surface methods. **Computers & Industrial Engineering**, 57(3), 822-831, 2009.

AMROUCH, S. and MOSTEFAI, S. Survey on the literature of ontology mapping, alignment and merging. **Information Technology and e-Services (ICITeS), 2012 International Conference on** (pp. 1-5). IEEE, 2012.

ARSENE, O., DUMITRACHE, I. and MIHU, I. Expert system for medicine diagnosis using software agents. **Expert Systems with Applications**, 42(4), pp.1825-1834, 2015.

ASADI, M., MOHABBATI, B., GRÖNER, G. & GASEVIC, D. Development and validation of customized process models. **Journal of Systems and Software**, 96, pp.73-92, 2014.

ASSY, N., CHAN, N.N., GAALOUL, W. An automated approach for assisting the design of configurable process models. **IEEE Transactions on Services Computing**, *8*(6), pp.874-888, 2015.

AYORA, C., TORRES, V., REICHERT, M., WEBER, B. & PELECHANO, V. Towards run-time flexibility for process families: open issues and research challenges. **International Conference on Business Process Management** (pp. 477-488), Springer Berlin Heidelberg, 2012.

AYORA, C., TORRES, V., WEBER, B., REICHERT, M. and PELECHANO, V. Enhancing modeling and change support for process families through change patterns. **Enterprise,**

Business-Process and Information Systems Modeling (pp. 246-260). Springer, Berlin, Heidelberg, 2013.

AYORA, C., TORRES, V., WEBER, B., REICHERT, M. and PELECHANO, V. VIVACE: A framework for the systematic evaluation of variability support in process-aware information systems. **Information and Software Technology**, *57*, pp.248-276, 2015.

AYORA, C., TORRES, V., DE LA VARA, J.L., PELECHANO, V. Variability management in process families through change patterns. **Information and Software Technology**, 74, pp.86-104, 2016.

BARUWA, O.T., PIERA, M.A. and GUASCH, A. TIMSPAT–Reachability graph search-based optimization tool for coloured Petri net-based scheduling. **Computers & Industrial Engineering**, 101, 372-390, 2016.

BECKER, J., DELFMANN, P., DREILING, A., KNACKSTEDT, R. and KUROPKA, D. Configurative Process Modelling – outlining an approach to increased business process model usability. Proceedings of the 15th IRMA International Conference (pp. 1-12). New Orleans: Gabler, 2004.

BECKER, J., DELFMANN, P. and KNACKSTEDT, R. Adaptive reference modeling: integrating configurative and generic adaptation techniques for information models. Reference Modeling, pp.27-58, 2007.

BEIMEL, D. and PELEG, M. Using OWL and SWRL to represent and reason with situation-based access control policies. **Data & Knowledge Engineering**, 70(6), pp.596-615, 2011.

BORST, W.N. Construction of engineering ontologies for knowledge sharing and reuse. Universiteit Twente, 1997.

BOSE, R.J.C. and VAN DER AALST, W.M.P. Process diagnostics using trace alignment: opportunities, issues, and challenges. **Information Systems**, 37(2), pp.117-141, 2012.

BRANK, J., GROBELNIK, M. and MLADENIĆ, D. A survey of ontology evaluation techniques, 2005. Available in:
http://ailab.ijs.si/dunja/sikdd2005/papers/BrankEvaluationSiKDD2005.pdf

BRATOSIN, C., SIDOROVA, N. and VAN DER AALST, W.M.P. Distributed genetic process mining. Evolutionary Computation (CEC), 2010 IEEE Congress on (pp. 1-8). IEEE, 2010, July.

BUIJS, J.C., VAN DONGEN, B.F. and VAN DER AALST, W.M.P. Mining configurable process models from collections of event logs. **Business Process Management** (pp. 33-48). Springer, Berlin, Heidelberg, 2013.

BUIJS, J.C. and REIJERS, H.A. Comparing business process variants using models and event logs. **Enterprise, Business-Process and Information Systems Modeling** (pp. 154-168). Springer, Berlin, Heidelberg, 2014.

BÜHNE, S., HALMANS, G. and POHL, K. Modelling dependencies between variation points in use case diagrams. **Proceedings of 9th Intl. Workshop on Requirements Engineering-Foundations for Software Quality** (pp. 59-70), 2003.

BURATTIN, A., CONTI, M. and TURATO, D. Toward an anonymous process mining. Future Internet of Things and Cloud (FiCloud), 2015 3rd International Conference on (pp. 58-63). IEEE, 2015.

CAMBRIGDE DICTIONARY. Available in: http://dictionary.cambridge.org/dictionary/english-portuguese/flexibility

CHENG, H.J. and KUMAR, A. Process mining on noisy logs – Can log sanitization help to improve performance? **Decision Support Systems**, 79, pp.138-149, 2015.

CHRISTLEY, S., XIANG, X. and MADEY, G. An ontology for agent-based modeling and simulation. **Proceedings of the agent 2004 conference**, 2004.

CORCHO, O. and GÓMEZ-PÉREZ, A. A roadmap to ontology specification languages. **EKAW** (Vol. 2000, pp. 80-96), 2000.

CORCHO, O., FERNÁNDEZ-LÓPEZ, M. and GÓMEZ-PÉREZ, A. Methodologies, tools and languages for building ontologies. Where is their meeting point? **Data & knowledge engineering**, 46(1), pp.41-64, 2003.

CORCHO, O., FERNÁNDEZ-LÓPEZ, M. and GÓMEZ-PÉREZ, A. Ontological engineering: what are ontologies and how can we build them? Chapter 3, pp. 44-70, 2007.

CPN Tools. http://cpntools.org/. Accessed: 25.03.17.

CRESWELL, J.W. and POTH, C.N. Qualitative inquiry and research design: Choosing among five approaches. Sage publications, 2017.

DJELLALI, C. A new data mining system for ontology learning using dynamic time warping alignment as a case. **Procedia Computer Science**, 21, pp.75-82, 2013.

DE MEDEIROS, A.A. and GÜNTHER, C.W. Process mining: Using CPN tools to create test logs for mining algorithms. **Proceedings of the sixth workshop on the practical use of coloured Petri nets and CPN tools** (CPN 2005*)* (vol. 576), 2005.

DE MEDEIROS, A.K.A, PEDRINACI, C., VAN DER AALST, W.M.P., DOMINGUE, J., SONG, M., ROZINAT, A., NORTON, B., CABRAL, L. An outlook on semantic business process mining and monitoring. On the Move to Meaningful Internet Systems 2007: OTM 2007 Workshops, pp. 1244-1255. Springer Berlin Heidelberg, 2007.

DERGUECH, W., VULCU, G. and BHIRI, S. An Indexing Structure for Maintaining Configurable Process Models. *BMMDS/EMMSAD*, *50*, pp.157-168, 2010.

DESEL, J. and ESPARZA, J. Free choice Petri nets (Vol. 40). Cambridge university press., 2005.

DETRO, S.P., MOROZOV, D., LEZOCHE, M., PANETTO, H., SANTOS, E.P. and ZDRAVKOVIC, M. Enhancing semantic interoperability in healthcare using semantic process mining. **6th International Conference on Information Society and Technology**, ICIST 2016 (Vol. 1, pp. 80-85), 2016.

DETRO, S.P., SANTOS, E.A.P., PANETTO, H., LOURES, E.D.F.R. and LEZOCHE, M. Managing business process variability through process mining and semantic reasoning: An application in healthcare. **Working Conference on Virtual Enterprises** (pp. 333-340). Springer, Cham., 2017.

DRESCH, A., LACERDA, D.P. and MIGUEL, P.A.C. A distinctive analysis of case study, action research and design science research. Revista Brasileira de Gestão de Negócios, 17(56), p.1116, 2015.

EINSTEIN, A. What Life Means to Einstein – An interview by George Sylvester Viereck. The Saturday Evening Post, 1929. Available in: http://www.saturdayeveningpost.com/wp-content/uploads/satevepost/what_life_means_to_einstein.pdf

EL FAQUIH, L., SBAI, H. AND FREDJ, M. Semantic variability modeling in business processes: A comparative study. **Internet Technology and Secured Transactions (ICITST), 2014 9th International Conference for** (pp. 131-136). IEEE, 2014.

EL FAQUIH, L., SBAI, H., FREDJ, M. Configurable process models: A semantic validation. **2015 10th International Conference on Intelligent Systems: Theories and Applications** (SITA) (pp. 1-6), IEEE, 2015.

FENSEL, D. Ontologies: Silver Bullet for Knowledge Management and Electronic Commerce. Springer-Verlag, Berlin, 2000.

FERNÁNDEZ-LLATAS, C., BENEDI, J.M., GARCÍA-GÓMEZ, J.M. and TRAVER, V. Process mining for individualized behavior modeling using wireless tracking in nursing homes. **Sensors**, 13(11), pp.15434-15451, 2013

FERNÁNDEZ-LÓPEZ, M. and GÓMEZ-PÉREZ, A. and JURISTO, N. Methontology: From ontological art towards ontological engineering. Paper presented at the Spring Symposium on Ontological Engineering of AAAI (pp. 33-40). Stanford University, 1997.

FERREIRA, D.R. & GILLBLAD, D. Discovering Process Models from Unlabelled Event Logs. **BPM** (Vol. 9, pp. 143-158), 2009.

FISH, J. Ranking Scale. On KREUTZER, J. S., DELUCA, J., and CAPLAN, B. Encyclopedia of clinical neuropsychology pp.2110-2112, Springer New York, 2011.

FLUXICON, 2017. Available in: https://fluxicon.com/disco/

FOLINO, F., GRECO, G., GUZZO, A. and PONTIERI, L. Discovering expressive process models from noised log data. Proceedings of the 2009 International Database Engineering & Applications Symposium, pp. 162–172. ACM, 2009.

FU, G. FCA based ontology development for data integration. **Information processing & management**, 52(5), pp.765-782, 2016.

GUARINO, N. and GIARETTA, P. Ontologies and knowledge bases: towards a terminological clarification. In N. MARS, Ed. Towards Very Large Knowledge Bases: Knowledge Building and Knowledge Sharing 1995. pp. 25 – 32. Amsterdam: IOS Press.

GUARINO, N. Formal ontology and information systems. **Proceedings of FOIS** (Vol. 98, No. 1998, pp. 81-97), 1998.

GAŠEVIĆ, D., DJURIC, D. and DEVEDŽIC, V. Model driven engineering and ontology development. Springer Science & Business Media, 2009.

GODOY, A.S. Introdução à pesquisa qualitativa e suas possibilidades. Revista de administração de empresas, *35*(2), pp.57-63, 1995.

GÓMEZ-PÉREZ, A., FERNÁNDEZ-LÓPEZ, M. and CORCHO, O. Ontological engineering. **AI Magazine**, 36(56), p.12, 1991.

GÓMEZ-PÉREZ, A. and CORCHO, O. Ontology languages for the semantic web. **IEEE Intelligent systems**, *17*(1), pp.54-60, 2002.

GOTTSCHALK, F., VAN DER AALST, W.M., JANSEN-VULLERS, M.H. and LA ROSA, M., 2008. Configurable workflow models. **International Journal of Cooperative Information Systems**, 17(02), pp.177-221.

GOTTSCHALK, F.F. Configurable process models, Ph.D. Dissertation, **Technische Universiteit Eindhoven**, 2009.

GOTTSCHALK, F., WAGEMAKERS, T.A., JANSEN-VULLERS, M.H., VAN DER AALST, W.M.P. and LA ROSA, M. Configurable process models: Experiences from a municipality case study. **International Conference on Advanced Information Systems Engineering** (pp. 486-500). Springer, Berlin, Heidelberg, 2009, June.

GRÖNER, G., BOŠKOVIĆ, M., PARREIRAS, F.S. and GAŠEVIĆ, D. Modeling and validation of business process families. **Information Systems**, 38(5), pp.709-726, 2013.

GRUBER, T.R. A translation approach to portable ontology specifications. **Knowledge acquisition**, 5(2), pp.199-220, 1993.

GRUBER, T.R., Toward principles for the design of ontologies used for knowledge sharing? International Journal of Human-Computer Studies, vol. 43, no. 5, pp.907-928, 1995.

GRÜNINGER, M. and FOX, M.S. Methodology for the design and evaluation of ontologies. In D. Skuce (Ed.), IJCAI95 Workshop on Basic Ontological Issues in Knowledge Sharing (pp. 6.1–6.10), 1995.

GUARINO, N., 1995. Formal ontology, conceptual analysis and knowledge representation. International journal of human-computer studies, 43(5-6), pp.625-640.

GUIZZARDI, G., WAGNER, G., ALMEIDA, J.P.A. and GUIZZARDI, R.S. Towards ontological foundations for conceptual modeling: the unified foundational ontology (UFO) story. **Applied ontology**, 10(3-4), pp.259-271, 2015.

GÜNTHER, C. and VAN DER AALST, W.M.P. Fuzzy mining–adaptive process simplification based on multi-perspective metrics. **Business Process Management**, pp.328-343, 2007.

GÜNTHER, C., ROZINAT, A., VAN DER AALST, W.M.P., VAN UDEN, K. Monitoring deployed application usage with process mining. **BPM Center Report BPM**-08-11 pp., 1-8, 2008.

HAAV, H.M. A Semi-automatic Method to Ontology Design by Using FCA. In *CLA*, 2004. Available in: http://ftp.informatik.rwth-aachen.de/Publications/CEUR-WS/Vol-110/paper2.pdf

HADZIC, M., WONGTHONGTHAM, P., DILLON, T. and CHANG, E. Ontology-based multi-agent systems. Germany: Springer Berlin Heidelberg, 2009.

HALLERBACH, A., BAUER, T. and REICHERT, M. Context-based configuration of process variants, 2008. Available in: http://dbis.eprints.uni-ulm.de/199/.

HALLERBACH, A., BAUER, T. and REICHERT, M. Capturing variability in business process models: the Provop approach. **Journal of Software: Evolution and Process**, *22*(6-7), pp.519-546, 2009a.

HALLERBACH, A., BAUER, T. and REICHERT, M. Configuration and management of process variants. **Handbook on Business Process Management 1** (pp. 237-255). Springer Berlin Heidelberg, 2010

HENDLER, J. Agents and the semantic web. **IEEE Intelligent systems**, 16(2), pp.30-37, 2001.

HOMAYOUNFAR, P. Process mining challenges in hospital information systems. **Fe***derated Conference on Computer Science &* **Information Systems** *(FedCSIS)*, p1135-1140, 6p., Publisher: IEEE, 2012.

HOOI, Y.K., HASSAN, M.F. and SHARIFF, A.M. Ontology evaluation—A criteria selection framework. **Mathematical Sciences and Computing Research (iSMSC), International Symposium on** (pp. 298-303). IEEE, 2015.

HORROCKS, I. DAML+OIL: A Description Logic for the Semantic Web. **IEEE Data Eng. Bull**, 25(1), pp.4-9, 2002.

HORROCKS, I., PATEL-SCHNEIDER, P.F., BOLEY, H., TABET, S., GROSOF, B. and DEAN, M. SWRL: A semantic web rule language combining OWL and RuleML. **W3C Member submission**, 21, p.79, 2004.

HUANG, Y., FENG, Z., HE, K., HUANG, Y. Ontology-based configuration for service-based business process model. **Services Computing** (SCC), 2013 IEEE International Conference on (pp. 296-303). IEEE, 2013.

HUANG, Z., LU, X., DUAN, H. and FAN, W. Summarizing clinical pathways from event logs. **Journal of biomedical informatics**, 46(1), pp.111-127, 2013.

JANS, M., VAN DER WERF, J., LYBAERT, N., VANHOOF, K. A business process mining application for internal transaction fraud mitigation. **Expert Systems with applications**, vol. 38 pp. 13351-13359, 2011.

JENSEN, K., KRISTENSEN, L.M. and WELLS, L. Coloured Petri Nets and CPN Tools for modelling and validation of concurrent systems. **International Journal on Software Tools for Technology Transfer**, 9(3-4), 213-254, 2007.

KALFOGLOU, Y. and SCHORLEMMER, M. Ontology mapping: the state of the art. **The knowledge engineering review**, 18(1), pp.1-31, 2003.

KALIBATIENE, D. and VASILECAS, O. Survey on ontology languages. **Perspectives in Business Informatics Research**, pp.124-141, 2011.

KANG, K.C., COHEN, S.G., HESS, J.A., NOVAK, W.E. and PETERSON, A.S. Feature-oriented domain analysis (FODA) feasibility study (No. CMU/SEI-90-TR-21). Carnegie-Mellon Univ. Pittsburgh Pa Software Engineering Inst., 1990.

KARP, P.D., CHAUDHRI, V.K. and THOMERE, J., 1999. XOL: An XML-based ontology exchange language. Available in: http://xml.coverpages.org/xol-03.html.

KAYMAK, U., MANS, R., VAN DE STEEG, T., DIERKS, M. On process mining in health care. **Systems, Man, and Cybernetics (SMC)**, 2012 IEEE International Conference on, pp. 1859-1864. IEEE, 2012.

KHAN, J.A. and KUMAR, S. Deep analysis for development of RDF, RDFS and OWL ontologies with protege. **Reliability, Infocom Technologies and Optimization** (ICRITO) (Trends and Future Directions), 2014 3rd International Conference on (pp. 1-6). IEEE, 2014.

KIETZ, J.U., MAEDCHE, A., & VOLZ, R. A method for semi-automatic ontology acquisition from a corporate intranet. In N. Aussenac-Gilles, B. Biébow, & S. Szulman (Eds.), **Proceedings 51 of EKAW'00 Workshop on Ontologies and Texts, CEUR Workshop**, Juan-Les-Pins, France, pp. 4.1-4.14. Amsterdam, The Netherlands, 2000. Retrieved October 23, 2006, from http://CEUR-WS.org/Vol-51/

KÖSCHE, J.C. Fundamentos de metodologia científica: teoria da ciência e iniciação à pesquisa. Petrópolis, RJ; Vozes, 2011.

KOTHARI, C.R. Research methodology: Methods and techniques. New Age International, 2004.

KUMAR, A. and YAO, W. Design and management of flexible process variants using templates and rules**. Computers in Industry**, 63(2), pp.112-130, 2012.

LA ROSA, M., DUMAS, M., TER HOFSTEDE, A.H., MENDLING, J. and GOTTSCHALK, F. Beyond Control-Flow: Extending Business Process Configuration to Roles and Objects. ER (pp. 199-215), 2008.

LA ROSA, M. Managing variability in process-aware information systems (Doctoral dissertation, Queensland University of Technology), 2009.

LA ROSA, M., DUMAS, M. and TER HOFSTEDE, A.H. Modelling Business Process Variability for Design-Time Configuration. **Handbook of Research on Business Process Modelling** (pp. 204-228). IGI Global, 2009.

LA ROSA, M., VAN DER AALST, W.M., DUMAS, M. AND TER HOFSTEDE, A.H. Questionnaire-based variability modeling for system configuration. **Software and Systems Modeling**, 8(2), pp.251-274, 2009.

LA ROSA, M., DUMAS, M., TER HOFSTEDE, A.H. AND MENDLING, J. Configurable multi-perspective business process models. **Information Systems**, 36(2), pp.313-340, 2011.

LA ROSA, M., DUMAS, M., UBA, R. and DIJKMAN, R. Business process model merging: An approach to business process consolidation. **ACM Transactions on Software Engineering and Methodology** (TOSEM), *22*(2), p.11, 2013.

LA ROSA, M., VAN DER AALST, W.M.P., DUMAS, M. & MILANI, F.P. Business process variability modelling: A survey. **ACM Computing Surveys**, 50(1), 2:1-2:45, 2017.

LACERDA, D.P., DRESCH, A., PROENÇA, A., ANTUNES JUNIOR, J.A.V. Design Science Research: Método de Pesquisa para a Engenharia de Produção. Gestão e Produção, vol.20 no.4 São Carlos, Nov. 26, 2013.

LAKSHMANAN, G.T. and KHALAF, R. Leveraging Process-Mining Techniques. **IT Professional**, 15(5), pp.22-30, 2013.

LEE, C.S. and WANG, M.H., Ontology-based intelligent healthcare agent and its application to respiratory waveform recognition. **Expert Systems with Applications** 33, no. 3, pp. 606-619, 2007.

LI, C., REICHERT, M. and WOMBACHER, A. Discovering reference process models by mining process variants. ICWS'08, Beijing, 2008a.

LI, C., REICHERT, M. and WOMBACHER, A. Mining based on learning from process change logs. **International Conference on Business Process Management** (pp. 121-133). Springer, Berlin, Heidelberg, 2008b.

LI, C., REICHERT, M. and WOMBACHER, A. The MinAdept clustering approach for discovering reference process models out of process variants. **International Journal of Cooperative Information Systems**, *19*(03n04), pp.159-203, 2010.

LI, C., REICHERT, M. and WOMBACHER, A. Mining business process variants: Challenges, scenarios, algorithms. **Data & Knowledge Engineering**, 70(5), pp.409-434, 2011.

LINDLEY, R. I., WARLOW, C. P., WARDLAW, J. M., DENNIS, M. S., SLATTERY, J., and SANDERCOCK, P.A. Interobserver reliability of a clinical classification of acute cerebral infarction. **Stroke**, 24(12), 1801-1804, 1993.

LYDEN, P., LU, M., JACKSON, C., MARLER, J., KOTHARI, R., BROTT, T. and ZIVIN, J. Underlying structure of the National Institutes of Health stroke scale. **Stroke**, 30(11), 2347-2354, 1999.

MAHMOD, N.M. and CHIEW, W.Y. Structural similarity of business process variants. In Open Systems (ICOS), **2010 IEEE Conference on** (pp. 17-22). IEEE, 2010, December.

MAITA, A.R., FANTINATO, M., PERES, S.M., THOM, L.H. and HUNG, P.C. 2017. Mining unstructured processes: An exploratory study on a distance learning domain. In Neural Networks (IJCNN), 2017 International Joint Conference on (pp. 3240-3247). IEEE, 2017.

MANS, R., REIJERS, H., BERENDS, H., BANDARA, W. and ROGIER, P. Business process mining success. **Proceedings of the 21st European Conference on Information Systems.** AIS Electronic Library (AISeL), 2013.

MARCH, S.T., SMITH, G.F. Design and Natural Science Research on Information Technology. Decision Support Systems 15, 251-266, 1995.

MARTINEZ-GIL, J. Automated knowledge base management: A survey. **Computer Science Review**, 18, pp.1-9, 2015.

MARTINS, S.C.O., FREITAS, G.R.D., PONTES-NETO, O.M., PIERI, A., MORO, C.H.C., JESUS, P.A.P.D., LONGO, A., EVARISTO, E.F., CARVALHO, J.J.F.D., FERNANDES, J.G.

AND GAGLIARDI, R.J. Guidelines for acute ischemic stroke treatment: part II: stroke treatment. **Arquivos de neuro-psiquiatria**, 70(11), pp.885-893, 2012.

MARUSTER, L., WEIJTERS, A., VAN DER AALST, W.M.P. and VAN DEN BOSCH, A. A Rule-Based Approach for Process Discovery: Dealing with Noise and Imbalance in Process Logs. Data Mining and Knowledge Discovery **13**(1), 67–87, 2006.

McGUINNESS, D.L. and VAN HARMELEN, F. OWL web ontology language overview. **W3C recommendation**, 10(10), p.2004.

MEERKAMM, S. and JABLONSKI, S. Configurable process models: experiences from a medical and an administrative case study. **ECIS** (p. 33), 2011, October.

MENÁRGUEZ-TORTOSA, M. and FERNÁNDEZ-BREIS, J.T. OWL-based reasoning methods for validating archetypes. **Journal of biomedical informatics**, 46(2), pp.304-317, 2013.

MINAYO, Maria Cecília de Souza. **O desafio do conhecimento.** São Paulo: Hucitec,1993.

MITCHELL, T.M. The discipline of machine learning (Vol. 3). Carnegie Mellon University, School of Computer Science, Machine Learning Department, 2006.

MOHRI, M., ROSTAMIZADEH, A. and TALWALKAR, A. Foundations of machine learning. **MIT press**., 2012.

MONTANI, S. and LEONARDI, G. Retrieval and clustering for supporting business process adjustment and analysis. **Information Systems**, 40, pp.128-141, 2014.

MURATA, T. Petri nets: Properties, analysis and applications. **Proceedings of the IEEE**, 77(4), pp.541-580, 1989.

MUSEN, M.A.: Dimensions of knowledge sharing and reuse. Computers and Biomedical Research 25: 435-467, 1992.

NECHES, R., FIKES, R. E., FININ, T., GRUBER, T., PATIL, R., SENATOR, T., & SWARTOUT, W.R. Enabling technology for knowledge sharing. **AI magazine**, 12(3), 36-56, 1991.

NIH STROKE SCALE. Available in: http://www.nihstrokescale.org/

NOY, N.F. and MUSEN, M.A. An algorithm for merging and aligning ontologies: Automation and tool support. **Proceedings of the Workshop on Ontology Management at the Sixteenth National Conference on Artificial Intelligence** (AAAI-99) (pp. 1999-0799), 1999.

NOY, N.F. and MUSEN, M.A. Algorithm and tool for automated ontology merging and alignment. **Proceedings of the 17th National Conference on Artificial Intelligence** (AAAI-00). Available as SMI technical report SMI-2000-0831, 2000.

OBITKO, M. Translations between ontologies in multi-agent systems. Ph. D. Dissertation, Czech Technical University, Faculty of Electrical Engineering, 2007.

OLIVEIRA, S.R. and ZAIANE, O.R. Privacy preserving frequent itemset mining. Proceedings of the IEEE international conference on Privacy, security and data mining-Volume 14 (pp. 43-54). Australian Computer Society, Inc., 2002.

PANNEERSELVAM, R. Research methodology. **PHI Learning Pvt**. Ltd., 2014.

PASSOS, V., ISHITANI, L.H., FRANCO, G.C., LANA, G.C., ABREU, D.M., MARINHO, M.D.F. and FRANÇA, E.B., 2016. Consistent declining trends in stroke mortality in Brazil: mission accomplished? **Arquivos de neuro-psiquiatria**, 74(5), pp.376-381.

PEDRINACI, C. and DOMINGUE, J. Towards an ontology for process monitoring and mining, CEUR Workshop Proceedings, vol. 251, pp. 76-87, 2007.

PEFFERS, K., TUUNANEN, T., ROTHENBERGER, M.A. and CHATTERJEE, S. A design science research methodology for information systems research. **Journal of management information systems**, 24(3), pp.45-77, 2007.

PINTO, H.S., GÓMEZ-PÉREZ, A. and MARTINS, J.P. Some issues on ontology integration. **IJCAI and the Scandinavian AI Societies**. CEUR Workshop Proceedings; 1999.

PERCEPTIVE SOFTWARE, 2017. Available in: http://www.perceptivesoftware.co.uk/content/psw/www-perceptivesoftware-com/en_gb/products/perceptive-process/process-mining.html

PROCESS MINING, 2017. Available in http://www.processmining.org/online/decisionmining

PROM, 2017. Available in http://www.promtools.org/doku.php.

PROM IMPORT FRAMEWORK. http://www.promtools.org/promimport/. Accessed: 25.03.17.

PROPER, H.A. and VAN DER WEIDE, T.P. Information Disclosure in Evolving Information Systems: Taking a Shot at a Moving Target. **Data & Knowledge Engineering**, 15(2), pp.135-168, 1995.

PROTÉGÉ. Available in https://protege.stanford.edu/

PUHLMANN, F., SCHNIEDERS, A., WEILAND, J. and WESKE, M. Variability mechanisms for process models. PESOA-Report TR, 17, pp.10-61, 2005.

QUAGLINI, S. Compliance with clinical practice guidelines. Stud Health Technol Inform, 139, pp.160-79, 2008.

QUINLAN, J.R. C4. 5: programs for machine learning. Elsevier, 2014.

REBUGE, Á. and FERREIRA, D.R. Business process analysis in healthcare environments: A methodology based on process mining. **Information systems**, 37(2), pp.99-116, 2012.

REICHERT, M. and WEBER, B. Enabling flexibility in process-aware information systems: challenges, methods, technologies. **Springer Science & Business Media**. (Chapter 5), 2012.

REINHARTZ-BERGER, I., SOFFER, P., STURM, A. Organizational reference models: Supporting an adequate design of local business processes. **International Journal of Business Process Integration and Management**, *4*(2), pp.134-149, 2009.

REINHARTZ-BERGER, I. and STURM, A. Enhancing UML models: a domain analysis approach. **Selected Readings on Database Technologies and Applications**, p.330, 2008.

REINHARTZ-BERGER, I., SOFFER, P. AND STURM, A. Extending the adaptability of reference models. **IEEE Transactions on Systems, Man, and Cybernetics-Part A: Systems and Humans**, 40(5), pp.1045-1056, 2010.

REISIG, W. and ROZENBERG, G. Lectures on Petri Nets I: Basic Models: Advances in Petri Nets (Lecture Notes in Computer Science), 1998.

ROGGE-SOLTI, A., MANS, R.S., VAN DER AALST, W.M.P. and WESKE, M. Repairing event logs using stochastic process models (Vol. 78). Universitätsverlag Potsdam, 2013.

ROGGE-SOLTI, A. and WESKE, M. Prediction of business process durations using non-Markovian stochastic Petri nets. **Information Systems**, 54, pp.1-14, 2015.

ROJAS, E., MUNOZ-GAMA, J., SEPÚLVEDA, M. and CAPURRO, D. Process mining in healthcare: A literature review. **Journal of biomedical informatics**, 61, pp.224-236, 2016.

ROSEMANN, M. and VAN DER AALST, W.M.P. A configurable reference modelling language. **Information Systems**, 32(1), pp.1-23, 2007.

ROZINAT, A., MANS, R.S. and VAN DER AALST, W.M.P. Mining CPN models: discovering process models with data from event logs. **Workshop and Tutorial on Practical Use of Coloured Petri Nets and the CPN**, 2006.

ROZINAT, A. and VAN DER AALST, W.M.P. Decision mining in business processes. Beta, Research School for Operations Management and Logistics, 2006b.

ROZINAT, A., DE JONG, I.S.M., GÜNTHER, C.W., VAN DER AALST, W.M.P. Process mining applied to the test process of wafer scanners in ASML. **Systems, Man, and Cybernetics, Part C: Applications and Reviews**. IEEE Transactions on vol. 39, no. 4, pp. 474-479, 2009.

SERNA, E. and SERNA, A. Ontology for knowledge management in software maintenance. **International Journal of Information Management**, 34(5), pp.704-710, 2014.

SIRIN, E., PARSIA, B., GRAU, B.C., KALYANPUR, A. and KATZ, Y. Pellet: A practical owl-dl reasoner. Web Semantics: science, services and agents on the World Wide Web, 5(2), pp.51-53, 2007.

SHARMAN, R., KISHORE, R. and RAMESH, R. Ontologies: a handbook of principles, concepts and applications in information systems. Springer's Integrated Series in Information Systems, eds., 2007.

SCHONENBERG, H., MANS, R., RUSSELL, N., MULYAR, N. and VAN DER AALST, W.M.P. Process flexibility: A survey of contemporary approaches. In Advances in Enterprise Engineering I (pp. 16-30). Springer, Berlin, Heidelberg, 2008.

SONG, M., GÜNTHER, C.W. and VAN DER AALST, W.M.P. Trace clustering in process mining. **International Conference on Business Process Management** (pp. 109-120). Springer, Berlin, Heidelberg, 2008.

SONG, M. and VAN DER AALST, W.M. Towards comprehensive support for organizational mining. **Decision Support Systems**, 46(1), pp.300-317, 2008.

SONG, F., ZACHAREWICZ, G. and CHEN, D. An ontology-driven framework towards building enterprise semantic information layer. **Advanced Engineering Informatics**, 27(1), pp.38-50, 2013.

STAAB, S., SCHNURR, H.P., STUDER, R. and SURE, Y. Knowledge processes and ontologies. IEEE Intelligent Systems, 16(1), 26–34, 2001.

SU, X. and ILEBREKKE, L. A comparative study of ontology languages and tools. **CAiSE**, vol. 2002, pp. 761-765, 2002.

SULTER, G., STEEN, C., and DE KEYSER, J. Use of the Barthel index and modified Rankin scale in acute stroke trials. **Stroke**, 30(8), 1538-1541, 1999.

SWARTOUT, B., RAMESH, P., KNIGHT, K. and RUSS, T. (1997). Toward distributed use of large-scale ontologies. In A. Farquhar, M. Gruninger, A. Gómez-PÉREZ, M. USCHOLD, & P. VAN DER VET (Eds.), AAAI'97 Spring Symposium on Ontological Engineering (pp. 138-148). Stanford University.

SWARTOUT, W. and TATE, A. Ontologies. **IEEE Intelligent Systems**, 14(1):18-19, 1999.

SZABÓ, I. and K., VARGA, K. Knowledge-Based Compliance Checking of Business Processes. On the Move to Meaningful Internet Systems: OTM 2014 Conferences, pp. 597-611, Springer Berlin Heidelberg, 2014.

TAGARIS, A., KONNIS, G., BENETOU, X., DIMAKOPOULOS, T., KASSIS, K., ATHANASIADIS, N., RÜPING, S., GROSSKREUTZ, H. and KOUTSOURIS, D. Integrated Web Services Platform for the facilitation of fraud detection in health care e-government services. **Information Technology and Applications in Biomedicine**, 2009. ITAB 2009. 9th International Conference on, pp. 1-4. IEEE, 2009.

TANKELEVIČIENĖ, L. Ontology and Ontology Engineering: Analysis of Concepts, Classifications and Potential Use in E-Learning Context. Technical Report MII-SED-08-01, 2008.

TAYE, M.M. The State of the Art: Ontology Web-Based Languages: XML Based, **arXiv** preprint arXiv: 1006.4563, 2010.

TORRES, V., ZUGAL, S., WEBER, B., REICHERT, M., AYORA, C. and PELECHANO, V. A qualitative comparison of approaches supporting business process variability. **International Conference on Business Process Management** (pp. 560-572). Springer Berlin Heidelberg, 2012, September.

TUDORACHE, T., NYULAS, C., NOY, N.F. and MUSEN, M.A. WebProtégé: A collaborative ontology editor and knowledge acquisition tool for the web. **Semantic web**, 4(1), pp.89-99, 2013.

TURNER, C.J., TIWARI, A., OLAIYA, R. and XU, Y. Process mining: from theory to practice. **Business Process Management Journal**, 18(3), pp.493-512, 2012.

USCHOLD, M. and KING, M. Towards a methodology for building ontologies. In D. Skuce (Ed.), IJCAI'95 Workshop on Basic Ontological Issues in Knowledge Sharing (pp. 6.1-6.10), Montreal, Canada, 1995.

VAISHNAVI, V., KUECHLER, W., and PETTER, S. Design Science Research in Information Systems. January 20, 2004 (created in 2004 and updated until 2015 by Vaishnavi, V. and Kuechler, W.); last updated (by Vaishnavi, V. and Petter, S.), December 20, 2017. Available in: http://www.desrist.org/design-research-in-information-systems/.

VALENÇA, G., ALVES, C., ALVES, V. & NIU, N. A systematic mapping study on business process variability. **International Journal of Computer Science & Information Technology**, 5(1), p.1, 2013.

VAN DER AALST, W.M.P. (2004) Business Process Management Demystified: A Tutorial on Models, Systems and Standards for Workflow Management. In: Desel J., Reisig W., Rozenberg G. (eds) Lectures on Concurrency and Petri Nets. ACPN 2003. Lecture Notes in Computer Science, vol 3098. Springer, Berlin, Heidelberg

VAN DER AALST, W.M.P. and WEIJTERS, A.J.M.M. Process mining: a research agenda. **Computers in industry**, 53(3), pp.231-244, 2004.

VAN DER AALST, W.M.P. and DE MEDEIROS, A.K.A. Process mining and security: Detecting anomalous process executions and checking process conformance. **Electronic Notes in Theoretical Computer Science**, 121, pp.3-21, 2005.

VAN DER AALST, W.M.P., REIJERS, H.A. and SONG, M. Discovering social networks from event logs. **Computer Supported Cooperative Work** (CSCW), 14(6), pp.549-593, 2005.

VAN DER AALST, W., DREILING, A., GOTTSCHALK, F., ROSEMANN, M., and JANSEN-VULLERS, M. Configurable Process Models as a Basis for Reference Modeling. Proceedings of the Business Process Management 2005 Workshops, E. Kindler and M. Nüttgens, Eds. Springer, 76–82, 2006.

VAN DER AALST, W.M.P. Trends in business process analysis. **Proceedings of the 9th International Conference on Enterprise Information Systems** (ICEIS), pp. 12-22, 2007.

VAN DER AALST, W.M., REIJERS, H.A., WEIJTERS, A.J., VAN DONGEN, B.F., DE MEDEIROS, A.A., SONG, M. and VERBEEK, H.M.W. Business process mining: An industrial application. **Information Systems**, 32(5), pp.713-732, 2007.

VAN DER AALST, W.M.P., DUMAS, M., GOTTSCHALK, F., TER HOFSTEDE, A.H., LA ROSA, M. and MENDLING, J. Correctness-preserving configuration of business process models. **International Conference on Fundamental Approaches to Software Engineering** (pp. 46-61). Springer, Berlin, Heidelberg, 2008, March.

VAN DER AALST, W.M.P. Process Discovery: Capturing the invisible. **IEEE Computational Intelligence Magazine**, February 2010.

VAN DER AALST, W.M., DUMAS, M., GOTTSCHALK, F., TER HOFSTEDE, A.H., LA ROSA, M. and MENDLING, J. Preserving correctness during business process model configuration. **Formal Aspects of Computing,** *22*(3-4), pp.459-482, 2010.

VAN DER AALST, W.M.P. Process Mining Discovery, Conformance and Enhancement of Business Processes, Springer Berlin Heidelberg, 2011.

VAN DER AALST, W., ADRIANSYAH, A., DE MEDEIROS, A.K.A., ARCIERI, F., BAIER, T., BLICKLE, T., BOSE, J.C., VAN DEN BRAND, P., BRANDTJEN, R., BUIJS, J. and BURATTIN, A. Process mining manifesto. International Conference on Business Process Management (pp. 169-194). Springer, Berlin, Heidelberg, 2011.

VAN DER AALST, W.M.P. Service Mining: Using Process Mining to Discover, Check, and Improve Service Behavior. **IEEE Transactions on Service Computing**, vol. 6, no. 4, 2012.

VAN DER AALST, W.M. and DUSTDAR, S. Process mining put into context. **IEEE Internet Computing**, 16(1), pp.82-86, 2012.

VAN DER AALST, W.M., 2013. Business process management: a comprehensive survey. Hindawi Publishing Corporation, **ISRN Software Engineering**, Vol. 2013. Available: http://dx.doi.org/10.1155/2013/507984

VAN DER SPOEL, S., VAN KEULEN, M. and AMRIT, C. Process prediction in noisy data sets: a case study in a Dutch hospital. International Symposium on Data-Driven Process Discovery and Analysis (pp. 60-83). Springer, Berlin, Heidelberg, 2012.

VAN HEIJST, G., SCHREIBER, A.T. and WIELING, B.J. Using explicit ontologies in KBS development. **International journal of human-computer studies**, 46(2-3), pp.183-292, 1997.

VERMA, A. An abstract framework for ontology evaluation. In **Data Science and Engineering (ICDSE), 2016 International Conference on** (pp. 1-6). IEEE, 2016.

YAO, J. and GU, M. Conceptology: Using Concept Map for Knowledge Representation and Ontology Construction. **JNW**, 8(8), pp.1708-1712, 2013.

YOO, S., CHO, M., KIM, E., KIM, S., SIM, Y., YOO, D., HWANG, H. and SONG, M. Assessment of hospital processes using a process mining technique: Outpatient process analysis at a tertiary hospital. **International journal of medical informatics**, 88, pp.34-43, 2016.

WEIJTERS, A.J.M.M., VAN DER AALST, W.M.P. and DE MEDEIROS, A.A. Process mining with the heuristics miner-algorithm. **Technische Universiteit Eindhoven**, Tech. Rep. WP, 166, pp.1-34, 2006.

WESKE, M. Business process management: concepts, languages, architectures. **Springer Science & Business Media**, 2012.

WITTEN, I.H, FRANK, E. Data mining: practical machine learning tools and techniques. Elsevier, 2nd ed., 2005.

W3C Recommendation. Available in https://www.w3.org/TR/2004/REC-rdf-concepts-20040210/

WORLD HEALTH ORGANIZATION. Available in: http://www.who.int/gho/mortality_burden_disease/causes_death/top_10/en/

WORLD STROKE ORGANIZATION. Available in: http://www.world-stroke.org/

WU, H., LIU, J., YE, D., WEI, J. and ZHONG, H. Scalable Horn-Like Rule Inference of Semantic Data Using MapReduce. **International Conference on Knowledge Science, Engineering and Management** (pp. 270-277). Springer, Cham., 2014, October.

ZHOU, L. Ontology learning: state of the art and open issues. **Information Technology and Management**, 8(3), pp.241-252, 2007.