



UNIVERSIDADE ESTADUAL PAULISTA “JÚLIO DE MESQUITA FILHO”

Programa de Pós-Graduação em Ciência da Computação

LUKAS RIEHL FIGUEIREDO

**MAPEAMENTO DE MODELOS DE
PROCESSOS DE NEGÓCIO PARA
ONTOLOGIAS, INCLUINDO SISTEMA
DE CONSULTAS**

Rio Claro - SP

2018

LUKAS RIEHL FIGUEIREDO

MAPEAMENTO DE MODELOS DE PROCESSOS DE NEGÓCIO PARA ONTOLOGIAS, INCLUINDO SISTEMA DE CONSULTAS

Dissertação apresentada como parte dos requisitos para obtenção do título de Mestre em Ciência da Computação, junto ao Programa de Pós-Graduação em Ciência da Computação, área de concentração em Computação Aplicada, linha de Pesquisa Sistemas de Informação, da Universidade Estadual Paulista “Júlio de Mesquita Filho”.

Orientadora: Profa. Dra. Hilda Carvalho de Oliveira

Rio Claro - SP

2018

Figueiredo, Lukas Riehl.

Mapeamento de modelos de processos de negócio para ontologias, incluindo sistema de consultas / Lukas Riehl Figueiredo. -- Rio Claro, 2018

139 f. : il., tabs.

Orientador: Hilda Carvalho de Oliveira

Dissertação (mestrado) – Universidade Estadual Paulista “Júlio de Mesquita Filho”, Instituto de Biociências, Letras e Ciências Exatas

1. Computação. 2. Modelagem de informação. 3. Ontologias (Recuperação da informação) I. Universidade Estadual Paulista "Júlio de Mesquita Filho". Instituto de Biociências, Letras e Ciências Exatas. II. Título.

CDU – 518.72

Ficha catalográfica elaborada pela Biblioteca do IBILCE
UNESP - Campus de São José do Rio Preto

LUKAS RIEHL FIGUEIREDO

MAPEAMENTO DE MODELOS DE PROCESSOS DE NEGÓCIO PARA ONTOLOGIAS, INCLUINDO SISTEMA DE CONSULTAS

Dissertação apresentada como parte dos requisitos para obtenção do título de Mestre em Ciência da Computação, junto ao Programa de Pós-Graduação em Ciência da Computação, área de concentração em Computação Aplicada, linha de Pesquisa Sistemas de Informação, da Universidade Estadual Paulista “Júlio de Mesquita Filho”.

Comissão Examinadora

Profa. Dra. Hilda Carvalho de Oliveira
UNESP - Rio Claro
Orientadora

Prof. Dr. Ivan Rizzo Guilherme
UNESP - Rio Claro

Prof. Dr. Marcelo Medeiros Eler
USP (EACH) - São Paulo

Rio Claro - SP

1º de fevereiro de 2018

Agradecimentos

Agradeço à minha esposa, Juma, pelo companheirismo, apoio e incentivo durante a elaboração deste trabalho.

Aos meus pais, Renata e Nelson, agradeço pelo apoio e incentivo em toda a minha vida profissional, acadêmica e acima de tudo pessoal.

Agradeço à minha orientadora, Profa. Dra. Hilda Carvalho de Oliveira, por ter acreditado e investido seu tempo em mim, e pelas orientações que serviram e servirão não apenas para meu crescimento na vida acadêmica, mas também pessoal.

Aos colegas de Mestrado, agradeço pela amizade e pela construtiva interação durante as disciplinas e outros trabalhos acadêmicos.

Ao Fernando Aparecido Nogueira, agradeço pela disponibilidade, apoio e discussões construtivas que contribuíram para a elaboração deste trabalho.

À Maria Anatania Pucci, agradeço pela disponibilidade, apoio e discussões construtivas na parte de modelos de processos de negócio.

Ao Alessandro Viola Pizzoleto, agradeço pela disponibilidade, apoio e discussões construtivas na parte de ontologias.

“Quando acho que cheguei ao ponto máximo,
descubro que é possível superá-lo.”
(Ayrton Senna)

RESUMO

Modelos de processos de negócio têm sido utilizados em ambientes organizacionais para o entendimento da cadeia de processos, visando melhorias técnicas, gerenciais e estratégicas. Esses modelos abrangem variadas informações sobre os recursos utilizados e interações entre diferentes setores e pessoas, evidenciando interdependências entre os processos. Contudo, esses modelos podem apresentar problemas de legibilidade e dificuldades de navegação à medida em que se tornam extensos. A representação do conhecimento implícito é complexa e não está padronizada; as interdependências nem sempre são fáceis de serem compreendidas. O uso de ontologias tem aberto uma perspectiva complementar para mitigar esses problemas e prover processos com semânticas acessíveis por máquina. As ontologias contribuem para a conceituação e organização da informação embutida e desestruturada que se encontra presente nos processos de negócio e que deve ser explorada. As ontologias são utilizadas para estruturar o conhecimento implícito que está presente nos processos de negócio, possibilitando a compreensão desse conhecimento por máquina. As ontologias também contribuem para o alinhamento das visões das equipes de negócio e de Tecnologia da Informação da organização. Nesse contexto, este trabalho apresenta um processo sistemático para gerar uma ontologia a partir de um modelo de processos de negócio em BPMN v2.0, permitindo consultar informações sobre o modelo. Uma ferramenta denominada PM2ONTO foi então desenvolvida, visando gerar a ontologia em OWL automaticamente e disponibilizar uma interface para consultas predefinidas, elaboradas na linguagem SPARQL. Os modelos de processos de negócio devem ser exportados para arquivos em XPDL v2.2, os quais serão as entradas do sistema PM2ONTO. As ontologias resultantes ficam disponíveis em um banco de dados para consultas e também podem ser manipuladas por outras diferentes ferramentas. A avaliação do processo sistemático apresentado neste trabalho foi feita com base em estudos de caso, usando modelos de processos de negócio de diferentes áreas e diferentes níveis de complexidade.

Palavras-chave: Modelos de processos de negócio. Documentação de processos de negócio. Boas práticas de modelagem. Ontologias. Ontologia de processos. Construção de ontologias. BPMN. XPDL. OWL. SPARQL.

ABSTRACT

Aiming technical, managerial and strategic improvements, business process models have been used in organizational environments to understand the process chain. Although these models may present readability problems and navigation difficulties as they become extensive, they can cover many information about resources and interactions between different sectors and people, showing interdependencies among the processes. Interdependencies are not always easy to understand because of its complex and not standardized knowledge. The use of ontologies has opened a complementary perspective to mitigate these problems and to provide processes with machine-accessible semantics. Also, they contribute in the conceptualization and organization of the embedded and unstructured information that is present in the business processes and that must be explored. In order to ease machine knowledge, ontologies are used to structure the implicit knowledge in business processes. They also contribute to the alignment of the organization's business and IT teams. In this context, this work presents a systematic process to generate an ontology from a business process model in BPMN v2.0, allowing to query information about the model. In this way, PM2ONTO tool was developed aiming to generate the ontology in OWL automatically and to provide an interface for predefined queries elaborated in the SPARQL language. Business process models should be exported to files in XPDL v2.2, which will be the inputs of the PM2ONTO system. The resulting ontologies stay available in a database for queries and can also be manipulated by other different tools. The systematic process evaluation presented in this work was made based on case studies, using business process models of different areas and different levels of complexity.

Keywords: *Business process models. Documentation of business processes. Good modeling practices. Ontologies. Process ontologies. Construction of ontologies. BPMN. XPDL. OWL. SPARQL.*

LISTA DE FIGURAS

	Página
Figura 1 - Visão geral do processo de geração de ontologias e do sistema de consultas.....	17
Figura 2 - Visão geral das atividades desenvolvidas neste trabalho.	19
Figura 3 - Evolução das linguagens/notações voltadas a processos de negócio.	28
Figura 4 - Exemplo de modelo de processo na notação BPMN.	30
Figura 5 - Ciclo de vida de SBPM.	40
Figura 6 - Metamodelo do pacote XPDL.	42
Figura 7 - Exemplo de um modelo de processos de negócio em XPDL.	43
Figura 8 - Hierarquia dos tipos de ontologia.	49
Figura 9 - Parte de uma ontologia de domínio de tipos de problemas e falhas em SGBDs.	50
Figura 10 - Exemplo de uma declaração simples em RDF, em forma de grafo.	54
Figura 11 - Parte de uma ontologia escrita na linguagem OWL.	56
Figura 12 - Exemplo de uma consulta em SPARQL.....	58
Figura 13 - Tradução de um cabeçalho em XPDL para a ontologia oXPDL.	60
Figura 14 - Metamodelo da ontologia de processo de Ternai, Török e Varga (2016).	63
Figura 15 - Exemplo de atributo estendido definido para uma atividade utilizando o modelador Bizagi.....	66
Figura 16 - Visão geral do processo sistemático para a geração das ontologias.	67
Figura 17 - Visão geral da ferramenta PM2ONTO.....	73
Figura 18 - Funcionalidades da ferramenta PM2ONTO.	75
Figura 19 - Tela da ferramenta PM2ONTO para a geração de ontologias.....	75
Figura 20 - Parte do metamodelo de classes Meta2Onto.	77
Figura 21 - Categorização das classes para a ontologia.	80
Figura 22 - Código SPARQL da consulta avançada nº 8.	87
Figura 23 - Tela do sistema PM2ONTO para seleção de uma ontologia para consultas.....	88
Figura 24 - Tela do sistema PM2ONTO para seleção da consulta e respectivo filtro.	88
Figura 25 - Atividades para a geração de ontologias.	93
Figura 26 - Modelo de processos de negócio "Contas a Pagar".	99
Figura 27 - Parte da ontologia gerada para o modelo "Contas a Pagar".....	101
Figura 28 - Subprocesso "Aprovação de Viagem" do modelo "Processo de Viagem".	104
Figura 29 - Subprocesso "Completar dados dos novos pedidos" do modelo "Compra de Livros".	109
Figura 30 - Diagrama de requisitos de decisão proposto na notação DMN.	130
Figura 31 - Exemplo de modelo de plano de caso de uso.....	132
Figura 32 - Melhores práticas para as notações BPMN, DMN e CMMN.	134

LISTA DE TABELAS

	Página
Tabela 1 - Características principais de diagrama, mapa e modelo.....	25
Tabela 2 - Inconsistências estruturais nos modelos de processos de negócio.....	35
Tabela 3 - Inconsistências de layout nos modelos de processos de negócio.....	36
Tabela 4 - Inconsistências de rotulagem nos modelos de processos de negócio.....	36
Tabela 5 - Níveis de classificação dos elementos BPMN com base em sua documentação.....	38
Tabela 6 - Heurísticas de negócio para documentar as atividades.....	38
Tabela 7 - Correspondência entre elementos de BPMN e XPDL.....	45
Tabela 8 - Etapas e atividades da metodologia Methontology.....	52
Tabela 9 - Etapas da metodologia para desenvolvimento de ontologias no projeto TOVE.....	53
Tabela 10 - Heurísticas de negócio utilizadas neste trabalho.....	71
Tabela 11 - Heurísticas de requisitos utilizadas neste trabalho.....	72
Tabela 12 - Relações presentes no metamodelo Meta2Onto.....	76
Tabela 13 - Correspondência entre os elementos XPDL e as classes do metamodelo Meta2Onto.....	78
Tabela 14 - Propriedades básicas de objeto definidas para a ontologia.....	82
Tabela 15 - Propriedades avançadas de objeto para a ontologia.....	83
Tabela 16 - Síntese das consultas avançadas definidas no sistema PM2ONTO v1.0.....	85
Tabela 17 - Elementos/relações dos modelos em BPMN testados nas ontologias geradas.....	94
Tabela 18 - Plano de avaliação das consultas SPARQL (continua...).	95
Tabela 19 - Modelo “Contas a pagar”: número de fluxos <i>versus</i> classes geradas.....	102
Tabela 20 - Modelo “Contas a pagar”: elementos de conexão <i>versus</i> axiomas gerados.....	102
Tabela 21 - Modelo “Processo de Viagem”: número de fluxos <i>versus</i> classes geradas.....	107
Tabela 22 - Modelo “Processo de Viagem”: elementos de conexão <i>versus</i> axiomas gerados.....	107
Tabela 23 - Modelo “Compra de Livros”: número de fluxos <i>versus</i> classes geradas.....	112
Tabela 24 - Modelo “Compra de Livros”: elementos de conexão <i>versus</i> axiomas gerados.....	112
Tabela 25 - Visão geral dos resultados dos três estudos de caso.....	113
Tabela 26 - Representações para as atividades na notação BPMN.....	124
Tabela 27 - Representações de eventos na notação BPMN.....	125
Tabela 28 - Representações de <i>gateways</i> na notação BPMN.....	126
Tabela 29 - Estruturas básicas da notação BPMN.....	127
Tabela 30 - Artefatos presentes na notação BPMN.....	127
Tabela 31 - Representações de subprocessos na notação BPMN.....	128
Tabela 32 - Comparação entre as notações BPMN, CMMN e DMN.....	133

LISTA DE ABREVIATURAS E SIGLAS

API	<i>Application Programming Interface</i>
ARIS	<i>Architecture of Integrated Information Systems</i>
BH	<i>Business Heuristic</i>
BPAL	<i>Business Process Abstract Language</i>
BPM	<i>Business Process Management</i>
BPMI	<i>Business Process Management Institute</i>
BPMN	<i>Business Process Model and Notation</i>
BPMS	<i>Business Process Management System</i>
CMMN	<i>Case Management Model and Notation</i>
DAWG	<i>Data Access Working Group</i>
DFD	Diagramas de Fluxos de Dados
DMN	<i>Decision Model Notation</i>
EML	<i>Enterprise Modelling Language</i>
EPC	<i>Event-driven Process Chain</i>
HN	Heurística de Negócio
HR	Heurística de Requisitos
HTTP	<i>Hypertext Transfer Protocol</i>
IB	Instituto de Biociências
OASIS	<i>Open Advancing Standards for the Information Society</i>
OCL	<i>Object Constraint Language</i>
OMG	<i>Object Management Group</i>
OWL	<i>Web Ontology Language</i>
PLN	Processamento de Linguagem Natural
PM2ONTO	<i>Process Model to Ontology</i>
RIF	<i>Rule Interchange Format</i>
RDF	<i>Resource Description Framework</i>

RDFS	<i>Resource Description Framework Schema</i>
RH	Recursos Humanos
SBP	<i>Semantic Business Process</i>
SBPM	<i>Semantic Business Process Management</i>
SGBD	Sistema de Gerenciamento de Banco de Dados
SKOS	<i>Simple Knowledge Organization System</i>
SRPD	<i>Software Requirements from Process Definitions</i>
STATI	Seção Técnica de Aquisição e Tratamento da Informação
SWS	<i>Semantic Web Services</i>
TI	Tecnologia da Informação
TIC	Tecnologias da Informação e Comunicação
TOVE	<i>TOronto Virtual Enterprise</i>
UML	<i>Unified Modeling Language</i>
URI	<i>Uniform Resource Identifier</i>
W3C	<i>World Wide Web Consortium</i>
Web	<i>World Wide Web</i>
WFMS	<i>Workflow Management System</i>
WfMC	<i>Workflow Management Coalition</i>
WSML	<i>Web Service Modeling Language</i>
XML	<i>eXtensible Markup Language</i>
XPDL	<i>XML Process Definition Language</i>
XSD	<i>XML Schema Definition</i>
XSLT	<i>Extensible Stylesheet Language for Transformation</i>
YAWL	<i>Yet Another Workflow Language</i>

SUMÁRIO

Página

1	INTRODUÇÃO	13
1.1	Objetivos do trabalho	16
1.2	Metodologia de trabalho.....	18
1.3	Organização dos capítulos.....	20
2	MODELOS DE PROCESSOS DE NEGÓCIO	22
2.1	Modelagem de processos de negócio	24
2.2	Visão geral da notação BPMN	27
2.3	Semântica na notação BPMN	31
2.4	Boas práticas de modelagem em BPMN	34
2.5	Modelagem semântica de processos de negócio.....	39
2.6	Processos de negócio e a linguagem XPD.....	41
2.7	Mapeamento de elementos BPMN para elementos XPD.....	44
2.8	Considerações finais.....	45
3	ONTOLOGIAS A PARTIR DE MODELOS DE PROCESSOS DE NEGÓCIO	47
3.1	Visão geral sobre ontologias.....	48
3.2	Metodologias para construção de ontologias	51
3.3	Linguagens para representação de ontologias.....	53
3.4	Linguagem SPARQL para consultas em ontologias.....	57
3.5	Trabalhos relacionados.....	59
3.6	Considerações finais.....	64
4	GERAÇÃO AUTOMÁTICA DE ONTOLOGIAS A PARTIR DE MODELOS DE PROCESSOS DE NEGÓCIO	65
4.1	Crítérios necessários para os modelos de processos de negócio.....	68
4.2	Recomendações para os modelos de processos de negócio	69
4.2.1	Ajustes recomendados com base nas heurísticas de negócio.....	69
4.2.2	Ajustes recomendados com base nas heurísticas de requisitos.....	72
4.3	Processo sistemático para geração das ontologias	73
4.4	Ferramenta PM2ONTO.....	74
4.4.1	Mapeamento das classes do metamodelo para as classes da ontologia	76
4.4.2	Consultas sobre as ontologias geradas.....	84
4.5	Considerações finais.....	89
5	AValiação DO PROCESSO SISTEMÁTICO.....	90
5.1	Aviação do processo de geração das ontologias	92

5.2 Avaliação das consultas sobre as ontologias geradas	95
5.3 Estudo de caso 1: modelo “Contas a Pagar”	98
5.4 Estudo de caso 2: modelo “Processo de Viagem”	103
5.5 Estudo de caso 3: modelo “Compra de Livros”	108
5.6 Avaliação geral dos resultados dos estudos de caso	111
6 CONCLUSÃO E CONSIDERAÇÕES FINAIS	114
REFERÊNCIAS.....	119
 <i>APÊNDICE A – NOTAÇÃO BPMN.....</i>	 <i>124</i>
<i>APÊNDICE B - NOTAÇÕES RELACIONADAS: DMN, CMMN E BPMN</i>	<i>129</i>
<i>APÊNDICE C – METAMODELO META2ONTO (CD-R anexo)</i>	<i>135</i>
<i>APÊNDICE D – FRAMEWORK APACHE JENA.....</i>	<i>136</i>
<i>APÊNDICE E – SISTEMA PM2ONTO: USO E CÓDIGO (CD-R anexo)</i>	<i>137</i>
<i>APÊNDICE F – MODELOS EM BPMN E XPD L (CD-R anexo)</i>	<i>138</i>
<i>APÊNDICE G – ONTOLOGIAS GERADAS (CD-R anexo)</i>	<i>139</i>

1 INTRODUÇÃO

Modelos de processos de negócio são utilizados pelos profissionais da área de negócio para o entendimento da cadeia de processos de um ambiente organizacional, visando melhorias técnicas, gerenciais e estratégicas. Esses modelos contemplam variadas informações sobre os recursos utilizados, além de interações entre diferentes pessoas e setores. Dessa forma, são evidenciadas as interdependências entre os processos do ambiente organizacional. Apesar da importância dos processos de negócio, nem sempre esses processos fazem parte da base de conhecimento da organização, ficando disponíveis apenas nas ferramentas de modelagem de processos de negócios (GUIDO; PANDURINO; PAIANO, 2016).

No entanto, os modelos de processos de negócio podem apoiar de fato a organização em várias direções, inclusive no desenvolvimento de softwares que ajudem a automatizar os processos. Alguns trabalhos na literatura já propõem a utilização dos modelos de processos de negócio como fonte de informação para os profissionais de Tecnologia da Informação (TI), como, por exemplo: Dias et al. (2006), Cruz, Machado e Santos (2014), Nogueira e Oliveira (2017b), entre outros.

Segundo Oliveira et al. (2013), quando uma organização descreve seus processos de negócio por meio de modelos, é possível utilizar esses modelos para se entender melhor o contexto do domínio de negócio. Isso contribui para que o desenvolvedor de software entenda melhor a linguagem utilizada e as necessidades do negócio. Assim, os requisitos do software tendem a refletir as necessidades reais do negócio, podendo ser especificados de forma mais completa e com poucas redundâncias.

Os modelos de processos de negócio podem ser elaborados de diferentes maneiras, utilizando diferentes linguagens e notações gráficas, incluindo textos descritivos que servem de documentação dos modelos. A notação que tem se destacado nos últimos anos é BPMN (*Business Process Model and Notation*), que é um padrão do Consórcio OMG (*Object Management Group*), cuja primeira versão foi criada em 2005. BPMN é uma notação gráfica

que permite uma ampla descrição visual de processos de negócio.

O conjunto de símbolos gráficos de BPMN é amplo, com mais de 100 símbolos. Contudo, estudos mostram que menos de 20% do vocabulário da notação BPMN são utilizados em mais de 50% dos modelos de processos de negócio, segundo Borger (2012). O autor informa ainda que há cinco elementos mais utilizados na modelagem de processos de negócio: fluxo, atividade, evento, início e fim. Segundo Correia e Abreu (2015), esse percentual baixo de utilização do vocabulário de BPMN está diretamente relacionado à grande quantidade de elementos de sua especificação. Geralmente, a falta de conhecimento da especificação de BPMN pode ocasionar a repetição e a utilização limitada do seu vocabulário.

BPMN pode ser integrada a duas outras notações do Consórcio OMG: DMN (*Decision Model Notation*), que permite melhor descrição das regras de negócio, e CMMN (*Case Management Model and Notation*), que permite a representação e expressão de casos através dos modelos. Observa-se que casos são processos que podem ocorrer de modo *ad-hoc*, com maior controle do processo e decisão negocial na mão de especialistas que executam as atividades.

A especificação de BPMN prevê que uma documentação textual pode ser associada a seus elementos gráficos, para apoiar a modelagem dos processos de negócio. Contudo, essa especificação não define a estrutura dessa documentação, de modo a se ter uma forma padronizada para se documentar textualmente os modelos. Essa documentação pode ser inserida pelas equipes da área de negócios por meio dos sistemas de modelagem de processos de negócio disponíveis. As informações provenientes dessa documentação são relevantes para o processo de levantamento de requisitos do software e, portanto, fundamentais para atingir os objetivos deste trabalho. Além disso, Correia e Abreu (2015) destacam que a especificação de BPMN é um documento relativamente complexo para os modeladores de processos.

A falta de padrão para se documentar os elementos de BPMN na cadeia de processos de negócio reflete-se na forma como os sistemas de modelagem representam a descrição semântica dos elementos de BPMN e como tratam os atributos (estendidos ou não) desses elementos. Além disso, a descrição dos atributos em linguagem natural favorece ambiguidades na interpretação dos modelos de processos de negócio e dificulta a automatização por software (THOMAS; FELLMANN, 2007).

É importante mencionar que os modelos de processos de negócio em BPMN podem ser convertidos para documentos na linguagem XPD (XML *Process Definition Language*), baseada em XML (*eXtended Markup Language*) (VAN DER AALST, 2003). A linguagem XPD é definida pelo Consórcio WfMC (*Workflow Management Coalition*) e possibilita uma estrutura

digital padronizada para representação de processos de negócio. A utilização da linguagem XPD L permite a interoperabilidade entre diferentes ferramentas de modelagem BPMN, devido à utilização de um metamodelo. Isso possibilita a importação e exportação de modelos entre diferentes ferramentas de modelagem, desde que atendam aos níveis de conformidade exigidos pela linguagem XPD L. Contudo, os sistemas de modelagem podem diferir nas versões de BPMN e XPD L. Além disso, modelos importados nem sempre conservam a documentação no formato original.

Considerando que um modelo de processos de negócio documentado pode auxiliar o alinhamento da comunicação entre as áreas de negócio e de TI, ele também pode apoiar a especificação das soluções de software necessárias. Contudo, Pedrinaci et al. (2008) observam que a extração de requisitos a partir de modelos de processos de negócio em um ambiente organizacional pode ser comprometida. O processo de extração pode ser feito manualmente, com alto custo e propensão a erros, ou pode ser automatizado, com dificuldades para se cobrir a grande quantidade de elementos de um modelo de processos de negócio.

Para minimizar esses erros, é essencial o entendimento do domínio de negócio por parte dos profissionais da área de TI. Eles são responsáveis pelas soluções de software e pela automatização dos fluxos de processos da organização. Os requisitos do software devem ser aderentes ao ambiente de negócio, tanto em relação aos processos executados no ambiente como em relação ao vocabulário e à linguagem do domínio do negócio. Muitas vezes o distanciamento entre a visão/linguagem dos profissionais de negócios e os de TI é muito grande. Przybylek (2014) já observava que o processo de Engenharia de Requisitos é dificultado pela falha de comunicação, em função da utilização de muitas notações diferentes entre a área de negócios e a de TI.

Os modelos de processos de negócio em BPMN auxiliam a comunicação entre os profissionais do negócio e de TI em muitos casos. Mas, por outro lado, eles podem ter grandes dimensões e serem, de certo modo complexos, com vários subprocessos, por exemplo. À medida que esses modelos se tornam complexos, a navegabilidade e a legibilidade deles são afetadas. Esses modelos também não são manipuláveis por máquina e não é possível realizar consultas a respeito desses modelos, por meio dos sistemas tradicionais utilizados.

Alguns trabalhos na literatura apontam a direção de se organizar o conhecimento explícito e implícito de um modelo de processos de negócio na forma de uma ontologia, como por exemplo: Ternai, Török e Varga (2016), Guido, Pandurino e Paiano (2016), Haller et al. (2007), entre outros. Contudo, esses trabalhos não exploram os detalhes e particularidades dos elementos da notação BPMN, assim como suas interdependências em um modelo de

processos de negócio. Outros trabalhos, baseados em abordagens de BPM (*Business Process Management*) seguem uma perspectiva ontológica, visando prover processos anotados semanticamente por meio de ontologias, tornando esses processos acessíveis por máquina (GÓMEZ-PÉREZ, 2010). Embora sigam a perspectiva de ontologias, esses trabalhos não automatizam o processo de representação dos modelos de processos de negócio por meio de ontologias.

A motivação para se criar ontologias junto a modelos de processos de negócio está em representar o conhecimento que se encontra implícito nos processos de negócio. As ontologias fornecem uma representação visual e textual dos processos, dados, informações, recursos, colaborações e outras medições, evidenciando interdependências entre os elementos. Segundo Ternai, Török e Varga (2016), as ontologias propiciam representações que expressam o conhecimento a respeito de conceitos, suas propriedades, grupos de conceitos e a relação entre eles.

As ontologias auxiliam na legibilidade, navegação e entendimento das interdependências existentes nos modelos de processos de negócio. Por outro lado, considerando que uma ontologia relacionada a um modelo de processos de negócio pode contribuir para organizar e complementar o conhecimento desse modelo, ela pode servir de apoio para a área de desenvolvimento de soluções de software, além da área de negócios. Isso porque as ontologias são compreensíveis tanto pela equipe de negócios quanto de TI e, nesse caso, constituem uma alternativa complementar para a navegabilidade e a legibilidade dos modelos de processos de negócio. Adicionalmente, as ontologias possibilitam que diferentes consultas sobre o modelo sejam definidas e implementadas, com base em suas classes e relações. Além disso, as ontologias podem representar conhecimento implícito nos modelos, que podem ser derivados de inferências, uma vez que as ontologias são estruturas manipuláveis computacionalmente.

Nesse contexto, os objetivos deste trabalho são apresentados na *Seção 1.1* e a metodologia utilizada para o seu desenvolvimento é apresentada na *Seção 1.2*. A *Seção 1.3*, por sua vez, apresenta a organização dos demais capítulos desta dissertação.

1.1 OBJETIVOS DO TRABALHO

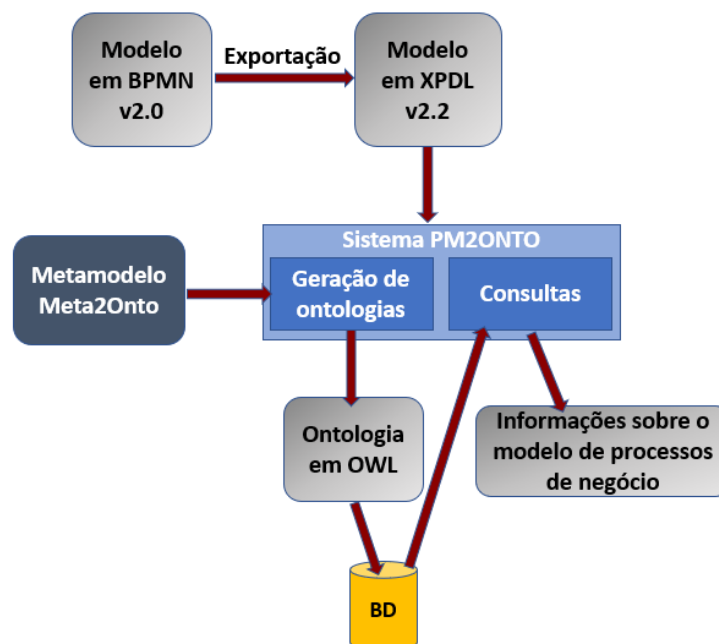
O principal objetivo deste trabalho é apresentar um processo sistemático para a geração automática de uma ontologia a partir de um modelo de processos de negócio na notação BPMN v2.0. A ontologia contempla todos os elementos do modelo de processos de negócio, bem como as relações e interdependências presentes no modelo, com a intenção

de apoiar a sua compreensão e obtenção de informações sobre ele.

O resultado obtido a partir da aplicação do processo sistemático é uma ontologia descrita na linguagem OWL (*Web Ontology Language*), que pode ser manipulada e consultada conforme a necessidade. Assim, foi definido um conjunto de consultas, que podem ser utilizadas para retornar informações sobre o modelo de processo de negócios, usando a ontologia gerada. Para esses propósitos, supõe-se que o modelo de processos negócio seja documentado textualmente e construído segundo boas práticas de modelagem, que são bem conhecidas e exploradas em alguns trabalhos encontrados na literatura, como por exemplo Correia e Abreu (2015).

Na **Figura 1**, é apresentado um esquema que proporciona uma visão geral do processo de geração das ontologias de modo automatizado e do sistema de consultas. Como pode ser visto na figura, o modelo de processos de negócio em BPMN v2.0 deve ser convertido para arquivos na linguagem XPD L v2.2. Isso pode ser realizado por meio de uma ferramenta de modelagem de processos que contemple essas versões da notação e linguagem. Nos exemplos utilizados neste trabalho, foi utilizada a ferramenta *Bizagi Modeler*. Cada arquivo XPD L gerado corresponde à representação de um processo ou um subprocesso. Porém, só é gerado um arquivo XPD L para um subprocesso quando ele é invocado a partir de outro processo, caracterizando assim um subfluxo (WMFC, 2012). É importante mencionar que o processo de conversão para XPD L não ocasiona perdas de informações em relação ao modelo de processos de negócio original na notação BPMN.

Figura 1 - Visão geral do processo de geração de ontologias e do sistema de consultas.



Fonte: elaborado pelo autor.

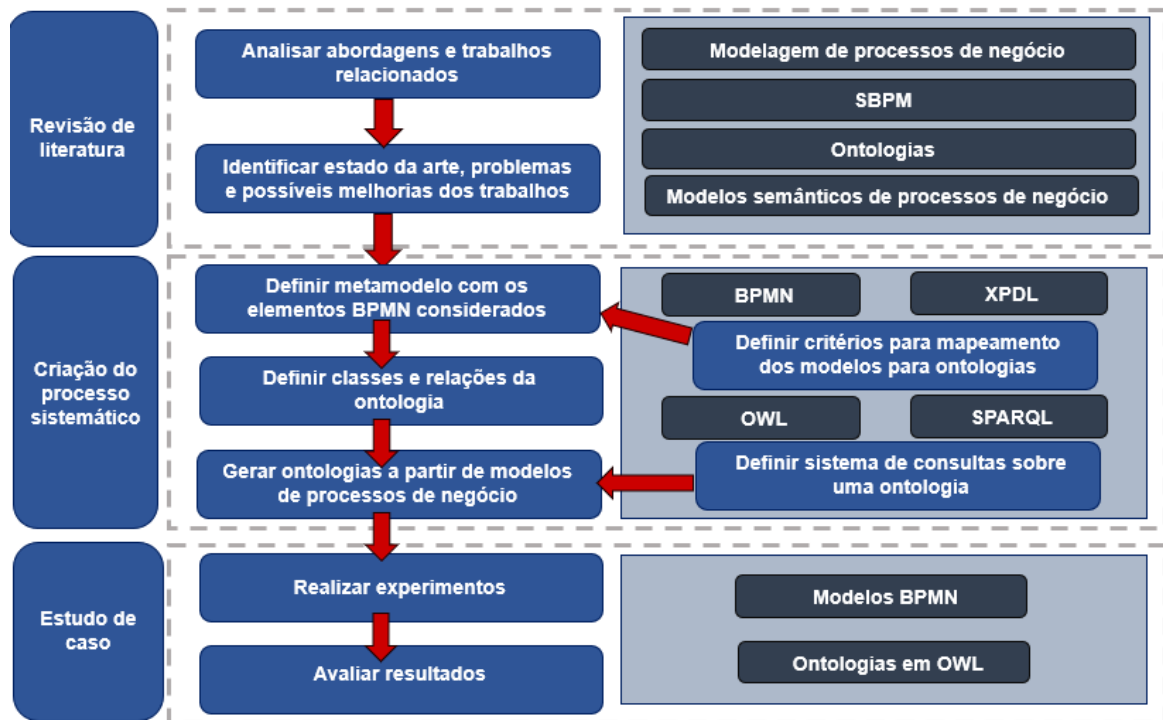
Para apoiar a geração das classes que devem compor a ontologia em OWL, foi definido um processo de mapeamento dos elementos XPDL relevantes para este trabalho para um metamodelo de classes, criado para este trabalho e denominado Meta2Onto. Para automatizar esse processo, foi desenvolvido um sistema denominado PM2ONTO (*Process Model To Ontology*), que também contempla uma interface para consultas sobre as ontologias geradas e armazenadas em um banco de dados. Seleccionada uma ontologia, o sistema de consultas permite o retorno de informações textuais sobre o modelo de processos de negócio correspondente. O sistema PM2ONTO foi desenvolvido usando a linguagem Java para plataforma Web (*World Wide Web*), bem como recursos do *framework* de licença gratuita Apache JENA (ASF, 2017). Observa-se que como as ontologias estão descritas na linguagem OWL, elas podem ser devidamente analisadas e visualizadas através de ferramentas para manipulação de ontologias, como, por exemplo, o sistema *Protégé*.

De modo geral, a solução apresentada objetiva proporcionar um modo automático de geração de uma ontologia, a qual constitui uma representação alternativa de um modelo de processos de negócio, apoiando tanto a equipe de negócios como de TI. Esse tipo de representação é adequado, principalmente, para modelos de média a grande extensão e/ou complexidade, pois melhora a navegabilidade pelos componentes dos modelos. Além disso, melhora a visualização das interdependências inerentes à orquestração dos processos, envolvendo a documentação textual. Como as ontologias podem ser legíveis por máquina, permitem consultas sobre os modelos de modo simples e objetivo, inclusive sobre conhecimentos implícitos, decorrentes de inferências.

1.2 METODOLOGIA DE TRABALHO

Buscando atingir os objetivos apresentados na Seção 1.1, foi desenvolvido um conjunto de atividades, conforme apresentado na **Figura 2**. Inicialmente, foi feita uma revisão bibliográfica sobre modelos de processos de negócio, como estão sendo utilizados atualmente e como se dá a modelagem com a notação BPMN, incluindo ferramentas e boas práticas para essa modelagem. Foram, então, feitos experimentos com alguns sistemas de modelagem com suporte a BPMN v2.0 e XPDL v2.2, como, por exemplo, Bizagi *Modeler*, HEFLO! e Camunda. Os experimentos incluíram comparações no modo de documentação dos modelos que essas ferramentas propiciam, bem como o comportamento na exportação e importação dos arquivos XPDL relativos a modelos de processos de negócio.

Figura 2 - Visão geral das atividades desenvolvidas neste trabalho.



Fonte: elaborado pelo autor.

Diante dos problemas e desafios encontrados para organizar o conhecimento e os inter-relacionamentos existentes nos processos de negócio, foi estudada a abordagem de BPM Semântico, conhecida como SBPM (*Semantic Business Process Management*), que é baseada em ontologias. Os desafios nessa área ainda são grandes, mas pode-se perceber, por meio de levantamentos bibliográficos, que os modelos de processos de negócio podem ser representados por mais de um tipo de ontologia. O estudo da abordagem SBPM permitiu se observar os benefícios das ontologias para representação dos modelos de processos de negócio. Entre esses benefícios, pode-se ressaltar a legibilidade por máquina das ontologias, a possibilidade de se fazer inferências e de se obter informações implícitas, além de melhorar a visualização das interdependências nos modelos de processos de negócio.

Com o objetivo de entender como as ontologias podiam representar os modelos de processos de negócio, foram estudados alguns trabalhos que envolvem a construção de ontologias a partir desses tipos de modelos. Alguns desses trabalhos são apresentados na Seção 3.5 deste trabalho. Foram necessários estudos adicionais sobre a construção de ontologias, incluindo a ferramenta *Protégé* e sua API (*Application Programming Interface*) para OWL. Observou-se que o processo de aprendizagem ontológica pode ser composto de técnicas de Processamento de Linguagem Natural (PLN), mineração de dados e de textos, além de técnicas de aprendizado de máquina, anotação de *corpus* e definição de heurísticas.

O objetivo dos estudos realizados foi analisar os meios para melhor extração de conceitos a partir dos dados textuais da documentação dos modelos de processos de negócio.

As pesquisas e estudos realizados possibilitaram a definição do processo sistemático apresentado neste trabalho, que visa gerar uma ontologia na linguagem OWL a partir de um modelo de processos de negócio na linguagem XPD. A ontologia gerada é formada por classes que representam alguns dos principais elementos de BPMN. Para a definição dessas classes, foram realizados estudos sobre o *framework* PROV, para proveniência de dados na Web (W3C, 2013a). Esse *framework* apresenta, inclusive, uma ontologia, denominada PROV-O, que auxiliou na definição das classes que representam os elementos de BPMN em um modelo de processos de negócio. Essas classes compõem o metamodelo criado para este trabalho.

Para a construção da ferramenta PM2ONTO neste trabalho, foram realizados estudos a respeito do *framework* Apache JENA, que possibilita a criação e manipulação de ontologias em diversas linguagens, inclusive OWL. Para o desenvolvimento de consultas sobre as ontologias geradas, foi estudada a linguagem SPARQL. Através de consultas estruturadas nessa linguagem é possível retornar qualquer informação sobre uma determinada ontologia.

Para avaliar a aplicação do processo sistemático com a ferramenta PM2ONTO e a ontologia gerada, foram realizados estudos de casos com modelos de processos de negócio de diferentes complexidades e áreas de aplicação. O objetivo foi verificar a completude e a representatividade das ontologias em relação aos seus respectivos modelos de processos de negócio.

1.3 ORGANIZAÇÃO DOS CAPÍTULOS

Considerando os objetivos deste trabalho, esta dissertação está organizada em mais cinco capítulos, que buscam apresentar os principais levantamentos e os resultados obtidos durante as atividades realizadas.

O *Capítulo 2* aborda tópicos relacionados aos modelos de processos de negócio, com foco na notação BPMN e na linguagem XPD. O capítulo apresenta uma visão geral sobre a modelagem de processos de negócio. Complementarmente, o APÊNDICE A apresenta todos os elementos de BPMN. Já o APÊNDICE B apresenta uma visão geral das notações DMN e CMMN, incluindo um comparativo com BPMN. Apesar dessas notações não serem consideradas neste trabalho, podem ser consideradas em trabalhos futuros, uma vez que estão despontando ferramentas atualmente que as implementam integradas a BPMN.

Também são apresentadas boas práticas de modelagem, com base em alguns trabalhos encontrados na literatura, para obtenção de modelos mais claros e bem documentados. Esse capítulo também aborda a linguagem XPDL, assim como o processo de mapeamento de elementos BPMN para elementos XPDL.

O *Capítulo 3* apresenta uma visão geral sobre ontologias, bem como metodologias e linguagens para construção de ontologias. Esse capítulo também aborda a linguagem para consultas de ontologias SPARQL. Também são apresentados trabalhos relacionados à construção de ontologias a partir de modelos de processos de negócio, que contribuíram para a definição do processo sistemático deste trabalho.

O *Capítulo 4*, por sua vez, apresenta o processo sistemático para a geração de ontologias a partir de modelos de processos de negócio. Nesse capítulo, são listados os critérios que devem ser considerados para os modelos de processos de negócio que podem ser mapeados para ontologias. Além do processo de mapeamento dos elementos XPDL para os elementos da ontologia gerada, é apresentada a ferramenta PM2ONTO, desenvolvida para automatizar o processo sistemático mencionado e propiciar as consultas em SPARQL. No APÊNDICE C, é apresentada uma visão geral sobre o *framework* Apache JENA, utilizado no desenvolvimento da ferramenta PM2ONTO. No APÊNDICE D, é descrito o metamodelo criado para o mapeamento dos elementos XPDL para os elementos da ontologia gerada no processo sistemático. Por fim, o APÊNDICE E apresenta o código-fonte da versão 1.0 da ferramenta PM2ONTO e instruções para a instalação e utilização da ferramenta.

O *Capítulo 5* apresenta a avaliação do processo sistemático desenvolvido por meio de estudos de caso envolvendo três modelos de processos de negócio reais, com diferentes complexidades. O APÊNDICE F traz os arquivos XPDL referentes aos modelos de processos de negócio utilizados, bem como as ontologias geradas na ferramenta PM2ONTO.

Por fim, o *Capítulo 6* apresenta as considerações finais sobre o trabalho, incluindo comentários sobre a relevância do processo sistemático desenvolvido para as organizações que buscam melhor entendimento e uso dos modelos de processos de negócio.

2

MODELOS DE PROCESSOS DE NEGÓCIO

Os modelos de processos de negócio têm sido cada vez mais utilizados como ferramentas estratégicas por diferentes organizações empresariais, em diferentes áreas de negócio, com a visão de facilitar adaptações a mudanças internas e do mercado. O gerenciamento desses modelos permite conhecer melhor os processos, de modo a se identificar pontos de melhorias para o desempenho dos processos. Esses modelos fazem parte da abordagem de gerenciamento de processos de negócio, conhecida como BPM (*Business Process Management*).

Atualmente, no mercado, há vários sistemas, de domínio público ou proprietários, que apoiam o gerenciamento de processos sob a ótica de BPM. Seguem alguns exemplos de alguns desses sistemas, conhecidos como BPMS (*BPM System*): Bizagi, HEFLO!, Bonita, Camunda, Studio Lecom, Sydle Seed, entre outros. Geralmente, cada um desses sistemas tem um módulo modelador, responsável pela modelagem dos processos de negócio.

Os modelos de processos de negócio podem usar diferentes notações, como, por exemplo: fluxogramas, Diagramas de Fluxos de Dados (DFDs), redes de Petri, diagrama de atividades em UML (*Unified Modeling Language*), etc. Contudo, este trabalho considera a notação gráfica BPMN (*Business Process Model and Notation*), especificada pelo Consórcio OMG (*Object Management Group*).

Na especificação de BPMN é prevista a inserção de informações junto aos elementos de BPMN nos modelos de processos de negócio, para fins de documentação dos modelos. Contudo, como não é especificado quais são as informações e como elas devem estar associadas aos modelos, cada sistema de modelagem utiliza um conjunto de atributos para a documentação. Isso impacta na importação dos modelos por outro sistema de modelagem. Embora a parte visual se mantenha íntegra, nem sempre a documentação se mantém com o mesmo formato de apresentação, muito embora alguns atributos se mostrem comuns a vários sistemas.

Sem a documentação textual, é difícil se compreender a semântica expressa pelos elementos gráficos de BPMN. Contudo, essa documentação não é uma tarefa fácil e depende do conhecimento que se tem sobre os processos, da habilidade da equipe de modelagem, da

estratégia de modelagem e do tempo disponível para a modelagem. Assim, a documentação dos modelos deveria ser adequada à compreensão clara dos processos por todos aqueles que os usarão para execução ou apoio em tomadas de decisão. A falta de uma documentação adequada, com muito atributos incompletos, deixa de expressar a semântica dos elementos do modelo. Isso pode ocasionar falhas na execução dos processos, além de dificultar a compreensão das interdependências existentes entre processos, atividades, atores, artefatos, etc.

A ideia inerente à abordagem BPM é automatizar os processos envolvidos e aproximar a visão da área de negócios e de Tecnologia da Informação (TI). Contudo, visando agregar valor semântico e informacional aos modelos de processos de negócio com perspectivas ontológicas, foi introduzida a abordagem estendida de BPM, conhecida como Gerenciamento de Processos de Negócio Semânticos ou SBPM (*Semantic Business Process Management*). Essa abordagem propõe adicionar semântica ao ciclo de vida tradicional de BPM. Usa, essencialmente, tecnologias semânticas, com destaque para as ontologias, visando enriquecer esses modelos com melhorias na documentação e no entendimento dos mesmos. Essa abordagem possibilita compreensão desses modelos em nível computacional, favorecendo buscas, análises e reaproveitamento do modelo em geral por meio de softwares.

Os modelos semânticos de processos de negócio se diferem dos modelos tradicionais pelo fato de usarem ontologias para representação de fragmentos ou até mesmo do processo de negócio como um todo. Isso lhes proporciona maior valor semântico em comparação com os modelos de processo de negócio tradicionais. Essa associação com ontologias é feita através da anotação semântica definida para a modelagem semântica de processos de negócio. Alguns aspectos dessa abordagem de modelagem semântica serviram de inspiração para a definição do processo sistemático apresentado neste trabalho, pois é utilizada uma perspectiva ontológica para enriquecer o entendimento e a semântica de um determinado modelo de processos de negócio tradicional.

Com base nesse contexto, a *Seção 2.1* apresenta uma visão geral de modelagem de processos de negócio, enquanto a *Seção 2.2* apresenta uma visão específica da notação BPMN. Na *Seção 2.3*, são apresentados alguns aspectos semânticos referentes à BPMN. A *Seção 2.4*, por sua vez, apresenta propostas alinhadas a boas práticas de modelagem de processos de negócio que usam BPMN, de modo a se prezar pela qualidade dos modelos. A *Seção 2.5* apresenta alguns conceitos sobre modelagem semântica de processos de negócio, a qual inspirou a definição do processo sistemático apresentado neste trabalho. A *Seção 2.6* apresenta uma visão geral da linguagem XPD (XML Process Definition Language), utilizada para a representação das definições dos processos de negócio. Na *Seção 2.7*, é apresentada uma síntese do mapeamento de elementos da notação BPMN para elementos da linguagem

XPDL. Por fim, na Seção 2.8 são apresentadas as considerações finais para este capítulo.

2.1 MODELAGEM DE PROCESSOS DE NEGÓCIO

Para um gerenciamento efetivo dos processos de negócio, é importante empregar técnicas adequadas para a modelagem desses processos, de modo que se represente de maneira correta e concisa as particularidades do negócio. Isso contribui para que os integrantes do ambiente organizacional tenham uma visão ampla e clara dos procedimentos internos do local onde trabalham.

A modelagem dos processos de negócio representa o conjunto de atividades dos processos de negócio existentes (modelo AS IS) ou propostos (modelo TO BE) (ABPMP, 2013). O objetivo da modelagem é criar uma representação precisa e completa sobre o funcionamento de um processo (ABPMP, 2013). Para isso, o nível de detalhamento do modelo, assim como o tipo do modelo de processos de negócio, deve variar de acordo com o caso em que será empregado.

Para Tessari (2008), uma metodologia de modelagem de processos de negócio deve ser capaz de representar os diferentes aspectos dos processos sobre as diferentes perspectivas dos atores que os executam. Segundo o autor, a criação de um modelo de processos pode ser instrutiva por si só, revelando problemas, inconsistências e oportunidades para melhorias. Uma vez criado, esse modelo se torna um valioso meio de compartilhar o conhecimento através da empresa. De modo geral, os modelos de processos possuem muitas aplicações nos ambientes de negócios, como: organização (estruturação), descoberta (aprendizagem), previsão (estimativas), medição (quantificação), entre outras (ABPMP, 2013).

Um modelo de processos é formado por ícones gráficos, que representam atividades, eventos, decisões, condições e outros elementos do processo (ABPMP, 2013). Um simples processo já pode envolver diversos atores, que são entidades normalmente representadas por pessoas, organizações ou sistemas que executam tarefas (TESSARI, 2008). Para cumprir uma tarefa, os atores devem seguir uma ordem de cumprimento das subtarefas, que podem ser executadas em paralelo ou sequencialmente. Dessa forma, pode existir uma gama de variações para um fluxo de processos, que poderá variar de acordo com pontos de decisão do modelo.

Os termos “diagrama de processo”, “mapa de processo” e “modelo de processos” são muitas vezes associados como se fossem sinônimos. Entretanto, segundo a Associação de

profissionais de BPM ou ABPMP (*Association of Business Process Management Professionals*), cada um desses termos representa um diferente estágio de desenvolvimento, cada qual agregando um conjunto de informações ao entendimento, análise e desenho dos processos (ABPMP, 2013). Como apresentado na **Tabela 1**, as características dos diagramas e dos mapas de processos de negócio são praticamente as mesmas, mas diferem bastante dos modelos de processos de negócio.

Tabela 1 - Características principais de diagrama, mapa e modelo.

Diagrama e mapa de processo	Modelo de processo
A notação pode apresentar ambiguidades	Convenção padronizada de notação
Baixa precisão	Tão preciso quanto necessário
Menor nível de detalhamento	Maior nível de detalhamento
Ícones que representam os componentes do processo: “inventados” ou com definição vaga	Ícones objetivamente definidos e padronizados
Os relacionamentos dos ícones são retratados de maneira visual	Relacionamentos dos ícones definidos e explicados em anotações, glossário do modelo de processos e narrativas de processo
Retrata apenas um momento específico da realidade	Pode crescer, evoluir e amadurecer
Pode ser criado com ferramentas simples de diagramação	Deve ser criado com a ferramenta de modelagem adequada ao objetivo
Simple de utilizar, mas não permite explorar a informação de forma detalhada	Pode fornecer simulação manual ou automatizada do processo
Difícil de conectar com outros processos existentes	Ligações verticais e horizontais, mostrando relacionamentos entre os processos em diferentes níveis
Apropriado para certas capturas rápidas de ideias	Apropriado para qualquer nível de captura de processos, análises e desenhos

Fonte: ABPMP (2013).

Um diagrama retrata os principais elementos do fluxo de processo e por isso ajuda a identificar rapidamente e entender as principais atividades de um processo; porém, omite detalhes dos fluxos de trabalho (ABPMP, 2013). Assim, um diagrama fornece uma representação inicial do processo.

De forma semelhante, o mapa permite uma visão dos principais componentes do processo, mas com maior precisão do que um diagrama. Dessa forma, agrega mais detalhes sobre o processo e alguns dos relacionamentos mais importantes com outros elementos, como atores, eventos e resultados (ABPMP, 2013).

Por outro lado, um modelo busca representar um determinado estado de negócio (presente ou futuro) e dos recursos envolvidos, tais como pessoas, informações, instalações,

automatizações, finanças e insumos (ABPMP, 2013). Um modelo utiliza padrões de notação que permite um nível mais alto de detalhamento com o uso de ferramentas específicas de modelagem. Por representar um funcionamento mais preciso daquilo que está sendo modelado, um modelo requer mais dados a respeito do processo e dos fatores que afetam seu comportamento. É importante observar que, para este trabalho, somente modelos de processos de negócio são considerados e não mapas e diagramas.

Os modelos de processos de negócio podem ser classificados em quatro tipos, utilizados de acordo com a necessidade da organização (SGANDERLA, 2012):

- **“as-is”**: reflete os processos da organização em seu estado atual. Esse tipo de modelo é utilizado como documentação para conhecimento do pessoal envolvido. Além disso, pode ser usado como entrada para a construção do modelo **“to-be”**;
- **“to-be”**: é uma evolução do modelo **“as-is”**, após a reavaliação das questões de negócios envolvidas, visando alcançar maior eficiência na execução do processo, por meio de melhorias culturais e organizacionais;
- **“to-do”**: baseado no processo **“to-be”**, tem por objetivo propor um olhar do processo a partir da visão da área de sistemas, identificando maneiras de agregar valor ao processo por meio de recursos tecnológicos;
- **“to-run”**: esse tipo de modelo utiliza recursos mais complexos de modelagem, o que permite a execução do processo por meio de ferramentas de modelagem avançadas, possibilitando, assim, a execução automatizada do modelo de processos de negócio.

Para se elaborar a modelagem de processos de negócio pode se contar com diversas linguagens e notações gráficas, como, por exemplo: UML (*Uniform Modeling Language*), fluxogramas, EPC (*Event-Driven Process Chain*), YAWL (*Yet Another Workflow Language*), BPMN (*Business Process Management Notation*), entre outras. A escolha da notação adequada não é algo simples e deve considerar as especificidades da organização. Em determinadas situações é apropriado utilizar diferentes notações para diferentes estágios, níveis ou finalidades de modelagem (ABPMP, 2013). Uma notação adequada oferece várias vantagens, como, por exemplo: conjunto de símbolos, linguagens e técnicas comuns para comunicação entre pessoas, consistência de forma e significado dos modelos resultantes, importação e exportação de modelos de processos entre diversas ferramentas e possibilidade de geração de aplicações a partir de modelos de processos (ABPMP, 2013). Tessari (2008), por sua vez, reforça que é conveniente o uso de uma notação padrão para a modelagem de

processos de negócio em uma organização. Isso facilita o entendimento da intercomunicação dos procedimentos de negócio internos à organização, bem como das colaborações e das transações envolvidas.

2.2 VISÃO GERAL DA NOTAÇÃO BPMN

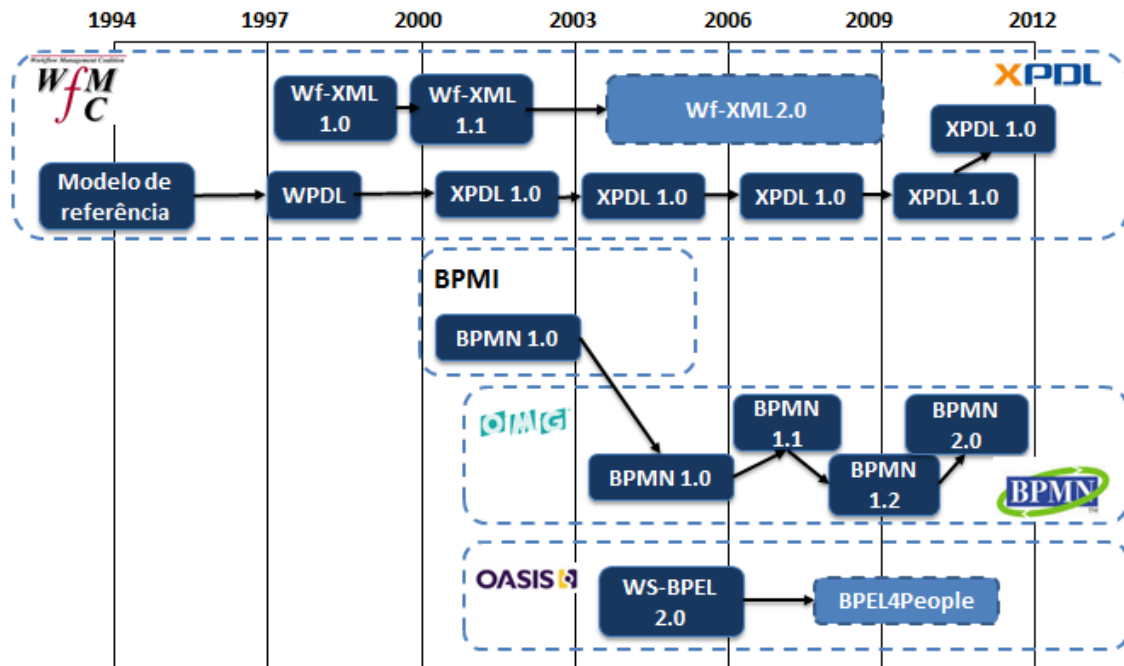
Segundo Chinosi e Trombetta (2012), BPMN tem destaque no universo de modelagem de processos de negócio por ser facilmente compreensível pelos grupos de usuários de negócio que podem usá-la. Esse grupo abrange desde analistas de negócios, que esboçam as versões iniciais dos processos, até os desenvolvedores técnicos, que são os reais encarregados da elaboração da modelagem dos processos. Além disso, inclui o pessoal de negócios que implanta e monitora tais processos (CHINOSI; TROMBETTA, 2012).

Segundo Harmon e Wolf (2011), BPMN é a notação de modelagem de processos de negócio mais usada atualmente entre profissionais que trabalham com BPM. Isso se deve à sua flexibilidade, abrangência e facilidade de uso por esses profissionais, que podem apresentar diferentes níveis técnicos de conhecimento (CORREIA; ABREU, 2015).

A notação BPMN foi publicada originalmente em 2004 pelo Instituto de Modelagem de Processos de Negócio, conhecido internacionalmente como BPMI (*Business Process Modeling Institute*) (TESSARI, 2008, CHINOSI; TROMBETTA, 2012). BPMN foi apresentada como uma notação gráfica, parcialmente inspirada nos diagramas de atividades de UML, para representar o *layout* gráfico dos processos de negócio. BPMN foi desenvolvida para ser de fácil entendimento e utilização, facilitando a modelagem de processos de negócio dos mais simples aos mais complexos. O crescente interesse e adoção por parte das organizações corporativas levou o consórcio OMG a lançar a primeira especificação de BPMN como um padrão em 2006.

A **Figura 3** mostra um esquema com a evolução de padrões de linguagens/notações utilizadas nas aplicações de gerenciamento de processos de negócio, evidenciando os organismos responsáveis pela especificação das mesmas; na figura, BPMN fica evidenciada até a versão 2.0, embora atualmente esteja na versão 2.0.2. Observa-se que a figura traz as linguagens que foram definidas por quatro consórcios/instituições: WfMC (*Workflow Management Coalition*), BPMI (*Business Process Management Initiative*), OMG (*Object Management Group*) e OASIS (*Open Advancing Standards for the Information Society*).

Figura 3 - Evolução das linguagens/notações voltadas a processos de negócio.



Fonte: traduzido de Chinosi e Trombetta (2012).

A especificação de BPMN apresenta um metamodelo formado por 115 metaclasses e 200 metassociações, segundo Correia e Abreu (2015). Correia (2015), por sua vez, observa que muitos padrões já definidos pelo Consórcio OMG apresentam uma especificação formal com regras bem definidas a respeito do metamodelo que esses padrões abordam. Isso não ocorre com a notação BPMN, cujas regras referentes aos elementos dessa notação são especificadas informalmente por meio de linguagem natural. A ausência de restrições bem definidas para o uso dos elementos de BPMN dificulta a criação de funcionalidades para as ferramentas de modelagem, com o objetivo de propor melhorias aos modelos de processos de negócio (ex.: utilização correta dos elementos BPMN) (CORREIA, 2015).

Os autores ressaltam que há três classes de notações para BPMN, sendo que os modelos considerados neste trabalho utilizam elementos da primeira classe apresentada:

- **Orquestração e colaboração de processos**, notação básica definida para a modelagem dos processos;
- **Conversão**, uma versão simples dos diagramas de colaboração;
- **Coreografia**, para modelagem de interações entre os participantes de um processo.

Ressalta-se que BPMN ajuda a se lidar com dois requisitos conflitantes da modelagem

de processos de negócio: a facilidade de se criar os modelos e a possibilidade de se conseguir abranger toda a complexidade inerente aos processos de negócio. Para isso, os elementos gráficos de BPMN são classificados em quatro categorias básicas: objetos de fluxo (*Flow objects*), objetos de conexão (*Connecting objects*), raias (*Swimlanes*) e artefatos (*Flow objects*) (OMG, 2011). Essas categorias são descritas a seguir, mas os elementos gráficos pertencentes a cada uma delas são apresentados no APÊNDICE A.

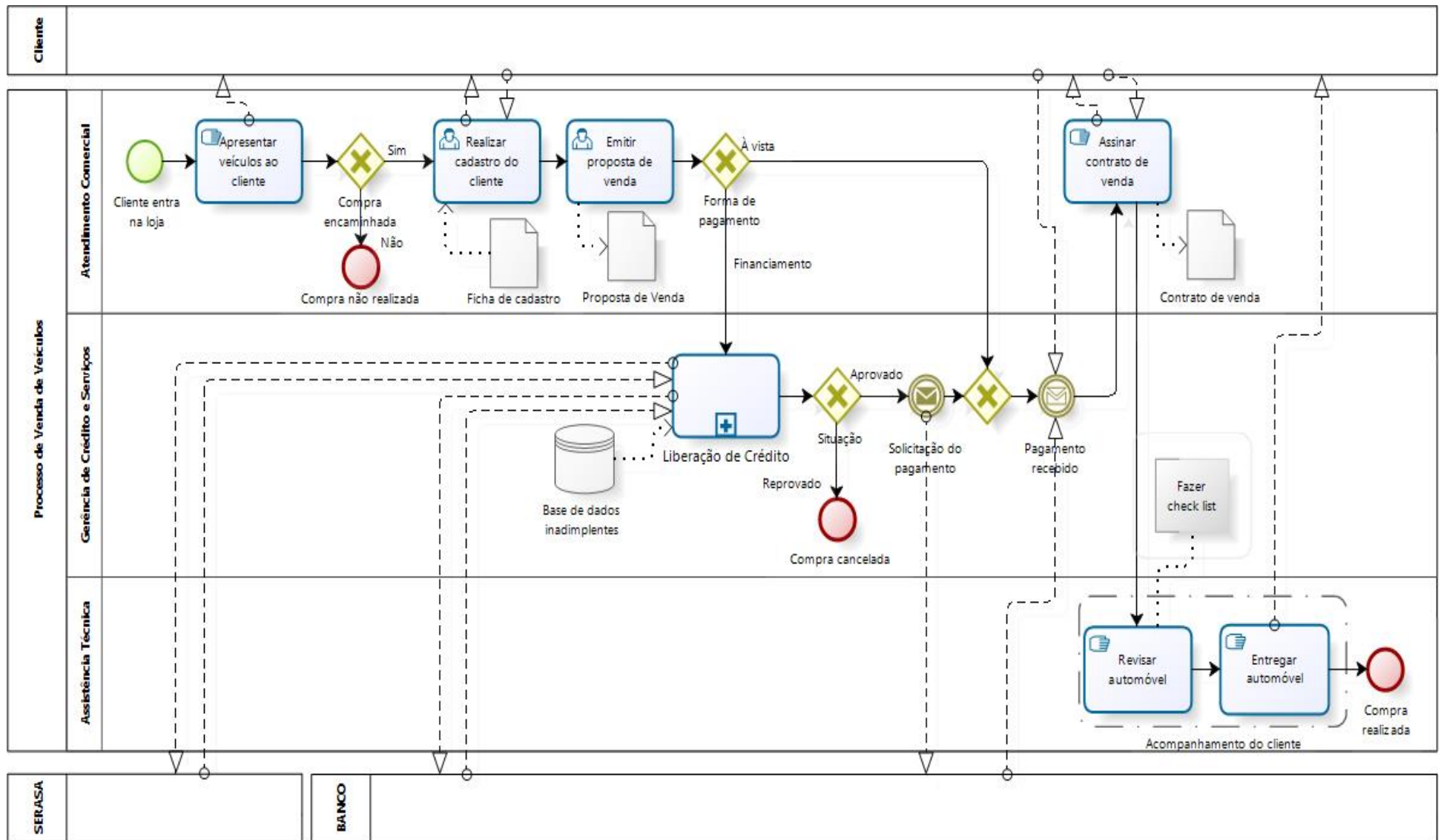
Os “objetos de fluxo” representam todas as ações que podem acontecer dentro de um processo de negócios, determinando seu comportamento. Eles consistem de eventos, atividades e *gateways*. Já os “objetos de conexão” oferecem três maneiras de se conectar objetos uns aos outros: fluxo de sequência, fluxo de mensagem e de associação. As “raias” fornecem a capacidade de agrupamento dos elementos de modelagem primária, contemplando agrupamentos denominados como piscinas e pistas. Os “artefatos”, por sua vez, são usados para fornecer informações adicionais sobre o processo e não afetam o fluxo. São eles: objeto de dados, grupo e anotação (CHINOSI; TROMBETTA, 2012).

A **Figura 4** exibe um exemplo de modelo de processo de negócios em BPMN para venda de veículos. Através da figura é possível observar sete atividades, representadas por retângulos, sendo que uma delas apresenta subprocessos (“Liberação de Crédito”). Pode-se observar também outros elementos, como: objetos de dados (“Base de dados inadimplentes”), *gateways* (“Situação”), e eventos de início (“Cliente entra na loja”) e de fim (“Compra realizada”).

Apesar de BPMN ser uma notação voltada a processos, pode se integrar com a notação DMN, voltada a decisões nas regras de negócio, e CMMN, voltada a casos de uso. DMN é utilizada para descrever especificamente regras de negócio. A notação CMMN, por sua vez, é utilizada para representar e expressar casos de usos através dos modelos de processos de negócio. Esses casos são processos que podem ocorrer de modo *ad-hoc*, controlando melhor os processos e a decisão negocial por parte dos especialistas. Observa-se que, como DMN e CMMN não são consideradas nos modelos de negócio neste trabalho, podendo ser em trabalhos futuros, o APÊNDICE B traz uma visão geral dessas duas linguagens comparativamente à BPMN.

A notação básica de BPMN é composta por mais de 100 elementos gráficos (CORREIA; ABREU, 2015). Esses elementos permitem uma rica modelagem visual dos processos de negócio, conforme mostrado no APÊNDICE A. Correia e Abreu (2015) mencionam que a notação BPMN. Ainda, que a combinação desses elementos, sem a devida validação, realizada pela ferramenta de modelagem, pode vir a gerar um modelo de processos de negócio inconsistente ou inválido.

Figura 4 - Exemplo de modelo de processo na notação BPMN.



Fonte: Dias (2014).

A grande quantidade de elementos, a semântica sofisticada e a existência de muitas escolhas representacionais trazem complexidade ao uso da notação BPMN (LEOPOLD; MENDLING; GÜNTHER, 2016). As 538 páginas da especificação de BPMN v2.0 também colaboram com essa complexidade de compreensão de uso de BPMN de maneira abrangente. Por isso é que Correia e Abreu (2015) mencionam que a especificação de BPMN é relativamente complexa para profissionais que trabalham com modelagem de processos. Geralmente, um conjunto de poucos elementos de BPMN são empregados nos modelos de processos de negócio (NOGUEIRA; OLIVEIRA, 2017a). Esse conjunto de elementos se refere ao que é mais comumente encontrado nos ambientes organizacionais e, de certa forma, mais compreensível pelas equipes envolvidas.

Por outro lado, a especificação de BPMN também não define um modo para documentação dos elementos de BPMN, embora referende essa prática. Por causa disso, especialistas da área de modelagem de processos recomendam a aplicação de boas práticas de modelagem e documentação textual coerente para cada elemento de um modelo de processos de negócio.

É importante ressaltar que a documentação textual dos modelos de processos de negócio utiliza linguagem natural, o que pode causar interpretações equivocadas ou informações incompletas. Certamente isso pode acarretar em dificuldades de compreensão e análises dos modelos por partes dos profissionais interessados (BORGER, 2012). Embora possa estar de acordo com especificação de BPMN, um modelo criado com essa notação pode apresentar descrições ambíguas e falta de especificação relevante, do ponto de vista semântico (NOGUEIRA; OLIVEIRA, 2017a).

2.3 SEMÂNTICA NA NOTAÇÃO BPMN

Linguagens de modelagem empresarial ou EML (*Enterprise Modelling Language*), a exemplo de BPMN, são definidas normalmente como linguagens de modelagem conceitual semiformal: sintaxe definida de maneira precisa, mas a semântica é definida de maneira informal (WAND; WEBER, 2002). De fato, de acordo com o Consórcio OMG (2016), a especificação da notação BPMN é direcionada fortemente para a sintaxe e o detalhamento da parte semântica limita-se a declarações informais.

Em termos sintáticos, BPMN é composta por um metamodelo que apresenta precisão, uma extensão de interface e um número notável de extensões consideráveis (BRAUN; ESSWEIN, 2016). Já as referências semânticas são destinadas apenas a alguns elementos da notação BPMN e as declarações sintáticas encontram-se subespecificadas, uma vez que

BPMN é projetada como uma linguagem de modelagem de processo independente de domínio. Vários objetos de fluxo são textualmente especificados em relação à sua execução e aos padrões de fluxo de trabalho correspondentes. Conceitos relacionados aos materiais semânticos, como por exemplo, tarefas manuais e objetos de dados, não apresentam um detalhamento específico semântico.

Di Marino et al. (2015) apontam que um fator limitante de quase todas as extensões existentes atualmente para a notação BPMN está relacionado à completa ausência de informação semântica associada. Isso pode tornar dificultosa a compreensão e a reutilização de processos de negócio provenientes de diferentes recursos, mesmo quando esses processos fazem parte de uma mesma organização. Consequentemente, existe a necessidade de se anotar semanticamente os processos de negócio com informação significativa, para que haja uma visão mais clara dos conceitos envolvidos em cada uma das tarefas, papéis e regras envolvidas nos processos de negócio.

Segundo Braun e Esswein (2016), alguns trabalhos da literatura apontam a necessidade de EMLs mais flexíveis e adaptáveis, de modo que essas linguagens possam se adequar melhor a situações ou domínios específicos. Nessa direção, os autores apresentam uma pesquisa voltada à análise da semântica existente no metamodelo das extensões de linguagens de modelagem empresarial (EMLs), especialmente BPMN.

A pesquisa realizada por Braun e Esswein (2016) compreendeu uma análise extensiva de trinta e seis extensões de BPMN. Segundo os autores, essa pesquisa revelou falhas notáveis em relação à semântica, tanto no projeto (aspecto processual) quanto na especificação (aspecto estático) das extensões. A análise realizada confirmou a suposição de que a sintaxe é enfatizada demais. As definições das EMLs devem consistir não apenas de sintaxe, mas sim de semântica e pragmática. Apenas 50% das extensões de BPMN analisadas na pesquisa apresentam uma descrição clara dos conceitos de domínio necessários; apenas uma minoria faz algum tipo de comparação ontológica com os elementos originais da notação BPMN. As semânticas das extensões dessa notação são provenientes em boa parte de adições sintáticas; entretanto, em algumas extensões, observaram-se semânticas adicionais sem mudanças sintáticas. De maneira positiva, constatou-se que as redefinições semânticas são muito raras. No entanto, 80% das extensões analisadas não fornecem qualquer especificação semântica para os elementos presentes nas extensões.

Di Marino et al. (2015) apresentam três abordagens válidas para o enriquecimento semântico dos conceitos existentes para a notação BPMN: (1) fornecimento de uma extensão para BPMN; (2) abordagem semântica completa; (3) anotação externa.

A abordagem “fornecimento de uma extensão para BPMN” é voltada ao

enriquecimento da especificação de BPMN v2.0 por meio de informação semântica. Isso é feito modificando-se o arquivo BPMN, definido na linguagem XML, adicionando-se marcações (*tags*) extras ao arquivo BPMN. Assim, cada elemento BPMN pode ser associado a novas propriedades e atributos específicos. Embora possa parecer válida, essa abordagem é muito invasiva, segundo Di Marino et al. (2015). A partir do momento em que novas marcações são adicionadas, o arquivo resultante das modificações propostas por essa abordagem perde o seu padrão. A nova anotação BPMN gerada pode não ser compatível com as ferramentas existentes que implementam a notação BPMN padrão.

A abordagem “semântica completa” é baseada na criação de uma ontologia completa de processos que contenham e representem todos os conceitos e elementos da notação BPMN. Cada elemento do processo seria individualmente instanciado na ontologia, de modo a permitir análises estruturais mais fáceis do processo - desde que as conexões entre os elementos tenham uma representação correspondente. Assim, inferências sobre a composição estrutural de um processo e a avaliação de sua precisão seriam possíveis. Por outro lado, ainda não seria possível extrair informações a respeito do contexto do processo. Além disso, a base de conhecimento resultante se apresentaria bastante carregada, o que dificultaria operações de consulta nessa base (DI MARINO et al., 2015).

Ao contrário das abordagens anteriores, que focam mais na representação estrutural do processo através da notação BPMN, a abordagem de “anotação externa” foca no contexto do processo. Analisando a representação BPMN do processo, a ideia é instanciar indivíduos na ontologia, composta por poucas classes simples, definindo, assim, uma ontologia resumida para o processo analisado. Propriedades e atributos de cada indivíduo são fornecidos por ontologias externas, específicas ao contexto.

Com base nas abordagens apresentadas, Di Marino et al. (2015) optam por descartar a primeira abordagem (“fornecimento de uma extensão para BPMN”), por ocasionar perda da padronização de BPMN. Este trabalho também considera apenas a notação BPMN padrão, sem levar em conta elementos específicos de extensões existentes. Dessa forma, a ontologia gerada pelo processo sistemático definido neste trabalho é formada por classes que representam os elementos definidos na especificação BPMN v2.0, assim como suas relações.

Di Marino et al. (2015) unem as abordagens de “semântica completa” e de “anotação externa”, definindo, assim, uma abordagem híbrida. Embora essas duas abordagens apresentem prós e contras, a combinação dos pontos positivos de cada uma delas possibilita adquirir as características necessárias para se atingir os objetivos esperados pelos autores. Essa combinação permite alavancar tanto as informações estruturais, extraídas da própria notação BPMN, como as ontologias de domínio, para enriquecer a informação existente no

modelo de processos.

É importante ressaltar que este trabalho se baseia nos princípios da abordagem “semântica completa”. Isso porque a ontologia de processos gerada é composta por conceitos que representam os elementos da notação BPMN. Neste trabalho, os modelos de processos de negócio são então mapeados para uma ontologia que permite melhor visualização e análise dos processos de negócio. Porém, ao contrário da abordagem “semântica completa”, são geradas classes ao invés de instâncias para os elementos do modelo de processos de negócio. Assim, não existe uma ontologia base, mas sim um metamodelo definido para auxiliar o mapeamento de um modelo de processos de negócio para uma ontologia, como será apresentado na *Subseção 4.4.1*.

2.4 BOAS PRÁTICAS DE MODELAGEM EM BPMN

Nesta seção, são apresentadas propostas que visam melhorar a qualidade dos modelos de processos de negócio, com base em boas práticas de modelagem em BPMN. Foram selecionadas três propostas diferentes, apresentadas por Correia e Abreu (2015), Leopold, Mendling e Günther (2016) e Nogueira e Oliveira (2017a).

Correia e Abreu (2015) especificam e implementam “regras” por meio da linguagem declarativa OCL (*Object Constraint Language*), utilizada normalmente para complementar diagramas em UML. Os autores aproveitaram OCL para melhorar a semântica de BPMN, propondo uma forma de complementar o metamodelo de BPMN com as regras definidas por eles, em OCL. No total foram definidas 755 expressões em OCL, dentre as quais Correia e Abreu (2015) destacam quatro, que também devem ser consideradas nos modelos de processos de negócio neste trabalho:

- O evento de início de um processo não deve receber nenhum fluxo de sequência, pois esse evento define onde um processo específico será iniciado;
- Fluxos paralelos de sequência iniciados por um *gateway* paralelo devem ser unidos também por um *gateway* paralelo;
- Eventos de início implícitos em um processo necessitam de eventos de fim implícitos;
- Eventos de início do tipo não interrompíveis são apenas permitidos em subprocessos do tipo evento.

No geral, as regras podem ser classificadas em três tipos: (1) Regras de controle de fluxo, relacionadas à interação entre elementos da modelagem; (2) Regras de fluxo de dados, relacionadas ao compartilhamento de objetos ou depósitos de dados por atividades; (3) Recomendações de boas práticas, que são regras opcionais, para recomendar certos elementos de BPMN nos diagramas.

Para validação das regras, Correia e Abreu (2015) conduziram um estudo empírico, usando uma quantidade considerável de modelos na notação BPMN para determinar a conformidade desses modelos com a especificação BPMN. Observa-se que esses modelos estavam disponíveis em repositórios públicos. Ao mesmo tempo, avaliaram a efetividade das regras para verificação da consistência desses modelos. O resultado dos estudos permitiu que os autores observassem um percentual relevante de modelos de processos de negócio que violam as regras da especificação BPMN. Correia e Abreu (2015) também observaram que um percentual ainda maior dos modelos viola as regras de boas práticas por eles definidas.

Em direção similar, Leopold, Mendling e Günther (2016) analisaram 585 modelos de processos de negócio na notação BPMN v2.0. Os autores evidenciaram alguns problemas de qualidade em modelos de processos de negócio, compreendendo representações de divisão (*split*) e união (*join*), fluxos de mensagens e à decomposição do modelo e rotulagem de elementos (*labeling*). As inconsistências encontradas foram agrupadas segundo aspectos estruturais, de *layout* e de rotulagem, conforme apresentado nas **Tabelas 2, 3 e 4**, respectivamente.

Tabela 2 - Inconsistências estruturais nos modelos de processos de negócio.

Inconsistência	Explicação
Conexão inconsistente entre um subprocesso e o processo principal	É obrigatório que as piscinas do processo principal apresentem as mesmas funções que as piscinas do subprocesso.
Uso de fluxos de mensagem em nós incorretos	Fluxos de mensagens devem ser aplicados apenas em elementos corretos semanticamente.
Múltiplas junções	Múltiplas junções acarretam em execuções múltiplas do fluxo subsequente.
<i>Deadlocks</i>	<i>Deadlocks</i> impedem a continuidade de execução de um processo. Eles surgem com a combinação errada dos <i>gateways</i> .
Sintaxe incorreta	Elementos devem ser ligados de acordo com as regras de sintaxe da notação BPMN

Fonte: traduzido de Leopold, Mendling e Günther (2016).

Tabela 3 - Inconsistências de layout nos modelos de processos de negócio.

Inconsistência	Explicação
Tamanho excessivo do diagrama	O diagrama deve caber em uma página A3 (297 x 420 mm).
Bordas e nós sobrepostos	Bordas e nós não devem se sobrepor.
Inconsistência de comportamentos de entrada e saída	Arcos de controle e de mensagem devem ser consistentes, não podendo ser anexados ao fundo das atividades, por exemplo.
Direção de modelagem incorreta	A orientação para a modelagem deve ser da esquerda para a direita.
Espaçamento inadequado	A distância entre os elementos conectados deve corresponder a pelo menos 50% do tamanho do elemento.

Fonte: traduzido de Leopold, Mendling e Günther (2016).

Tabela 4 - Inconsistências de rotulagem nos modelos de processos de negócio.

Inconsistência	Explicação
Objetos de dados sem ligação para um glossário	A descrição de um objeto de dados não está ligada a um glossário.
Funções sem ligação para um glossário	A descrição das funções não está ligada a um glossário.
<i>Gateways</i> com rotulagem incompatível	Ocorre uma violação da convenção da pergunta objeto-particípio.
Atividades com rotulagem incompatível	Ocorre uma violação da convenção verbo-objeto.
Eventos com rotulagem incompatível	Ocorre uma violação da convenção objeto-particípio.

Fonte: traduzido de Leopold, Mendling e Günther (2016).

Com base nas inconsistências levantadas, Leopold, Mendling e Günther (2016) apresentam cinco recomendações para melhorar a modelagem de processos de negócio utilizando a notação BPMN:

- Primeira recomendação: evitar junções e separações implícitas. O uso inadequado de junções e separações é a principal causa dos *deadlocks* e múltiplas junções. Como a notação BPMN oferece várias opções para representação da semântica, então surge esse tipo de problema;
- Segunda recomendação: usar ferramentas de suporte para a decomposição de modelos de processos de negócio. À medida que se tornam extensos, esses modelos podem se tornar difíceis de serem compreendidos e apresentarem inconsistências. Essas ferramentas de decomposição poderiam notificar os modeladores quando seus modelos se tornassem extensos ou até mesmo sugerir a decomposição desses modelos;

- Terceira recomendação: omitir eventos do tipo lançamento de mensagens. Os estudos de Leopold, Mendling e Günther (2016) sugerem que muitos modeladores costumam ligar esse tipo de evento a outros elementos de maneira incorreta, causando confusão entre os modeladores. Assim, os autores sugerem a utilização de atividades, ao invés desse tipo de evento;
- Quarta recomendação: criar um glossário centralizado para manutenção de conceitos. A reutilização de conceitos centrais, como papéis de trabalho e objetos de dados torna-se necessária para a arquitetura de um processo. Os autores recomendam o uso de alguma ferramenta para isso, visto que nem todos os usuários modeladores podem atentar para manter esse glossário atualizado e consistente.
- Quinta recomendação: usar ferramentas de apoio para verificações linguísticas durante a modelagem de processos. Os autores afirmam que o uso estrutural consistente de linguagem natural é algo difícil. Os autores recomendam que a modelagem de processos poderia se utilizar desde o início de técnicas de refatoração para rotulagens incompatíveis dos elementos de um modelo de processos de negócio. Dessa forma, problemas com o uso de linguagem natural poderiam ser reduzidos desde o início da modelagem de processos.

Em outra direção, Nogueira e Oliveira (2017a) definem um conjunto de heurísticas, denominadas heurísticas de negócio, com o objetivo de ajudar as equipes de modelagem de processos na melhoria da qualidade textual e da expressividade gráfica dos modelos de processos de negócio. A avaliação da aplicação dessas heurísticas pode ser feita por meio de um sistema elaborado pelos autores, denominado SRPD (*Software Requirements from Process Definitions*), que realiza a extração automática de requisitos de software. É importante ressaltar que esse sistema recebe como entrada um arquivo XPD v2.2 exportado previamente a partir de um modelo de processo de negócio em BPMN v2.0.

Antes da aplicação das heurísticas de negócio, os autores recomendam utilizar critérios bem definidos por eles para classificar o nível da documentação textual dos elementos na notação BPMN. Os autores definiram uma escala de 1 a 4 para os elementos BPMN, conforme apresentado na **Tabela 5**. O nível 4 indica que o elemento apresenta documentação textual e expressividade gráfica adequadas para o contexto. Se o elemento não estiver nesse nível, então Nogueira e Oliveira (2017a) recomendam a aplicação das heurísticas de negócio. Observa-se que cada heurística de negócio define um princípio que deve ser satisfeito no modelo de processos de negócio. Além disso, essas heurísticas foram

definidas com base nas técnicas de questionamento 5W1H (*Who, What, Why, When, Where e How*).

Tabela 5 - Níveis de classificação dos elementos BPMN com base em sua documentação.

Nível	Descrição
1	Sem documentação: não há informação textual definida para o elemento.
2	Rotulado: apenas o nome do elemento está presente no diagrama. Nenhuma outra informação foi definida para esse elemento ou a informação existente não é sintaticamente válida.
3	Documentado sintaticamente: além do nome, o elemento apresenta informação textual válida sintaticamente, de acordo com a utilização da sentença “termo-fato”.
4	Completo: o elemento apresenta a documentação do nível 3 com informações adicionais que respondem as seis questões definidas nas técnicas 5W1H.

Fonte: traduzido de Nogueira e Oliveira (2017a).

As heurísticas de negócio foram classificadas em quatro grupos, com a representação BH (*Business Heuristic*): heurísticas de negócio para documentar eventos (BHe), atividades (BHa), fluxos de decisão (BHg) e artefatos e repositórios de dados (BHd). Como exemplo, a **Tabela 6** apresenta as heurísticas de negócio para documentação das atividades.

Tabela 6 - Heurísticas de negócio para documentar as atividades.

Questão	Descrição
<i>Who</i>	BHa1 - Cada atividade deve apresentar informação sobre quem a executa. BHa2 - O tipo da atividade deve ser: “Serviço”, “Usuário” ou “Script”.
<i>What</i>	BHa3 - O nome da atividade deve ser único no modelo BHa4 - A descrição da atividade deve usar verbos no infinitivo para formar sentenças simples e concisas. Ex: “Verificar permissão do usuário”.
<i>Why</i>	BHa5 - A informação textual a respeito do objetivo da atividade deve constar na documentação dela. BHa6 - Se existe uma regra de negócio envolvida, então um atributo identificado por “RULE” deve ser definido.
<i>When</i>	BHa7 - Se a atividade for agendada, então um evento agendado deve ser incluído também. Restrições de tempo devem ser registradas na documentação da atividade. BHa8 - Se existe uma regra de negócio envolvida, então um atributo identificado por “RULE” deve ser definido.
<i>Where</i>	BHa9 - A atividade deve ser colocada sobre a piscina/raia representando o departamento/setor da organização que executa essa atividade.
<i>How</i>	BHa10 - A forma como a atividade é executada deve ser descrita na documentação dela. BHa11 - Se existe uma regra de negócio envolvida, então um atributo identificado por “RULE” deve ser definido.

Fonte: traduzido de Nogueira e Oliveira (2017a).

Nogueira e Oliveira (2017a) compararam alguns modelos de processos de negócio originais com as versões ajustadas após a aplicação das heurísticas de negócio, usando o sistema SRPD. Isso porque a quantidade e o nível de detalhamento dos requisitos extraídos dependem do nível de completude e precisão da documentação do modelo de processos de negócio. Os autores mencionam que observaram melhorias significativas no conteúdo dos modelos após os ajustes feitos com as heurísticas de negócio.

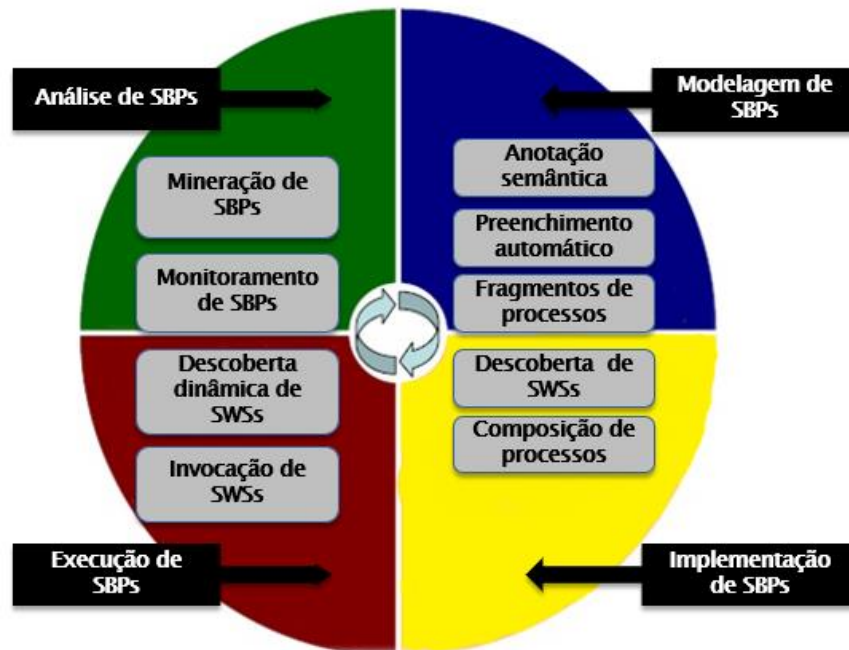
2.5 MODELAGEM SEMÂNTICA DE PROCESSOS DE NEGÓCIO

A abordagem SBPM (*Semantic Business Process Management*) busca combinar BPM com tecnologias semânticas, em particular ontologias e *Web Services* semânticos ou SWS (*Semantic Web Services*). A intenção é fornecer recursos mais convenientes para os usuários de negócios e engenheiros de TI, bem como melhorar a automação no ciclo de vida de BPM. SBPM requer fundamentos de ontologia, de modo que a semântica de cada elemento em um modelo de processos de negócio seja definida de maneira “compreensível por máquina”, propiciando suporte a todo o ciclo de gerenciamento de processos de negócio (LAUTENBACHER; BAUER; SEITZ, 2008).

De acordo com Wetzstein et al. (2007), o uso de tecnologias semânticas não afeta as principais fases do ciclo de vida de BPM, mas aumenta o grau de automação dentro das fases e adiciona melhorias às funcionalidades de um sistema de gerenciamento de processos de negócio ou BPMS (*Business Process Management System*). De fato, todo o ciclo de vida do processo de negócio em BPM pode ser apoiado por recursos semânticos, contemplando as fases de modelagem, implementação (automação dos processos, desenvolvimento e integração de sistemas), execução (implantação de mudanças) e monitoramento (análise e identificação de possíveis melhorias) dos processos de negócio (WETZSTEIN et al., 2007).

O ciclo de vida da abordagem SBPM contempla as quatro fases seguintes: (1) modelagem do processo semântico de negócio ou SBP (*Semantic Business Process*); (2) implementação do SBP; (3) execução do SBP; (4) análise do SBP. Essas fases estão ilustradas na **Figura 5**, incluindo as principais funcionalidades relacionadas. Observa-se que a primeira fase, de modelagem dos SBPs, é a que está sendo considerada neste trabalho, uma vez que visa a construção de uma ontologia para a representação de um modelo de processos de negócio. Assim, cada elemento do modelo é representado por um conceito da ontologia.

Figura 5 - Ciclo de vida de SBPM.



Fonte: adaptado de Wetzstein et al. (2007).

Na fase de modelagem, são produzidos modelos de processos de negócio anotados semanticamente (modelos de SBPs), que são definidos como um conjunto de atividades de descrição de diversos aspectos referentes aos processos de negócio, com base no uso de ontologias (LAUTENBACHER; BAUER; SEITZ, 2008, WETZSTEIN et al., 2007, THOMAS; FELLMANN, 2007). Assim, um modelo semântico de processos de negócio descreve um conjunto de atividades juntamente com os respectivos aspectos funcionais, comportamentais, organizacionais, operacionais e não funcionais dos processos de negócio. Tais aspectos são legíveis e compreensíveis por máquina, já que são anotados semanticamente, de forma que seja possível a inferência de fatos por computadores, através da utilização de uma ontologia básica (LAUTENBACHER; BAUER; SEITZ, 2008).

O principal objetivo da anotação semântica é especificar de maneira explícita as semânticas das tarefas e decisões presentes em um fluxo de processos. As especificações das tarefas não são mais feitas em linguagem natural como na modelagem de processos de negócio usual em BPM, e sim, referenciadas por ontologias (WETZSTEIN et al. 2007). A anotação semântica enriquece os elementos destinados à descrição de um processo de negócios, como, por exemplo, as atividades e eventos. Essas anotações são extraídas de ontologias já existentes ou que foram construídas especificamente para o domínio do processo (FRANCESCOMARINO et al., 2014).

A anotação semântica favorece a utilização de fragmentos de processos, que são

partes identificadas como reutilizáveis de um processo. O objetivo é evitar a duplicação de trabalho por um analista de processos de negócio no momento em que o mesmo se depara com um modelo de processo complexo. Esses fragmentos ficam armazenados em um repositório e são selecionados por meio de uma descoberta baseada em semântica. Essa descoberta só é possível após a especificação da funcionalidade do fragmento por parte do analista (WETZSTEIN et al. 2007).

Os modelos de processos de negócio semânticos permitem a definição de buscas avançadas nos modelos de processos através dos termos e seus relacionamentos anotados semanticamente. Outro benefício desses modelos é a melhoria tanto na validação dos modelos de processos quanto na utilização dos fragmentos de processos como elementos reutilizáveis para composição de um modelo de processo, a fim de evitar o retrabalho na modelagem de processos similares aos existentes (LAUTENBACHER; BAUER; SEITZ, 2008).

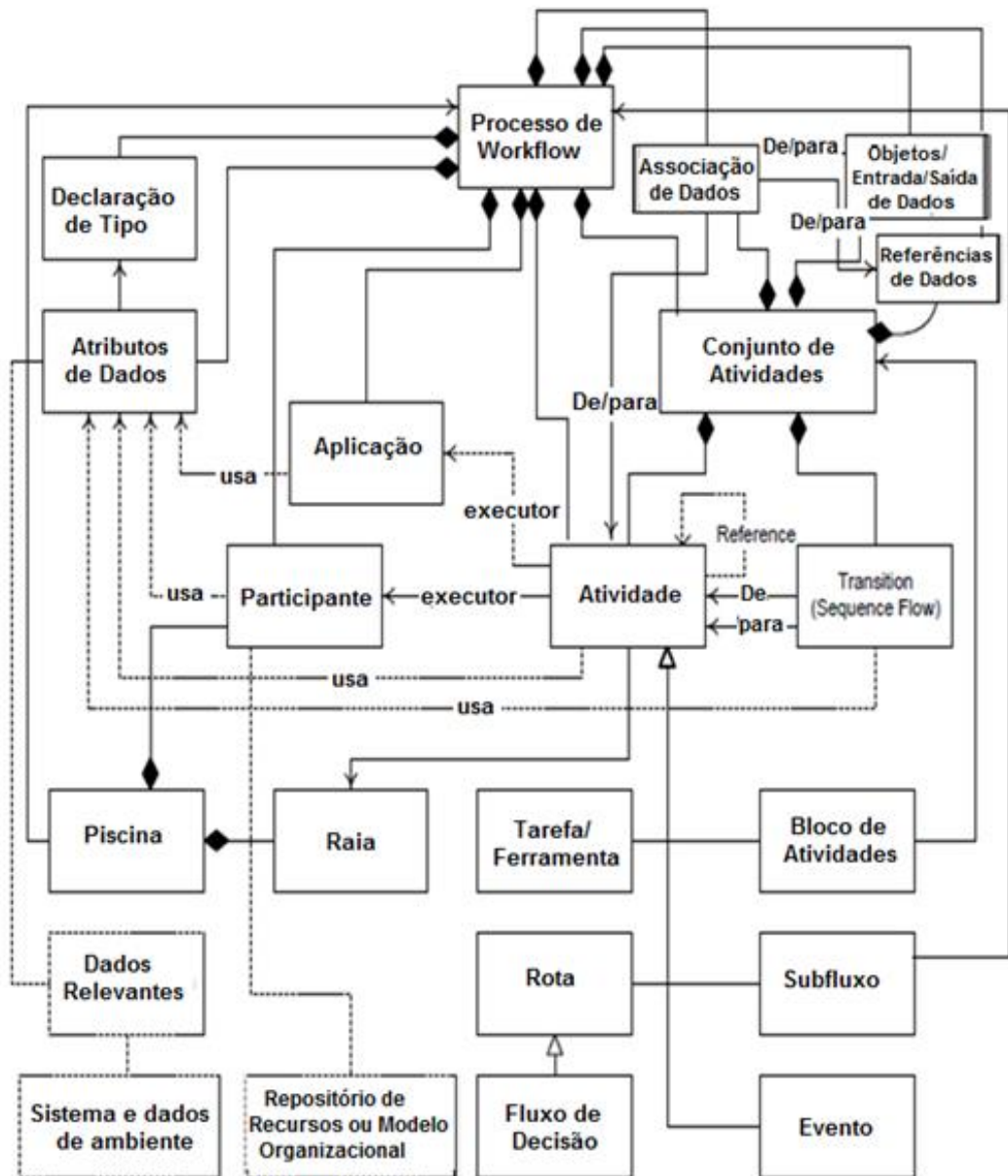
2.6 PROCESSOS DE NEGÓCIO E A LINGUAGEM XPDL

A linguagem descritiva XPDL foi proposta pelo Consórcio WfMC (*Workflow Management Coalition*), para troca de definições de processos entre diferentes sistemas de *Workflow*. A definição de um processo compreende: as atividades interligadas no processo e seus relacionamentos, critérios para definição do início e fim do processo e as informações específicas sobre cada atividade, como os atores participantes, por exemplo (VAN DER AALST, 2003). Segundo o Consórcio WfMC, um processo de negócio deve ser definido de forma que se torne possível a sua manipulação automática por algum sistema de *Workflow*.

Dessa forma, um processo em XPDL é formado por uma ou mais atividades, sendo que cada atividade representa uma unidade lógica de trabalho (HALLER; MARMOLOWSKI; GAALOUL, 2007). As atividades podem estar relacionadas a roteamentos, fluxos de decisão e eventos. Atividades envolvidas com roteamentos e fluxos de decisões são utilizadas para a implementação de decisões. Essas decisões afetam o caminho dos fluxos de sequência durante o processo. As atividades relacionadas a eventos, por sua vez, interferem no início das atividades, onde os caminhos devem ser seguidos durante a execução de um processo de negócio (WMFC, 2012).

Como diagramas de colaboração, que são formados por múltiplos processos que trocam mensagens entre si, um pacote XPDL contém múltiplas trocas de mensagens entre processos. Esses pacotes apresentam o comportamento de um contêiner, que agrupa definições de processos, além de entidades de informação associadas (WFMC, 2012). Na **Figura 6** é ilustrado o metamodelo do pacote XPDL e seus elementos.

Figura 6 - Metamodelo do pacote XPDL.



Fonte: traduzido de WfMC (2012).

A sintaxe e a estrutura da especificação XPDL se baseia no padrão XMI (*XML Metadata Interchange*), usado para troca de informações sobre definições de processos. Dessa forma, as ferramentas de software podem definir representações internas das definições de processos, visando a importação e exportação de modelos de processos de negócio (WfMC, 2012). Observa-se que, em XPDL, a estrutura e os tipos dos documentos são especificados por um esquema XSD (*XML Schema Definition*).

Segundo Van Der Aalst (2003), os principais elementos da linguagem XPDL são:

- **Pacote (Package):** representa o contêiner que abrange os demais elementos de

um documento XPDL;

- **Aplicação (*Application*):** especifica as aplicações e ferramentas invocadas pelos processos de *Workflow* definidos em um pacote;
- **Processo de *Workflow* (*WorkflowProcess*):** composto por atividades e transições, define processos de *Workflow* ou parte deles;
- **Atividade (*Activity*):** é o elemento básico da construção de uma definição de processo de *Workflow*. São definidos três tipos de atividades: (1) Rota (*Route*), que são atividades simples para representação de fluxos; (2) Atividades de bloqueio (*Block Activity*), que são utilizadas para executar um conjunto de atividades de pequena dimensão; (3) Implementação (*Implementation*), que são atividades implementadas por meio de procedimentos manuais. Essa implementação pode ser realizada por uma ou mais aplicações ou por outro processo de *Workflow*;
- **Transição (*Transition*):** representa o relacionamento entre atividades;
- **Participante (*Participant*):** especifica os participantes em um *Workflow*, tais como as entidades que podem realizar determinado trabalho.

Na **Figura 7**, encontra-se um exemplo de um modelo de processos de negócio em XPDL. O processo de *Workflow* é identificado como “Sequencia” e é composto por duas atividades denominadas “A” e “B”, interligadas por meio de um elemento de transição (*Transition*), definido como “AB”.

Figura 7 - Exemplo de um modelo de processos de negócio em XPDL.

```

1<WorkflowProcess Id="Sequencia">
2  <ProcessHeader DurationUnit="Y"/>
3  <Activities>
4    <Activity Id="A">
5      ...
6    </Activity>
7    <Activity Id="B">
8      ...
9    </Activity>
10 </Activities>
11 <Transitions>
12   <Transition Id="AB" From="A" To="B"/>
13 </Transitions>
14</WorkflowProcess>

```

Fonte: traduzido de Van Der Aalst (2003).

2.7 MAPEAMENTO DE ELEMENTOS BPMN PARA ELEMENTOS XPDL

Para a definição do processo sistemático apresentado no *Capítulo 5*, inicialmente, foi necessário analisar as relações entre os elementos BPMN e os seus elementos correspondentes da linguagem XPDL. Alguns trabalhos, como Jung et al. (2004) e Filograna et al. (2007), mencionam as semelhanças e diferenças entre os elementos de BPMN e XPDL.

Filograna et al. (2017) observam que os elementos principais da notação BPMN apresentam um elemento correspondente da linguagem XPDL. Os autores observam também que alguns elementos gráficos da notação BPMN não podem ser convertidos em elementos da linguagem XPDL. O mesmo ocorre para alguns atributos e elementos da linguagem XPDL, que são utilizados para a execução de processos e não apresentam um elemento gráfico de BPMN correspondente a eles.

Essas lacunas na correspondência de elementos de BPMN e XPDL é, de certa forma, natural. Isso porque, como mencionado por Jung et al. (2004), BPMN e a linguagem XPDL apresentam finalidades diferentes. A notação BPMN é empregada no mapeamento e representação visual dos processos de negócio, enquanto que a linguagem XPDL está voltada para a execução desses processos. Assim, em relação ao mapeamento entre os elementos de BPMN e XPDL, Jung et al. (2004) fazem duas observações:

- 1) Elementos da notação BPMN que não estão voltados à execução de processos (somente à modelagem e gerenciamento de processos de negócios) não precisam ser mapeados necessariamente para XPDL, com a intenção de se manter uma representação correta da cadeia de processos. É o caso dos elementos “raias”, exemplificados nessa situação pelos autores. Esses elementos caracterizam os participantes envolvidos na realização das atividades do processo;
- 2) Elementos de XPDL destinados à representação da execução de processos não são necessariamente mapeados a partir de elementos BPMN de maneira direta.

Na **Tabela 7**, pode ser observado um modo de mapeamento entre os elementos de BPMN e XPDL segundo Filograna et al. (2007). Esse mapeamento serviu como base para a definição do metamodelo de elementos BPMN utilizado no processo sistemático apresentado neste trabalho. Todos os elementos apresentados na tabela, com exceção das raia (*lanes*), foram mapeados para o metamodelo. As relações entre elementos, como transições e associações, também foram consideradas para o mapeamento. As raia não foram utilizadas, pois não apresentam a informação que vincula uma atividade a um participante. Essa informação está disponível na *tag Performer*, existente para os elementos XPDL que

representam as atividades.

Tabela 7 - Correspondência entre elementos de BPMN e XPDŁ.

Elemento BPMN	Elemento XPDŁ (<i>tag</i>)
Diagrama de processo de negócio	Pacote
Processo	Processo de <i>Workflow</i>
Atividade	Atividade
Subprocesso	Subfluxo
<i>Gateway</i>	Rota
Evento	Evento
Fluxo de sequência	Transição
Fluxo de mensagem	Fluxo de mensagem
Associação	Associação
Piscina (<i>pool</i>)	Piscina (<i>pool</i>)
Raia (<i>lane</i>)	Raia (<i>lane</i>)
Objeto de dados	Artefato do tipo “Objeto de dados”
Grupo	Artefato do tipo “Grupo”
Anotação	Artefato do tipo “Anotação”

Fonte: traduzido de Filograna et al. (2007).

2.8 CONSIDERAÇÕES FINAIS

Neste capítulo, foi apresentada, inicialmente, uma visão geral da modelagem de processos de negócio (Seção 2.1), incluindo a diferença de modelos, diagramas e mapas de processos de negócio. Na Seção 2.2 foi apresentada uma visão geral de BPMN, enquanto na Seção 2.3 foram apresentadas discussões sobre aspectos semânticos dessa notação. Já a Seção 2.4 apresentou três propostas diferentes de se implementar boas práticas de modelagem em BPMN, ressaltando as partes que foram consideradas neste trabalho. A Seção 2.5 mostrou uma síntese sobre a abordagem SBPM, que inclui a utilização de ontologias para anotações semânticas nos elementos BPMN dos modelos de processos de negócio. Na Seção 2.6 foram apresentadas considerações sobre a linguagem XPDŁ, importantes para este trabalho. Por fim, a Seção 2.7 tratou do mapeamento entre elementos da notação BPMN e da linguagem XPDŁ.

Diante desse contexto, podem ser feitas algumas considerações. A utilização de ferramentas tecnológicas e de notações para representação e descrição das cadeias de processos é algo indispensável para empresas que estão preocupadas em gerenciar seus

processos de negócio. As ontologias associadas aos modelos de negócio podem ser vistas como um recurso para facilitar a análise e o gerenciamento dos processos de negócio, bem como enriquecer a base de conhecimento da organização.

Observou-se que as ferramentas de modelagem de BPMN nem sempre têm suporte à mesma versão da notação. Apesar da versão 2.0.2 já ter sido especificada em 2014, ainda não são muitos os modeladores que implementam nem a versão 2.0, especificada em 2011. Alguns modeladores com suporte à BPMN v2.0 apresentam a opção de exportação dos modelos em BPMN v2.0 para XPD L v2.2. Isso é importante para este trabalho, uma vez que a estrutura baseada em XML contém definições de processos e favorece a interoperabilidade entre sistemas e, conseqüentemente, não restringe o uso de ferramentas de modelagem.

O mapeamento entre elementos da notação BPM e da linguagem XPD L apresentado na *Seção 2.7* contribuiu para compor o metamodelo de elementos de BPMN para geração de ontologias, apresentado no *Capítulo 4*. Além disso, os estudos apresentados evidenciaram o quanto ainda estão desalinhadas as visões da área de negócio e de TI. A descrição de elementos de BPMN nos modelos de processos de negócio por meio de linguagem natural ajuda muito ambas as áreas, mas podem causar ambiguidades na interpretação.

Assim, a abordagem SBPM evidenciou os benefícios em se utilizar ontologias para representar os modelos de processos de negócio. A modelagem semântica desses processos inspirou a definição do processo sistemático apresentado no *Capítulo 4*. Embora não conte com procedimentos para anotação semântica, esse processo sistemático se baseia na definição de uma ontologia como forma de melhor representar os modelos de processos de negócio.

As considerações sobre a complexidade de modelagem com BPMN mostraram que se pode gerar inconsistências nos modelos de processos de negócio, comprometendo a ampla compreensão desses modelos. Dessa forma, é importante que sejam consideradas boas práticas ao se modelar processos de negócio, como as mencionadas na *Seção 2.4*. Pesquisas nessa direção permitiram observar os benefícios da aplicação de boas práticas à modelagem, colaborando para melhor compreensão do modelo. Dessa forma, é recomendado que os modelos de processos de negócio para os quais serão geradas as ontologias sejam modelados seguindo esses tipos de recomendações.

3

ONTOLOGIAS A PARTIR DE MODELOS DE PROCESSOS DE NEGÓCIO

As ontologias, de modo geral, podem representar qualquer contexto e podem ser amplamente utilizadas em diversas áreas. Fornecem uma representação do conhecimento, mesmo quando ele se encontra implícito e quando necessita de melhor representação visual. As ontologias fornecem uma representação visual e textual dos processos, dados, informação, recursos, colaborações, entre outros recursos. Elas também fornecem meios para descrever, de maneira livre, diferentes aspectos de um domínio de negócio (TERNAI; TÖRÖK; VARGA, 2016). As ontologias proporcionam entendimento compartilhado de um domínio ou tarefa, além de facilitar a comunicação entre as diversas áreas presentes em um ambiente organizacional (FALBO, 2016). Dessa forma, elas podem representar a conceitualização expressa nos modelos de processos de negócio, bem como extrair, explicitar e estruturar o conhecimento embutido nesses processos, muitas vezes de modo implícito. Essa estruturação torna possível a compreensão desse conhecimento por máquina (GÁBOR; KÖ, 2016). A intenção é prover meios para auxiliar os processos de negócio com técnicas inteligentes e facilitar a automação do gerenciamento dos processos de negócio.

Considerando a aproximação das equipes de negócio e de Tecnologia da Informação (TI) de interesse à melhoria e automação de processos, cabe mencionar que as ontologias podem ser aplicadas amplamente em diversas frentes da Engenharia de Software. Ontologias podem ser integradas a infraestruturas de desenvolvimento de software, além de apoiar as atividades que compõem todo o processo de desenvolvimento. Isso é ratificado por Happel e Seedorf (2006), que mencionam que a aplicação de ontologias pode trazer contribuições para todas as etapas presentes no ciclo de desenvolvimento do software. Elas podem auxiliar na descrição de funcionalidades dos componentes de um software, tanto na etapa de análise quanto de projeto. No contexto da Engenharia de Requisitos, alguns trabalhos na literatura, como Sateli et al. (2012) e Falbo, Braga e Machado (2014), argumentam a possibilidade de se introduzir semântica nos documentos de ERS. A intenção é que esses documentos possam ser utilizados por ferramentas automatizadas, como as de Wiki semântico (combinação da tecnologia de Wiki com as de Web Semântica) (SCHAFFERT et al., 2006).

Diante das inúmeras aplicações e contribuições das ontologias, a *Seção 3.1* apresenta

uma visão geral sobre ontologias, incluindo uma breve descrição de alguns tipos de ontologias. A Seção 3.2 apresenta algumas metodologias voltadas à construção de ontologias, que inspiraram, de alguma forma, o processo sistemático apresentado neste trabalho. Na Seção 3.3, são apresentadas algumas linguagens para representação de ontologias, com destaque à linguagem OWL, considerada neste trabalho. A Seção 3.4 apresenta uma visão geral da linguagem para consultas de ontologias SPARQL, também considerada neste trabalho. Na Seção 3.5, são apresentados alguns trabalhos relacionados à geração de ontologias a partir de modelos de processos de negócio. Na Seção 3.6, são destacados os aspectos deste capítulo que contribuíram para o desenvolvimento deste trabalho.

3.1 VISÃO GERAL SOBRE ONTOLOGIAS

De modo geral, uma ontologia define as regras de combinações possíveis entre determinados termos e as relações entre eles; sobre esse conjunto de termos podem ser formuladas consultas. Ontologias têm se tornados populares, em grande parte, pelo fato de promoverem o entendimento comum e compartilhado sobre um domínio, que pode servir de comunicação entre pessoas e sistemas de aplicação (NARDI; FALBO, 2006).

Guizzardi (2000) trata as ontologias como um artefato computacional composto de três partes: (1) um vocabulário de conceitos, definições e as possíveis propriedades desses conceitos; (2) um modelo gráfico, mostrando todas as possíveis relações entre os conceitos; (3) um conjunto de axiomas formais, que restringem a interpretação dos conceitos e relações, representando o domínio de maneira clara e não ambígua. Para o autor, uma ontologia não descreve apenas um conhecimento imediato, que pode ser obtido diretamente a partir da observação do domínio, mas também o conhecimento derivado, que é obtido por meio de inferência sobre o conhecimento imediato disponível.

Guarino (1998) define uma ontologia como um conjunto de conceitos e termos usados para descrever um determinado domínio. Esses conceitos se relacionam por meio de atributos e relacionamentos. Para o autor, as ontologias são uma alternativa para caracterizar e relacionar entidades em um domínio, pois representam o conhecimento nele contido.

Na mesma direção, Almeida e Bax (2003) mencionam que uma ontologia é formada por conceitos e seus relacionamentos, com os seguintes componentes básicos:

- **classes:** representam os conceitos da ontologia dispostos de maneira hierárquica. As superclasses definem as classes superiores dessa disposição, enquanto que

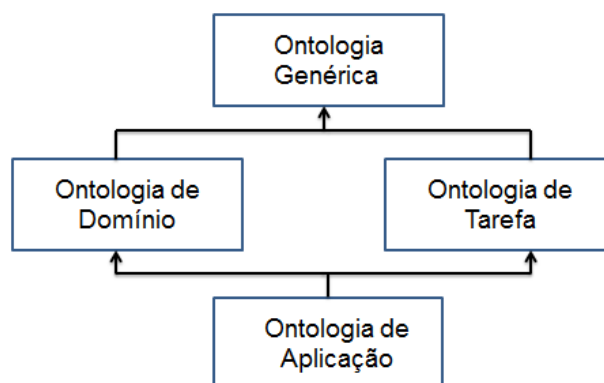
as subclasses definem as classes inferiores;

- **relações:** representam a interação entre os conceitos da ontologia;
- **axiomas:** representam restrições entre os conceitos, com base em suas relações;
- **instâncias:** representam elementos específicos da ontologia.

Segundo Gábor e Kő (2016), o termo “aprendizagem ontológica” trata do processo de criação de uma ontologia, seja de maneira automática ou semiautomática, sem que haja uma grande necessidade de esforço humano. Esse termo é referenciado como um processo destinado à extração de conhecimento conceitual a partir de várias fontes, bem como a construção/criação de uma ontologia a partir do zero, por meio do enriquecimento ou preenchimento de uma ontologia existente.

Guarino (1997) e Guizzardi (2000) classificam as ontologias em cinco categorias: (1) ontologias genéricas; (2) ontologias de domínio; (3) ontologias de tarefa; (4) ontologias de representação; (5) ontologias de aplicação. Esses tipos de ontologias estão relacionados de maneira hierárquica na **Figura 8**, onde as ontologias de domínio e de tarefa são especializações das ontologias genéricas. Já as ontologias de aplicação dependem tanto de um domínio particular quanto de uma tarefa específica e, por isso, fazem parte da junção das ontologias de domínio e de tarefa (GUIZZARDI, 2000).

Figura 8 - Hierarquia dos tipos de ontologia.



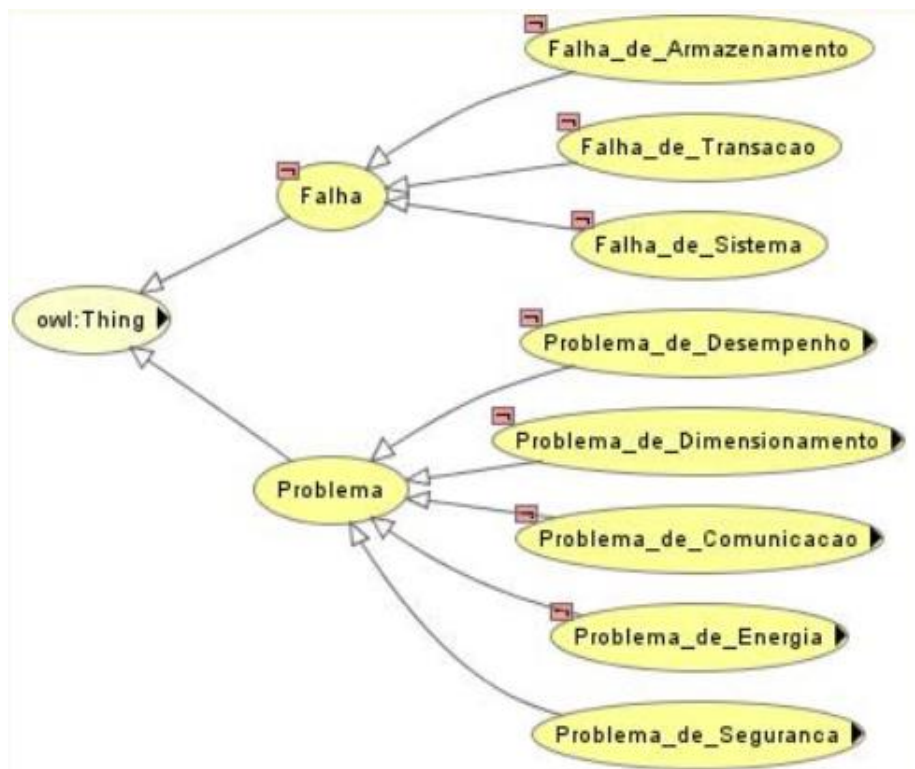
Fonte: traduzido de Guizzardi (2000).

As ontologias genéricas descrevem conceitos gerais, como tempo, espaço, matéria, objeto, evento, ação, entre outros aspectos que são independentes de um problema ou domínio particular.

As ontologias de domínio, por sua vez, expressam conceituações de domínios

particulares e descrevem o vocabulário relacionado a um domínio genérico. Segundo Parreira e Penteado (2015), uma ontologia de domínio explicita o conhecimento de um domínio particular, possibilitando, assim, um entendimento comum dos termos e relacionamentos presentes no domínio em questão. No caso de domínios particulares, tem-se o exemplo da Medicina e do Direito. Já a **Figura 9** mostra uma parte de uma ontologia de domínio sobre tipos de problemas e falhas em Sistemas de Gerenciamento de Banco de Dados (SGBDs).

Figura 9 - Parte de uma ontologia de domínio de tipos de problemas e falhas em SGBDs.



Fonte: Afonso e Pereira (2013).

As ontologias de tarefa descrevem o vocabulário em relação a uma tarefa ou atividade genérica, como por exemplo, vendas ou diagnose. Elas expressam conceituações para contribuir na resolução de problemas, independente do domínio em que ocorram.

Por outro lado, as ontologias de representação explicam as conceituações que fundamentam os formalismos de representação do conhecimento e procuram tornar claros os compromissos ontológicos, embutidos em formalismos de representação de conhecimento.

Por fim, as ontologias de aplicação, descrevem conceitos dependentes do domínio e da tarefa. Dessa forma, ontologias de aplicação representam conceitos que devem ser a especialização de termos das ontologias de domínio e de tarefa correspondentes. Esses conceitos normalmente correspondem a regras aplicadas a entidades de domínio enquanto

executam determinada tarefa.

É importante ressaltar que as ontologias de processo não possuem uma definição precisa na literatura. Alguns trabalhos, como Koschmider e Oberweis (2005), se referem à ontologia de processo como um *framework* de descrição conceitual de processos. Para esses autores, ontologias de processo são abstratas e gerais. Em contraste a isso, Gábor e Szabó (2013) mencionam que as ontologias de tarefa determinam um pequeno subconjunto do espaço do processo e a sequência de atividades em um determinado processo.

3.2 METODOLOGIAS PARA CONSTRUÇÃO DE ONTOLOGIAS

Para a construção de ontologias de modo geral, existem diversas metodologias. Porém, nesta seção são apresentadas informações relevantes que inspiraram a construção das ontologias consideradas no *Capítulo 4*.

Assim, Gábor e Kő (2016) mencionam que há três maneiras de se construir ontologias de processos de negócio: (1) reutilizando ou estendendo conceitos de uma ontologia existente para a composição de uma nova ontologia, usando um método, como, por exemplo, de mapeamento dos conceitos relevantes; (2) utilizando um *framework* existente que apresente recursos para a construção de ontologias; (3) transformando um arquivo em XML, gerado por uma ferramenta de modelagem de BPM, em um arquivo em uma linguagem de representação de ontologias. Essa transformação pode ocorrer com a definição de um método de mapeamento de *tags* existentes no arquivo XML, gerado pela ferramenta de modelagem BPM, por exemplo. É importante ressaltar que, neste trabalho, foi utilizada a terceira maneira para se construir as ontologias a partir de modelos de processos de negócio, convertidos em um ou mais arquivos XPD, por meio de uma ferramenta de modelagem BPMN; um método foi definido para mapear as *tags* de XPD para a construção das ontologias com a linguagem OWL (*Web Ontology Language*).

A metodologia para a construção de uma ontologia de processos denominada *Methontology* utiliza uma plataforma (*framework*) para dar suporte à construção de ontologias, no nível de conhecimento. Ela foi desenvolvida no laboratório de Inteligência Artificial da Universidade de Madri, na Espanha, de acordo com Giles (1979) e Gómez-Pérez (1996). O desenvolvimento da ontologia é baseado em uma sequência de atividades para cada uma das três etapas definidas, como apresentado brevemente na **Tabela 8**.

Cabe também apresentar a metodologia usada no projeto TOVE (*TOronto Virtual Enterprise*), definida por Gruninger e Lee (2012). Essa metodologia leva em conta parte do

domínio de processos de negócio e a modelagem das atividades. Ela é constituída por seis etapas, apresentadas, de forma resumida, na **Tabela 9**.

Tabela 8 - Etapas e atividades da metodologia Methontology.

Etapas	Descrição dos tipos de atividades
1. Atividades de gerenciamento do projeto.	– Planejamento: identificação das tarefas a serem desempenhadas, como essas tarefas devem ser organizadas, quanto tempo e quais recursos elas devem consumir até serem completadas. Essa atividade é essencial quando se pretende fazer reuso de ontologias existentes. – Controle: atividade que garante que as tarefas planejadas na fase anterior sejam executadas completamente. – Garantia de qualidade: atividade que assegura que os produtos resultantes das atividades (ontologia, software, documentação) sejam satisfatórios.
2. Atividades orientadas ao desenvolvimento.	– Especificação: atividades que definem porque a ontologia será construída, que uso será feito dela e quem serão seus usuários finais. – Conceituação: atividades de estruturação do domínio de conhecimento da ontologia, usando modelos de significado no nível do conhecimento. – Formalização: atividades de transformação do modelo conceitual da atividade anterior num modelo formal ou semicomputável. – Implementação: atividades de construção de modelos computáveis numa linguagem computacional. – Manutenção: atividades de atualização e correção da ontologia.
3. Atividades de suporte, desempenhadas em paralelo ao desenvolvimento	– Aquisição de conhecimento: atividades de aquisição de conhecimento sobre um determinado domínio. – Avaliação: atividades de julgamento técnico das ontologias, dos ambientes de software associados e da documentação produzida, usando frames de referência. – Integração: atividades essenciais quando há reuso de ontologias existentes; – Documentação: atividades de detalhamento claro e exaustivo das fases de desenvolvimento.

Fonte: traduzido de Gómez-Pérez (1996).

É possível correlacionar algumas etapas e atividades da metodologia Methontology e da metodologia do projeto TOVE com a etapa para geração de ontologias de modo automático, apresentada neste trabalho, na **Seção 4.3**.

A etapa 2 da metodologia Methontology contempla cinco atividades: especificação, conceituação, formalização, implementação e manutenção. Neste trabalho, a “especificação” foi resultante dos estudos sobre modelos de processos de negócio, limitações das ferramentas de modelagem e dificuldades no entendimento desses modelos em um ambiente organizacional. As atividades de “conceituação” e “formalização” correspondem ao mapeamento dos elementos XPDL de um modelo de processos de negócio para as classes que devem compor a ontologia de processos esperada. As atividades de “implementação” e

Tabela 9 - Etapas da metodologia para desenvolvimento de ontologias no projeto TOVE.

Etapas	Descrição
1. Descrição de cenários motivacionais.	Os cenários motivacionais são descrições de problemas ou exemplos que não são cobertos adequadamente por ontologias existentes. A partir desses cenários-problema se chega a um conjunto de soluções possíveis, que carregam a semântica informal dos objetos e relações que posteriormente serão incluídos na ontologia.
2. Formulação informal das questões de competência.	Com base nos cenários descritos, são elaboradas questões de competência, com a intenção de que seja possível representá-las e respondê-las usando a ontologia a ser desenvolvida.
3. Especificação dos termos da ontologia numa linguagem formal.	Definição de um conjunto de termos/conceitos, a partir das questões de competência. Esses conceitos servirão de base para a especificação formal, usando uma linguagem de representação de conhecimento.
4. Descrição formal das questões de competência.	Descrição das questões de competência, usando uma linguagem formal.
5. Especificação formal dos axiomas.	Criação das regras, descritas em linguagem formal, a fim de definir a semântica dos termos e relacionamentos da ontologia.
6. Verificação da completude da ontologia.	Estabelecimento de condições que caracterizem a ontologia como completa, através das questões de competência formalmente descritas.

Fonte: traduzido de Gruninger e Lee (2012).

“manutenção” correspondem ao algoritmo de geração de ontologias definido, implementado na ferramenta desenvolvida para automatizar o processo sistemático, apresentada na *Seção 4.3*.

As etapas 3, 5 e 6 da metodologia do projeto TOVE também podem ser relacionadas ao processo de geração de ontologias deste trabalho. A “especificação formal dos termos da ontologia” e dos “axiomas” corresponde ao mapeamento das classes do metamodelo Meta2Onto, desenvolvido neste trabalho para as classes e propriedades da ontologia, conforme apresentado na *Subseção 4.4.1*. A “verificação da completude da ontologia”, por sua vez, está relacionada ao processo de desenvolvimento da ferramenta que automatiza o processo sistemático de geração de ontologias. Ao longo do desenvolvimento, foram feitas análises/testes para garantir que a ontologia resultante contemplasse todos os elementos representantes do modelo de processos de negócio considerado, conforme apresentado na *Seção 5.1*.

3.3 LINGUAGENS PARA REPRESENTAÇÃO DE ONTOLOGIAS

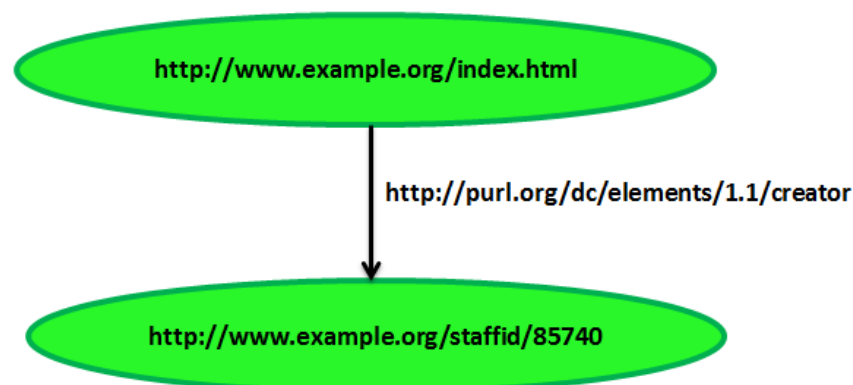
Em relação à representação das ontologias, o Consórcio W3C disponibiliza a especificação de vários recursos: RDF (*Resource Description Framework*), RDFS (*RDF*

Schema), SKOS (*Simple Knowledge Organization System*), OWL e RIF (*Rule Interchange Format*) (CATARINO; SOUZA, 2012). Nesta seção, são apresentadas informações relevantes para este trabalho sobre a linguagem OWL, bem como sobre os recursos RDF e RDFS, utilizados para a definição das consultas sobre as ontologias.

O padrão RDF consiste em um *framework* para representação da informação presente na *Web*. Em RDF, os recursos podem ser descritos por meio de triplas: sujeito, predicado e objeto (RDF WORKING GROUP, 2014). Os recursos possuem identificadores da *Web*, denominados URIs (*Uniform Resource Identifier*), que nomeiam coisas e relacionamentos existentes entre elas na *Web* (MANOLA; MILLER, 2004). Por meio desse modelo simples, torna-se possível a combinação, exposição e compartilhamento de dados (estruturados ou semiestruturados) por diversas aplicações (RDF WORKING GROUP, 2014). Todos os recursos definidos pelo padrão RDF são também descritos em termos de propriedades, ou seja, com atributos para descrever um recurso ou com as características que possibilitam a identificação de um recurso e seus valores correspondentes.

O recurso RDF Schema (RDFS) é uma extensão semântica de RDF, que fornece um vocabulário de modelagem de dados, para a descrição de grupos ou recursos relacionados, bem como para a descrição dos relacionamentos entre esses recursos (W3C, 2014). A **Figura 10** mostra um exemplo de uma declaração simples em RDF, representada por um grafo. Essa declaração apresenta dois conceitos referenciados por meio de suas respectivas URIs. A relação entre os dois é representada pela seta, que também apresenta uma URI. A relação em questão diz respeito ao criador da página que é referenciada pelo primeiro conceito.

Figura 10 - Exemplo de uma declaração simples em RDF, em forma de grafo.



Fonte: Manola e Miller (2004).

Quanto à linguagem OWL, para Hepp et al. (2007) essa é a linguagem de descrição mais importante para a Web Semântica. OWL é uma linguagem descritiva formada por duas camadas: sintática e semântica (ISOTANI; BITTENCOURT, 2015). A estrutura de um documento em OWL é formada pelas seguintes partes (W3C, 2004a):

- uma *tag* inicial: `<rdf:RDF... >`;
- um conjunto de “XML namespace”, detalhando os vocabulários que serão usados dentro da ontologia;
- o cabeçalho da ontologia no qual se coloca o comentário, a versão anterior, o rótulo e importação de outras ontologias;
- a ontologia propriamente dita, com a criação dos conceitos por meio de classes, relações etc., terminando com a *tag* `</rdf:RDF>`.

A linguagem OWL é subdividida em três classes de sintaxe: OWL *Lite*, OWL DL e OWL *Full*. A linguagem OWL DL é processada de maneira eficiente por um motor de inferências de lógica de descrição. Já a linguagem OWL *Lite* é acessível ao processamento de algoritmos de inferência simples. A linguagem OWL *Full*, por sua vez, apresenta todos os recursos de OWL *Lite* e OWL DL (W3C, 2004b).

Os *namespaces* definidos por “rdf:” ou “rdfs:” estão presentes nas linguagens RDF e RDF Schema, respectivamente. O prefixo “owl:” por sua vez, indica a inclusão da notação OWL para os termos introduzidos. A **Figura 11** apresenta parte de uma ontologia em OWL para mostrar que as classes assim como as relações entre elas e suas características. Observa-se que as classes podem ser vistas como conjuntos de indivíduos e representam de maneira concreta um conceito. As classes são definidas através da propriedade “owl:Class”. A relação de hierarquia entre classes, que representa um relacionamento entre indivíduos, é identificada pela propriedade “rdfs:subClassOf”. É possível definir relações para os indivíduos a partir de instâncias de objetos existentes na ontologia. Essas definições são feitas através da utilização da propriedade “owl:ObjectProperty”.

A linguagem OWL apresenta diversos elementos para definição de relações entre as classes que compõem uma ontologia. Alguns deles podem ser listados a seguir:

- *equivalentClass*: define as classes envolvidas como iguais;
- *equivalentObjectProperty*: define as propriedades de objeto envolvidas como iguais;

Figura 11 - Parte de uma ontologia escrita na linguagem OWL.

```
<?xml version="1.0"?>

<rdf:RDF xmlns="http://www.co-ode.org/ontologies/pizza/pizza.owl#"
  xml:base="http://www.co-ode.org/ontologies/pizza/pizza.owl"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema#"
  xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"
  xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  xmlns:owl="http://www.w3.org/2002/07/owl#">
  <owl:Ontology rdf:about="Pizzas">
    <rdfs:comment xml:lang="en">Ontologia de exemplo</rdfs:comment>
  </owl:Ontology>

  <owl:ObjectProperty rdf:about="#apresentaTempero">
    <rdf:type rdf:resource="http://www.w3.org/2002/07/owl#FunctionalProperty"/>
    <rdfs:range rdf:resource="#Tempero"/>
    <rdfs:comment xml:lang="en">Exemplo de propriedade de objeto</rdfs:comment>
  </owl:ObjectProperty>

  <owl:Class rdf:about="#RecheiodeAspargo">
    <rdfs:subClassOf rdf:resource="#RecheioVegetal"/>
    <rdfs:subClassOf>
      <owl:Restriction>
        <owl:onProperty rdf:resource="#apresentaTempero"/>
      </owl:Restriction>
    </rdfs:subClassOf>
  </owl:Class>

  ...
</rdf:RDF>
```

Fonte: elaborado pelo autor.

- *equivalentDataProperty*: define as propriedades de dados envolvidas como iguais;
- *differentFrom*: define as classes envolvidas como diferentes;
- *dataTypeProperty*: define uma propriedade de dados com tipo determinado para as classes;
- *inverseOf*: define uma propriedade como inversa de outra especificada;
- *transitiveProperty*: define uma propriedade como transitiva;
- *symmetricProperty*: define uma propriedade como simétrica (dada uma classe A relacionada com B por uma propriedade X, então a classe B está relacionada com A também pela propriedade X);
- *functionalProperty*: define uma propriedade como funcional;
- *inverseFunctionalProperty*: define uma propriedade inversa funcional;
- *disjointWith*: define uma disjunção entre classes;
- *unionOf*: define uma nova classe como uma união de outras classes;
- *intersectionOf*: define uma nova classe como uma intersecção de outras classes;

- *complementOf*: define uma nova classe por meio da diferença entre outras classes.

A linguagem OWL também apresenta os quantificadores universal e existencial para definir as relações em ontologias, os quais são listados a seguir:

- *someValuesFrom*: quantificador existencial (*some*). Descreve que para todos os indivíduos de uma determinada classe, há pelo menos um indivíduo pertencente a outra classe relacionado por uma determinada propriedade;
- *allValuesFrom*: quantificador universal (*only*). Descreve uma classe com indivíduos onde todos os relacionados devem ser de uma determinada classe.

Também é possível utilizar elementos da linguagem OWL para definir restrições de cardinalidade, como, por exemplo (ISOTANI; BITTENCOURT, 2015):

- *maxCardinality*: define o número máximo de indivíduos pertencentes a uma determinada relação;
- *minCardinality*: define o número mínimo de indivíduos pertencentes a uma determinada relação;
- *exactCardinality*: define o número exato de indivíduos pertencentes a uma determinada relação.

3.4 LINGUAGEM SPARQL PARA CONSULTAS EM ONTOLOGIAS

“SPARQL” é um acrônimo recursivo proveniente do inglês para *SPARQL Protocol and RDF Query Language*. A linguagem SPARQL foi criada pelo grupo RDF DAWG (*Data Access Working Group*) do Consórcio W3C (*World Wide Web Consortium*) em 2008. Atualmente na versão 1.1. Essa linguagem foi utilizada para a definição das consultas associadas ao processo sistemático apresentado no *Capítulo 4*.

A linguagem SPARQL pode ser utilizada para definir consultas sobre diversas fontes de dados, desde que esses dados estejam armazenados em formato RDF (tripas representado sujeito, predicado e objeto) ou sejam visualizados em RDF por meio de um *middleware* (W3C, 2013b). SPARQL apresenta recursos para a realização de consultas a respeito de padrões de grafos, sejam eles requeridos ou opcionais, em conjunto com suas disjunções e conjunções. Por isso SPARQL pode ser vista como uma linguagem destinada

essencialmente a consultas de grafos (ARENAS; PÉREZ, 2011). Uma consulta definida em SPARQL pode retornar desde um grafo em RDF até um conjunto de resultados.

A estrutura de uma consulta SPARQL é formada por duas partes: (1) um corpo, representado por uma expressão relativamente complexa para encontrar grafos em RDF; (2) um cabeçalho, que é formado por uma expressão que indica qual será o resultado da consulta (ARENAS; PÉREZ, 2011). Na **Figura 12** é apresentado um exemplo de consulta em SPARQL, onde pode ser observado o corpo, indicando a expressão destinada à busca do grafo, e o cabeçalho, indicando os campos esperados como retorno.

Figura 12 - Exemplo de uma consulta em SPARQL.

```
PREFIX rdf : <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX owl : <http://www.w3.org/2002/07/owl#>
SELECT distinct ?graph_uri
WHERE
{
    GRAPH ?graph_uri { ?s rdf:type owl:Class } . }
```

Fonte: He Group (2013).

A sintaxe de uma consulta em SPARQL normalmente é formada pela seguinte estrutura (HE GROUP, 2013):

- Prefixos, como por exemplo: *rdf*: *http://xmlns.com/foaf/0.1*;
- Cláusula de resultado da consulta: *SELECT* ...;
- Conjunto de dados envolvido na consulta: *FROM* <...>;
- Padrão de busca nos grafos: *WHERE* {...};
- Modificadores da consulta (opcionais): *GROUP BY*, *HAVING*, *ORDER BY*, *LIMIT*, *OFFSET*, *VALUES*.

A sintaxe de SPARQL considera os seguintes operadores que podem ser utilizados como padrão de busca nos grafos: *OPTIONAL*, *UNION* e *FILTER*. O símbolo “.” pode ser utilizado para realizar a concatenação desses operadores para formar a expressão de busca nos grafos (ARENAS; PÉREZ, 2011). Dessa forma, cada tripla definida dentro da cláusula *WHERE* pode ser concatenada às demais através do símbolo “.”.

A especificação RDF inclui uma série de URIs reservadas, que podem ser utilizadas em uma consulta em SPARQL. Para utilizá-las, basta declarar o prefixo *rdfs*:

<http://www.w3.org/2000/01/rdf-schema#>. Entre as URIs reservadas, pode-se citar: “rdf:type”, “rdfs:range”, “rdfs:domain”, “rdfs:subClassOf” e “rdfs:subPropertyOf”. É importante ressaltar que a especificação RDF considera dois tipos de dados nas consultas: (1) identificadores de recursos, representados pelas URIs; (2) *literals*, expressos por meio de tipos de dados primitivos (número, booleano, cadeia de caracteres, entre outros) (ARENAS; PÉREZ, 2011).

3.5 TRABALHOS RELACIONADOS

Na literatura podem ser encontrados alguns trabalhos que tratam da geração de ontologias a partir de modelos de processos de negócio em BPMN. Entre eles, são aqui destacados cinco trabalhos: Haller et al. (2007), Missikoff et al. (2010), Fanesi, Cacciagrano e Hinkelmann (2015), Ternai, Török e Varga (2016) e Guido, Pandurino e Paiano (2016). Embora alguns apresentem certas semelhanças com este trabalho, nenhum deles apresenta um processo sistemático para a geração automática de ontologias. De modo geral, em relação a este trabalho, os cinco trabalhos apresentados usam versões diferentes de XPDL, tratam de modo diferente o mapeamento dos elementos de BPMN para as classes da ontologia, a quantidade de relações consideradas e as interdependências definidas para a ontologia, assim como usam tipos de ontologias diferentes deste trabalho. No texto seguinte, essas diferenças serão abordadas de modo comparativo.

Convém observar que apenas Haller et al. (2007) e Missikoff et al. (2010) exportam os modelos para a linguagem XPDL, como neste trabalho, mas eles utilizam versões anteriores à XPDL v2.2. Além disso, os elementos de BPMN não são mapeados individualmente para a ontologia, como neste trabalho. Por outro lado, Guido, Pandurino e Paiano (2016) definem apenas relações básicas entre os elementos de BPMN e as interdependências não são exploradas profundamente. Já neste trabalho, são definidas todas as relações que são possíveis de se extrair a partir do contexto onde cada elemento se encontra no modelo.

Em termos de tipo e composição da ontologia gerada, Haller et al. (2007) e Ternai, Török e Varga (2016) propõem a criação de uma ontologia de processos, a exemplo deste trabalho. Já o trabalho de Missikoff et al. (2010) propõe a criação de um *framework* para compor uma base de conhecimento empresarial, representada por uma ontologia de domínio. O trabalho de Fanesi, Cacciagrano e Hinkelmann (2015), por sua vez, define uma ontologia multicamada, que é composta tanto por classes como instâncias, ao contrário deste trabalho, que define somente classes para a ontologia, favorecendo assim a categorização e navegabilidade.

A ontologia de processos proposta por Haller et al. (2007), denominada oXPDL, é

baseada em um arquivo definido na linguagem XPDL v2.0. O objetivo dos autores é aplicar os conceitos de ontologias à linguagem XPDL, de maneira que seja incluído valor semântico aos modelos de processos em XPDL. A intenção é aumentar o nível de interoperabilidade entre processos e fluxos de trabalho de um ambiente organizacional. Para isso, os autores apresentam uma ferramenta que realiza a conversão automática de instâncias de processos para a ontologia oXPDL. A ontologia proposta pelos autores é descrita por meio de cinco aspectos: funcional, de controle, informacional, organizacional e operacional.

O aspecto funcional fornece facilidades na modelagem para definir as hierarquias dos *Workflows* abordados na representação em XPDL. A maior parte das propriedades funcionais presentes no modelo oXPDL são referenciadas por meio de um conceito denominado *PackageHeader*. Um exemplo desse conceito é exibido na **Figura 13**. Esse conceito define a camada hierárquica dos *Workflows*, declarações de tipos, aplicações e artefatos compartilhados entre processos. Além disso, define propriedades relacionadas ao leiaute gráfico, para representar um modelo na notação BPMN ou em qualquer outra notação.

Figura 13 - Tradução de um cabeçalho em XPDL para a ontologia oXPDL.

```
instance PackageType_1 memberOf PackageType
  hasName hasValue _string("sample workflow process")
  hasPackageHeader hasValue PackageHeader_1
  hasTypeDeclaration hasValue Order
  hasParticipant hasValue DBConnection
  hasProcessType hasValue ProcessType1
  hasProcessType hasValue ProcessType1

instance PackageHeader_1 memberOf PackageHeader
  hasXPDLVersion hasValue _string("0.09")
  hasVendor hasValue _string("XYZ, Inc")
  hasCreated hasValue _string("6/18/2002 5:27:17 PM")
```

Fonte: Haller et al. (2007).

O aspecto de controle está relacionado à ordem das tarefas em um modelo de processo e ao roteamento ao longo dessas tarefas. Na linguagem XPDL, assim como na ontologia proposta por Haller et al. (2007), o tipo de processo constitui elementos de modelagem primária, agrupando atividades, dados e recursos relacionados. Na ontologia proposta, todos os tipos de atividades são modelados em conformidade com o que foi definido na linguagem XPDL.

O aspecto informacional trata da produção e do consumo de dados - o que representa o fluxo de dados entre as tarefas. Haller et al. (2007) apresentam um mapeamento direto de todos os tipos de dados para a ontologia em questão.

O aspecto organizacional, por sua vez, trata da definição do responsável pela execução de cada tarefa no *Workflow*, além de fornecer aos modeladores meios para atribuir restrições aos agentes, sejam eles humanos ou máquinas. Na ontologia proposta por Haller et al. (2007), as propriedades organizacionais são modeladas semanticamente.

Por fim, o aspecto operacional envolve a modelagem das aplicações que devem ser gerenciadas e invocadas pelos sistemas de gerenciamento de *Workflows* ou WFMS (*Workflow Management System*). Na ontologia proposta por Haller et al. (2007), esse aspecto foi enriquecido, definindo de maneira melhor quais os papéis desempenhados pelos processos e quais os desempenhados pelos associados.

Em outra direção, Missikoff et al. (2010) propõem a criação de um *framework* baseado na linguagem BPAL (*Business Process Abstract Language*), a fim de agregar semântica aos esquemas de processos de negócio por meio de formalismos e regras. Essa linguagem é utilizada em conjunto com ontologias de domínio, usando OWL para capturar o conhecimento dos processos de negócio e resultando em uma base de conhecimento empresarial. Observa-se que, neste trabalho, a construção da base de conhecimento não utiliza formalismos lógicos, e sim um mapeamento direto dos elementos de BPMN para uma ontologia em OWL.

Fanesi, Cacciagrano e Hinkelmann (2015) buscam agregar maior valor semântico aos modelos de processos de negócio usando a definição de uma ontologia unificada (composta por várias camadas) baseada na linguagem OWL-FA (uma extensão de OWL-DL, que é uma sublinguagem de OWL). A primeira camada da ontologia é formada por metamodelos, que contêm as classes da ontologia. A segunda camada consiste das instâncias dos conceitos que compõem a ontologia na primeira camada. As instâncias dessa segunda camada compõem a terceira camada. A intenção dos autores é permitir consultas na ontologia sobre diversos aspectos de um modelo de processos de negócio. A ontologia definida pelos autores deve classificar cada classe relativa a um elemento de BPMN, assim como sua instância, em uma camada. Essa abordagem classifica os conceitos como classes, instâncias ou até mesmo como classes e instâncias ao mesmo tempo. É importante mencionar que, ao contrário do proposto pelos autores, neste trabalho todos os elementos de um modelo são definidos como classes, favorecendo sua categorização na ontologia, bem como a definição das relações e consultas.

Em relação a consultas sobre as ontologias geradas, isso é tratado apenas por Fanesi, Cacciagrano e Hinkelmann (2015) entre os trabalhos destacados. Fanesi, Cacciagrano e Hinkelmann (2015) propõem algumas consultas como exemplos para explorar as classes da ontologia e seus relacionamentos. Com base nessas consultas, os autores levantam alguns problemas de representação encontrados dos modelos de processos de negócio em

ontologias. Convém observar que tanto as consultas quanto esses levantamentos auxiliaram a definição do processo sistemático apresentado no *Capítulo 4*, bem como as consultas definidas sobre as ontologias (ver *Subseção 4.4.2*).

Em relação ao uso de um metamodelo para a construção da ontologia, dentre os cinco trabalhos destacados nesta seção, somente Guido, Pandurino e Paiano (2016) também definem um metamodelo para a notação BPMN v2.0, composto por classes e propriedades para compor uma ontologia, como neste trabalho. Esse metamodelo engloba os mesmos elementos de BPMN que este trabalho, com as seguintes classes principais: *graphical_element* (subclasses *swimlane*, *flow_object*, *connecting_object*, *artifact* e *data*), *lane*, *pool* e *supporting_element* (subclasses *event_detail* e *participant*). Classes de restrição de disjunção são definidas a partir de instâncias e propriedades de objeto. Contudo, as relações definidas para os elementos de BPMN são limitadas e não são autoexplicativas, o que dificulta o entendimento das interdependências existentes nos modelos de processos de negócio. Além disso, essa forma de tratamento dificulta a definição de consultas sobre as ontologias, que, geralmente, precisam dessas interdependências para respostas significativas. Neste trabalho, esses tipos de problema foram mitigados, pois o metamodelo não é limitado apenas às relações diretas existentes no modelo de processos de negócio; as relações contemplam todos os elementos pertencentes às interdependências.

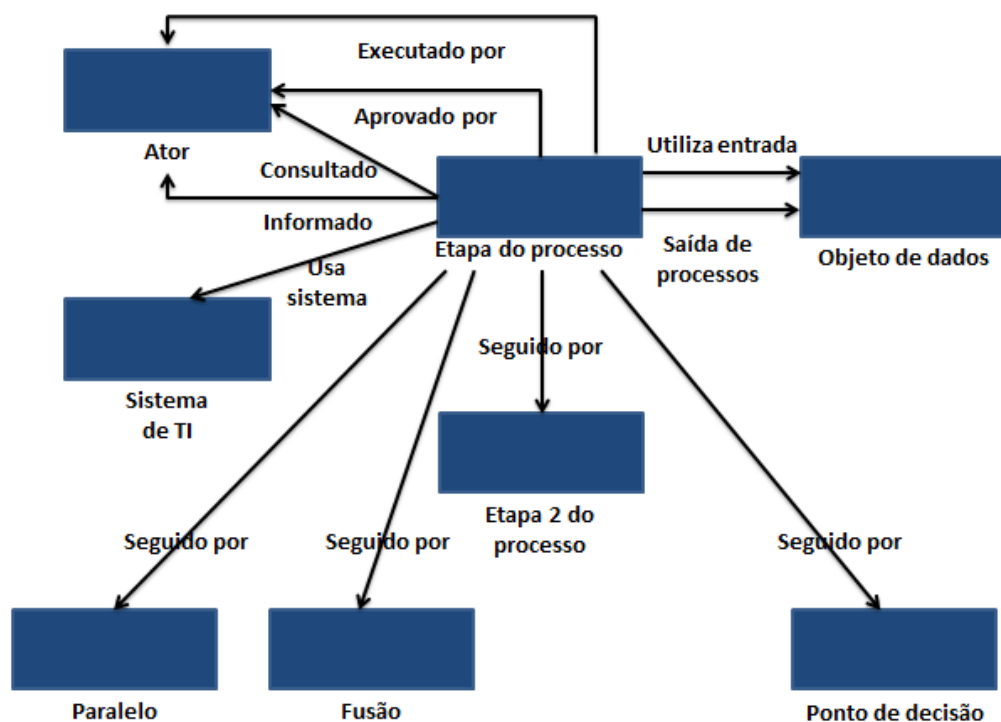
Outro trabalho destacado é o de Ternai, Török e Varga (2016). Os autores propõem a criação de uma ontologia de processo a partir da representação em XML de um modelo de processos de negócio, obtida por meio da plataforma de modelagem BOC ADONIS (Boc Group, 2013). A solução proposta é baseada em uma conexão entre o modelo de processos e uma base corporativa de conhecimento. Assim, a estrutura do processo foi utilizada para a construção da estrutura do conhecimento em uma ontologia. Os autores procuraram investigar uma forma de associar os elementos do modelo de processos de negócio com os conceitos de ontologias.

Assim, na proposta de Ternai, Török e Varga (2016) os modelos de processos de negócio devem ser, primeiramente, exportados para um arquivo XML bem estruturado e padronizado (os autores usaram a plataforma ADONIS para isso). Cada objeto do modelo de processo de negócio é representado por uma *tag* denominada *Instance* na estrutura do arquivo XML. Os atributos, por sua vez, são representados pela *tag Attribute*.

Para os autores, os elementos dos modelos de processos de negócio devem então ser mapeados para os elementos apropriados da ontologia. Para isso, usaram um conversor de modelos conceituais para modelos de ontologias. A regra geral que Ternai, Török e Varga (2016) seguiram foi expressar cada elemento do modelo de processos de negócio utilizado

com a plataforma ADONIS como uma classe na ontologia, e os atributos do elemento como atributos de classe. Essa conversão foi realizada por meio de um *script* em XSLT (*eXtensible Stylesheet Language for Transformation*). Os autores observam que essa transformação do modelo de processos de negócio visa preservar a sua semântica. Ainda, a estrutura do modelo de processo de negócios pode ser transformada com todos os seus objetos e seus atributos para uma ontologia de processo. Para essa ontologia, os autores definiram um metamodelo, apresentado na **Figura 14**.

Figura 14 - Metamodelo da ontologia de processo de Ternai, Török e Varga (2016).



Fonte: traduzido de Ternai, Török e Varga (2016).

De maneira semelhante a este trabalho, Ternai, Torök e Varga (2016) definem um metamodelo como base para a geração de uma ontologia de processos. Contudo, os autores não exploram todas os elementos de um modelo de processos de negócio e suas relações. Por exemplo, para as atividades sucedidas por um *gateway* é considerado apenas, no metamodelo, a relação “Seguido por”, indicando a sequência desses elementos no modelo. Por outro lado, este trabalho explora toda as relações entre uma atividade e um *gateway* presentes nos arquivos XPD, utilizados como entrada para o processo sistemático. Dessa forma, o metamodelo definido neste trabalho contempla relações além das que dizem respeito à sequência dos elementos em um modelo de processos de negócio. Além disso, o

metamodelo deste trabalho conta com mais classes do que o definido por Ternai, Torök e Varga (2016).

3.6 CONSIDERAÇÕES FINAIS

Neste capítulo foram apresentados conceitos básicos sobre ontologias e trabalhos que envolvem a geração de ontologias como forma de representação para modelos de processos de negócio. As ontologias contextualizam qualquer domínio de interesse. Elas podem representar diversos recursos, seja na forma textual ou visual. Entre esses recursos possíveis de se representar estão os processos de negócio.

Os processos de negócio quando modelados por meio de uma notação gráfica, como por exemplo BPMN, nem sempre são explicativos o suficiente para profissionais que não pertencem à área de negócios. Além disso, os modelos de processos de negócio apresentam conhecimento implícito que deve ser explorado. O uso de ontologias possibilita a estruturação desse conhecimento implícito e propõe uma representação legível por máquina dos modelos de processos de negócio.

Os estudos sobre linguagens para representação de ontologias, em especial a linguagem OWL, permitiram verificar que é possível estabelecer diversas relações entre as classes que compõem uma ontologia. Além disso, também é possível definir axiomas, que podem ser aproveitados para a realização de inferências sobre as classes da ontologia, gerando, assim, relações não explícitas antes da inferência. Já os estudos sobre a linguagem SPARQL mostraram que se pode retornar informações das mais simples às mais elaboradas provenientes das ontologias, conforme a necessidade.

As metodologias para construção de ontologias apresentadas neste capítulo orientaram o processo sistemático desenvolvido neste trabalho. Já os trabalhos relacionados apresentados neste capítulo serviram como inspiração para o processo sistemático desenvolvido para este trabalho. Entretanto, nenhum deles busca representar melhor as interdependências entre os elementos de um modelo de processos de negócio. Alguns desses trabalhos também não utilizam arquivos XPDL como entrada para a geração da ontologia. Vale ressaltar que, neste trabalho, tanto as definições das classes da ontologia gerada quanto as de suas relações para compor a ontologia foram realizadas visando explorar os elementos dos modelos de processos de negócio, assim como suas relações também. Essas definições auxiliam na geração de uma ontologia relevante.

4

GERAÇÃO AUTOMÁTICA DE ONTOLOGIAS A PARTIR DE MODELOS DE PROCESSOS DE NEGÓCIO

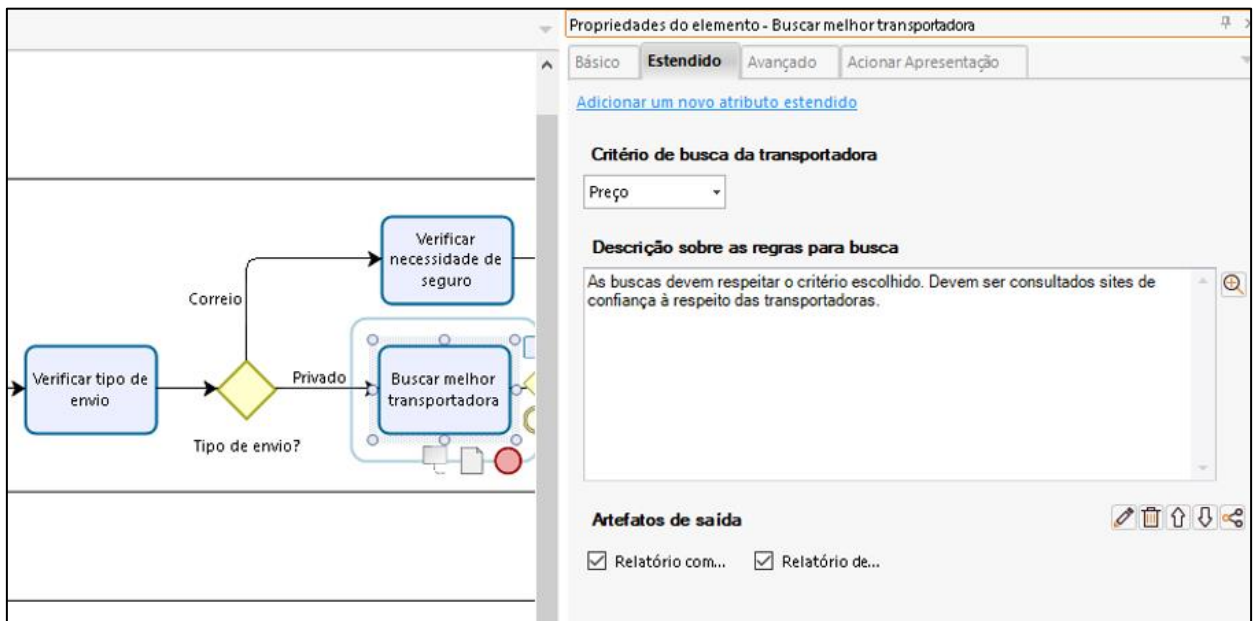
Este capítulo apresenta as ações e elementos envolvidos na sistematização da geração automática de ontologias de processos a partir de modelos de processos de negócio na notação BPMN v2.0. Tanto os elementos gráficos de BPMN como a documentação associada a esses elementos são devidamente mapeados para as classes da ontologia, evidenciando as relações e as interdependências entre esses elementos. Para isso, é importante que os elementos pertencentes aos modelos de processos apresentem documentação textual.

É importante ressaltar que os modelos de processo de negócios devem satisfazer doze critérios para que o mapeamento para a ontologia seja realizado corretamente. Esses critérios são definidos na *Seção 4.1*, de acordo com boas práticas de modelagem de processos de negócio, apresentadas na *Seção 2.4*. Quando um dos critérios não for satisfeito, então o modelo de processos de negócio deve ser ajustado pela equipe de modelagem para atender o critério. Esses tipos de ajustes proporcionam melhorias na documentação e na parte gráfica dos modelos de processos de negócio.

Adicionalmente, recomenda-se que os modelos de processos de negócio estejam em conformidade com determinadas heurísticas de negócio e de requisitos, selecionadas dos conjuntos de heurísticas definidos por Nogueira (2017) e Nogueira e Oliveira (2017a), bem como Nogueira e Oliveira (2017b). Os autores definiram 42 heurísticas de negócio e 9 heurísticas de requisitos, mas, para fins da composição da ontologia, recomenda-se pelo menos 11 heurísticas de negócio e 4 heurísticas de requisitos, apresentadas na *Seção 4.2*. Para estar em conformidade com esse conjunto mínimo de heurísticas, podem ser necessários ajustes no modelo de processos de negócio. Esses ajustes são opcionais, mas podem ajudar a fazer diferença nos resultados obtidos na ontologia gerada, de modo a torná-la mais representativa em relação às classes, relações e axiomas. Esses ajustes também contribuem para proporcionar melhorias textuais e gráficas nos modelos de processos de negócio, aumentando o conjunto de informações/conhecimento nos modelos e, conseqüentemente, nas ontologias correspondentes.

Neste trabalho, os atributos estendidos agregam importante valor à ontologia. Esses atributos não são disponíveis em todos os sistemas de modelagem BPMN, mas são muito úteis e seu uso é incentivado. Os atributos estendidos auxiliam na documentação do modelo de processos de negócio e na identificação de alguns elementos de software para este trabalho. A **Figura 15** mostra um exemplo de atributo estendido definido para uma atividade.

Figura 15 - Exemplo de atributo estendido definido para uma atividade utilizando o modelador Bizagi.



Fonte: elaborado pelo autor.

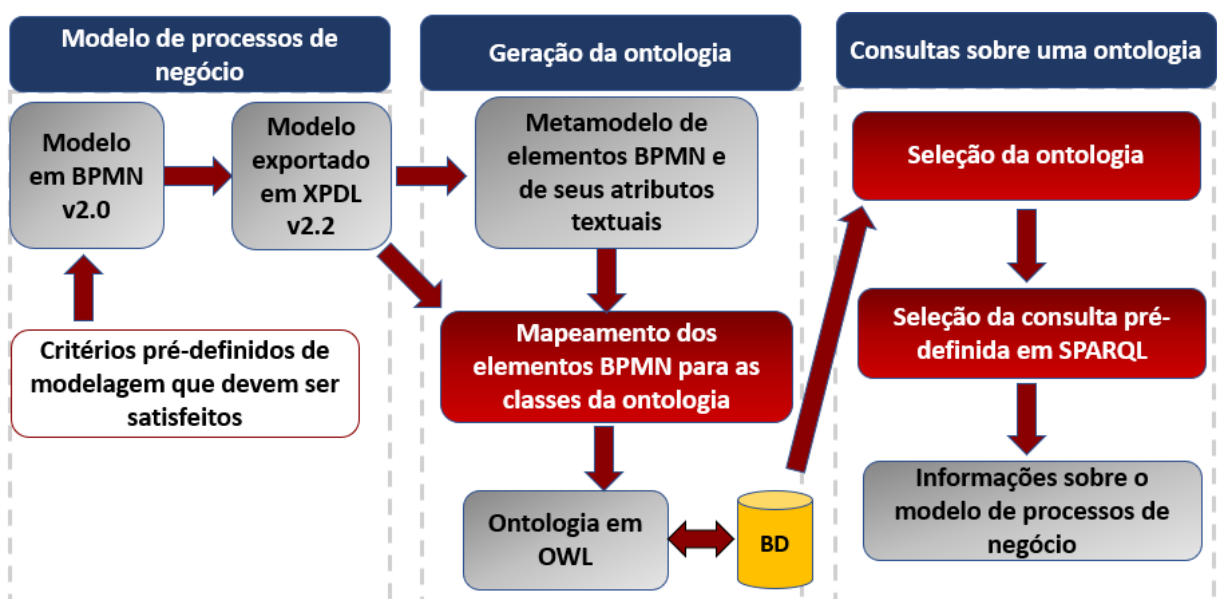
Os atributos estendidos podem ser representados por campos textuais, listas com opções de escolha, imagens, tabelas, entre outros. Os elementos dos modelos de processos de negócio que estão associados a um ou mais atributos estendidos poderiam ser identificados como um dos seguintes requisitos de software: regra de negócio, requisito funcional ou requisito não funcional. Esses requisitos podem vir representados na ontologia como propriedades de dados (“is_business_rule”, “is_functional_requirement” e “is_non_functional_requirement”) associadas aos conceitos representativos dos correspondentes elementos de BPMN. No entanto, isso é possível se as heurísticas de negócio e de requisitos forem satisfeitas, podendo requerer ajustes nos modelos de processos de negócio. Como essa recomendação é opcional, caso não sejam satisfeitas, não serão adicionadas, à ontologia, as propriedades de dados que indicam que um conceito é um requisito de software. Por meio de consultas pré-definidas neste trabalho sobre a ontologia, é possível saber, posteriormente, quais são os requisitos funcionais, não funcionais e as regras de negócio presentes no respectivo modelo de processo de negócio.

Em relação ao processo de geração das ontologias, esse é composto por três etapas, sendo uma delas opcional. Todas as etapas envolvem a utilização de pelo menos uma ferramenta de software. O processo para a geração da ontologia, de modo geral, assim como suas etapas, é apresentado na Seção 4.3.

Para que a ontologia gerada apresente tanto os elementos do modelo de processos de negócio quanto suas relações e interdependências, foi definido um processo de mapeamento. Esse processo utiliza um metamodelo denominado Meta2Onto, que é formado por classes referentes aos elementos XPD L e às relações entre esses elementos. Na Seção 4.4, é apresentado esse processo de mapeamento, enquanto o metamodelo completo pode ser visualizado no APÊNDICE C. Esse metamodelo possibilita a estruturação da ontologia de processos esperada, pois trata dos elementos de BPMN e suas relações com as classes, propriedades e axiomas que irão compor a ontologia.

A Seção 4.4 também apresenta uma visão geral da ferramenta PM2ONTO, que automatiza o processo sistemático apresentado. Essa ferramenta foi desenvolvida na linguagem Java e utiliza recursos do *framework* Apache JENA para a geração e consulta de ontologias. O APÊNDICE D apresenta uma breve definição sobre o *framework* Apache JENA, enquanto que o APÊNDICE E contém o código fonte da versão 1.0 da ferramenta PM2ONTO. Essa Seção apresenta, ainda, as consultas definidas em SPARQL sobre as ontologias geradas, para serem utilizadas por meio da ferramenta PM2ONTO. Na **Figura 16** pode ser visto um esquema do processo sistemático definido neste trabalho. Observa-se as considerações finais deste capítulo como um todo são apresentadas na Seção 4.5.

Figura 16 - Visão geral do processo sistemático para a geração das ontologias.



Fonte: elaborado pelo autor.

4.1 CRITÉRIOS NECESSÁRIOS PARA OS MODELOS DE PROCESSOS DE NEGÓCIO

Nesta seção, são apresentados os critérios definidos neste trabalho, os quais devem ser satisfeitos pelos modelos de processos de negócio antes de serem utilizados como entrada no sistema PM2ONTO, com ou sem ajustes no modelo original.

Os elementos de BPMN considerados nos modelos de processos de negócio para este trabalho são: atividades, subprocessos, eventos, *gateways* (exclusivos, inclusivos e paralelos), artefatos (grupos e anotações), objetos de dados, depósitos de dados, atributos estendidos, piscinas (*pools*), raias (*lanes*), fluxos de sequência e fluxos de mensagem. Esses elementos estão entre os 20% mais utilizados nos modelos de processos de negócio, segundo Borger (2012).

É importante que os elementos gráficos do modelo sejam associados à documentação textual. O formato dessa documentação pode seguir o formato do sistema de modelagem BPMN utilizado, uma vez que não há uma forma padronizada para se estruturar a informação textual dos elementos de BPMN.

Assim, os doze critérios, baseados em boas práticas de modelagem, são:

- **C1:** todo processo deve possuir um único evento de início e pelo menos um de fim, com nomes “Início” e “Fim”, respectivamente;
- **C2:** um processo deve conter pelo menos um participante;
- **C3:** a documentação textual de cada elemento de BPMN deve conter, no mínimo, uma de suas propriedades descritivas básicas: nome, descrição ou documentação;
- **C4:** o nome de uma atividade deve utilizar um verbo no infinitivo;
- **C5:** o tipo de uma atividade deve ser sempre definido;
- **C6:** elementos do mesmo tipo não devem possuir o mesmo nome;
- **C7:** uma atividade deve ser executada por pelo menos um ator.
- **C8:** as ações das atividades não devem ser atribuídas como nomes dos *gateways* (*gateways* representam a lógica de direcionamento após a tomada de uma decisão);
- **C9:** os fluxos de sequência de um *gateway* exclusivo devem ser nomeados, com exceção do caso em que só há dois fluxos representando “Sim” ou “Não”. Nesse caso, apenas um dos nomes pode ser explícito e o outro poderá ser deduzido;

- **C10:** os fluxos de sequência de um *gateway* inclusivo devem ser nomeados;
- **C11:** atividades que são precedidas por um *gateway* paralelo devem ser unidas no final por outro *gateway* do tipo paralelo;
- **C12:** atividades que são precedidas por um *gateway* inclusivo devem ser unidas ao final por outro *gateway* do tipo inclusivo.

Alguns desses critérios são baseados em algumas heurísticas de negócio definidas por Nogueira e Oliveira (2017a). O critério C6, por exemplo, foi baseado na heurística de negócio identificada como HNa3, que é uma heurística para atividades (item *What*), com a seguinte definição: “o nome da atividade deve ser único no modelo”. Um outro exemplo é o critério C7, baseado na heurística de negócio para atividades HNa1 (item *Who*): “cada atividade deve possuir a informação de seu executor”. Essas heurísticas, assim como as demais utilizadas neste trabalho, são apresentadas na próxima seção (*Subseção 4.2.1*).

4.2 RECOMENDAÇÕES PARA OS MODELOS DE PROCESSOS DE NEGÓCIO

Com a finalidade de proporcionar melhorias na documentação dos elementos de um modelo de processos de negócio, de modo a evidenciar suas relações e interdependências na ontologia gerada, recomenda-se satisfazer as heurísticas de negócio apresentadas na *Subseção 4.2.1*.

Por outro lado, para adicionar propriedades relacionadas a requisitos de software na ontologia, recomenda-se satisfazer as heurísticas de requisitos apresentadas na *Subseção 4.2.2*.

É importante ressaltar que a aplicação dessas heurísticas não implica em alteração nos fluxos dos modelos de processos de negócio - nem nos fluxos de sequência e nem nos de mensagens. Contudo, a aplicação das heurísticas requer esforço humano – não é feita automaticamente por enquanto. Apesar disso, os benefícios para o modelo de processos de negócio e para a correspondente ontologia são consideráveis.

4.2.1 Ajustes recomendados com base nas heurísticas de negócio

Das 42 heurísticas de negócio definidas por Nogueira e Oliveira (2017a), foram escolhidas onze delas, que se referem a eventos, a atividades, a fluxos de decisão e a

artefatos e repositórios de dados. Para a seleção dessas heurísticas, considerou-se a necessidade de identificação das relações entre elementos dos modelos de processos de negócio, bem como de identificação das regras de negócio a partir de alguns desses elementos. Na **Tabela 10**, são apresentadas as heurísticas de negócio consideradas neste trabalho. A aplicação dessas heurísticas pode ser feita pela equipe de modelagem dos processos de negócio, pois podem ser necessários alguns ajustes nos modelos. Esses ajustes não são voltados à documentação textual propriamente e não causam mudanças no contexto dos modelos, mas trazem melhorias à legibilidade.

Para se entender a identificação das heurísticas de negócio na **Tabela 10**, convém fazer as seguintes menções: (1) “HN” representa “Heurísticas de Negócio”; (2) a letra que vem na sequência tem a seguinte representação de tipos de elementos de BPMN: “e” para “eventos”, “a” para “atividades”, “g” para fluxos de decisão (*gateways*) e “d” para “repositório de dados”. A numeração que faz parte da identificação das heurísticas manteve a mesma apresentada por Nogueira e Oliveira (2017a). Observa-se que o autor define oito heurísticas de negócio para eventos, treze para atividades, onze para *gateways* e dez para repositórios de dados, num total de 42 heurísticas de negócio.

Na segunda coluna da **Tabela 10**, abaixo da especificação do tipo da heurística de negócio, encontram-se palavras que se referem a determinada parte da heurística de negócio. É que cada heurística de negócio é composta por seis partes, referentes às informações requeridas na técnica gerencial conhecida como 5W1H: *Who* (Quem), *What* (O quê), *Where* (Onde), *When* (Quando), *Why* (Por que), *How* (Como). Assim, de cada heurística de negócio selecionada de Nogueira e Oliveira (2017a), foi utilizada apenas a informação sobre a parte especificada na segunda coluna.

Caso essas heurísticas de negócio da **Tabela 10** sejam satisfeitas, é possível associar, aos conceitos que compõem a ontologia, a definição de propriedades de objetos que indicam relações entre os elementos do modelo de processos de negócio. A heurística HNa1, por exemplo, possibilita o mapeamento da associação entre uma atividade e o ator responsável por sua execução para a ontologia por meio da propriedade “isPerformedBy”. O critério C7 (Seção 4.1) foi baseado nessa heurística, mas ela foi colocada como recomendada ainda para reforçar esse requisito fundamental. De modo similar, a heurística HNa3 serviu de base para a definição do critério C6, a fim de evitar a geração de conceitos com nomes repetidos na ontologia.

As heurísticas HNa6, HNa9, HNa12, HNg5, HNg9 e HNg11, apresentadas na **Tabela 10**, permitem identificar regras de negócio. Dessa forma, com a aplicação do atributo estendido do tipo “REGRA”, o elemento associado a esse atributo será classificado na

ontologia como regra de negócio por meio da propriedade de dados “isBusinessRule” (Subseção 4.4.1). Já a heurística HNe5 possibilita o mapeamento da associação entre eventos e elementos que os antecedem no modelo de processos de negócio para a ontologia por meio das propriedades de objeto “isPreceededBy” e “isSucceededBy” (Subseção 4.4.1). As heurísticas HNd2 e HNd7 possibilitam o mapeamento das associações entre repositórios de dados e artefatos com atividades para a ontologia como axiomas, por meio das propriedades de objeto “usesInput” e “producesOutput” (Subseção 4.4.1).

Tabela 10 - Heurísticas de negócio utilizadas neste trabalho.

Heurística	Tipo	Descrição
HNa1	Heurística para atividades (item <i>Who</i>)	Cada atividade deve possuir a informação de seu executor.
HNa3	Heurística para atividades (item <i>What</i>)	O nome da atividade deve ser único no modelo.
HNe5	Heurística para eventos (item <i>When</i>)	A informação sobre o início do evento deve ser validada.
HNa6	Heurística para atividades (item <i>Why</i>)	Se regra de negócio envolvida, o atributo estendido do tipo “REGRA” deve ser usado.
HNa9	Heurística para atividades (item <i>When</i>)	Se regra de negócio envolvida, o atributo estendido do tipo “REGRA” deve ser usado.
HNa12	Heurística para atividades (item <i>How</i>)	Se regra de negócio envolvida, o atributo estendido do tipo “REGRA” deve ser usado.
HNg5	Heurística para fluxo de decisão (item <i>Why</i>)	O atributo estendido “REGRA” deve ser utilizado quando necessário mais informações sobre a regra de negócio.
HNg9	Heurística para fluxo de decisão (item <i>Where</i>)	O atributo estendido “REGRA” deve ser utilizado quando necessário mais informações sobre a regra de negócio.
HNg11	Heurística para fluxo de decisão (item <i>How</i>)	O atributo estendido “REGRA” deve ser utilizado quando necessário mais informações sobre a regra de negócio.
HNd2	Heurísticas para artefatos e repositório de dados (item <i>Who</i>)	O repositório de dados e os artefatos devem ser relacionados às atividades que manipulam suas informações.
HNd7	Heurísticas para artefatos e repositório de dados (item <i>When</i>)	O repositório de dados e os artefatos devem ser relacionados às atividades que manipulam suas informações.

Fonte: adaptado de Nogueira e Oliveira (2017a).

4.2.2 Ajustes recomendados com base nas heurísticas de requisitos

Com a intenção de se extrair regras de negócio e requisitos funcionais e não funcionais dos modelos de processos de negócio para a composição das ontologias correspondentes, recomenda-se satisfazer algumas heurísticas de requisitos, definidas por Nogueira (2017) e Nogueira e Oliveira (2017b). Das sete heurísticas de requisitos (HR) definidas pelos autores, foram selecionadas quatro, apresentadas na **Tabela 11**. A identificação numérica junto a “HR” segue a mesma numeração apresentada em Nogueira e Oliveira (2017b). A conformidade com essas heurísticas pode requerer ajustes em relação ao tipo de algumas atividades e na documentação de alguns elementos, que podem ser feitos com o apoio da equipe de TI.

As heurísticas da **Tabela 11** foram selecionadas por possibilitarem a identificação dos requisitos de software. A heurística H4, por exemplo, possibilita a classificação dos elementos da ontologia que apresentem o atributo estendido do tipo “RNF” como requisitos não funcionais, por meio da propriedade de dados “isFunctionalRequirement” (*Subseção 4.4.1*). A utilização das heurísticas HR5 e HR6 permite classificar os elementos que apresentem o atributo estendido do tipo “REGRA” como regras de negócio, por meio da propriedade de dados “isBusinessRule”. A heurística HR8, por sua vez, permite classificar as atividades do tipo “Usuário”, “Serviço” ou “Script” como requisitos funcionais, por meio da propriedade de dados “isNonFunctionalRequirement”.

Tabela 11 - Heurísticas de requisitos utilizadas neste trabalho.

Heurística	Tipo	Descrição
HR4	Heurística de requisitos não funcionais	- As atividades devem ter seu tipo selecionado: “Usuário”, “Serviço” ou “Script”. - Identificar nas atividades os atributos estendidos do tipo “RNF”.
HR5	Heurística de regras de negócio para fluxos de decisão	- Identificar os fluxos de decisão. - Identificar regras de negócio a partir da documentação desses fluxos.
HR6	Heurística de regras de negócio para atividades	- Identificar as atividades do tipo “Usuário”, “Serviço” ou “Script”. - Identificar regras de negócio a partir dos atributos estendidos do tipo “REGRA”.
HR8	Heurísticas para requisitos funcionais	- Identificar as atividades do tipo “Usuário”, “Serviço” ou “Script”. - Identificar requisitos funcionais a partir da documentação textual das atividades.

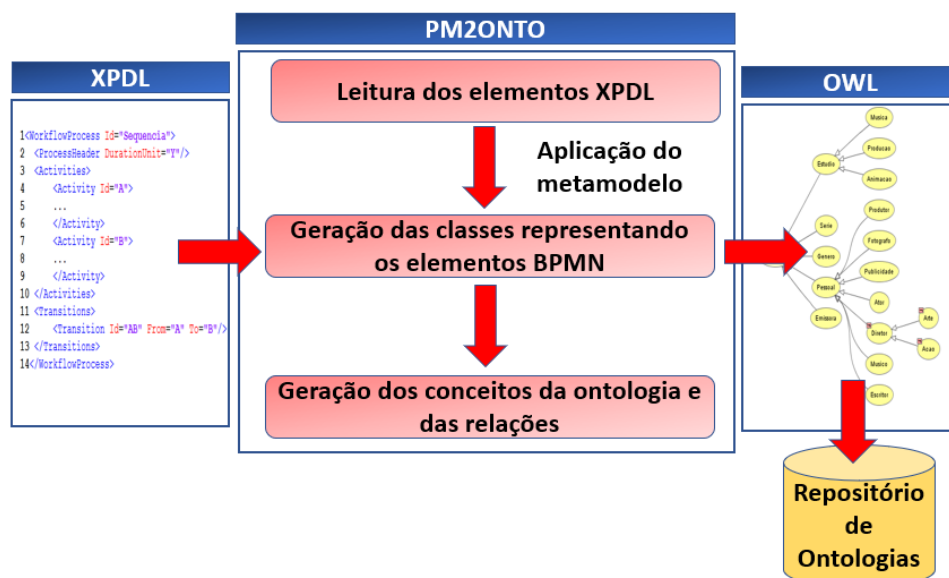
Fonte: adaptado de Nogueira e Oliveira (2017b).

4.3 PROCESSO SISTEMÁTICO PARA GERAÇÃO DAS ONTOLOGIAS

O processo para a geração da ontologia conta com três etapas, sendo a última delas opcional. Essas etapas devem ser executadas sequencialmente nesta ordem para um determinado modelo de processos de negócio, que deve estar em conformidade com os critérios apresentados na Seção 4.1:

- **Etapa 1:** o modelo de processos de negócio na notação BPMN v2.0 deve ser exportado para a linguagem XPDL v2.2 por meio de um sistema de modelagem de processos de negócio. Considerando que o modelo tenha “n” subprocessos, essa exportação gerará “n+1” arquivos XPDL. Isso acontece quando os subprocessos são definidos como processos invocados a partir do processo principal (como um subfluxo do processo principal);
- **Etapa 2:** a ferramenta PM2ONTO deve ser utilizada para gerar a ontologia automaticamente, usando os arquivos XPDL gerados na etapa anterior como entrada. O resultado será uma ontologia de processos definida na linguagem OWL, que representa o modelo de processos de negócio original. A ferramenta PM2ONTO executa as seguintes etapas nesta ordem, conforme ilustrado na **Figura 17**:

Figura 17 - Visão geral da ferramenta PM2ONTO.



Fonte: elaborado pelo autor.

- 2.1. Leitura e mapeamento dos elementos XPDL para as classes definidas a partir do metamodelo Meta2Onto, que contempla os elementos de BPMN neste

trabalho. É importante ressaltar que as relações existentes para os elementos XPDL que representam organizações externas (piscinas) ao processo principal do modelo não são considerados para a ontologia;

- 2.2. Geração dos conceitos da ontologia e das propriedades de relacionamento associadas a eles a partir das classes geradas na Etapa 2.1;
 - 2.3. Armazenamento do conteúdo da ontologia gerada em um banco de dados e disponibilização do arquivo OWL da ontologia para *download*;
- **Etapa 3:** a ferramenta *Protégé* (v. 5.0.0 ou superior) é utilizada para visualizar a ontologia e também para aplicar um dos *Reasoners* existentes nessa ferramenta. Essas estruturas contam com recursos para inferência. As inferências são feitas com base nos axiomas existentes para a ontologia gerada. Assim, é gerada a forma inferida dessa ontologia. Esta etapa é opcional e é recomendada para quando há necessidade de verificação de relações geradas após a inferência.

Em relação à Etapa 3, convém observar que um *Reasoner* é um mecanismo de inferência, definido como uma estrutura capaz de fazer inferências lógicas a partir de um conjunto de fatos semânticos ou axiomas existentes em uma ontologia (KIM et al., 2010). Eles também se baseiam na lógica descritiva para verificar a consistência da ontologia e computar automaticamente a hierarquia de classes.

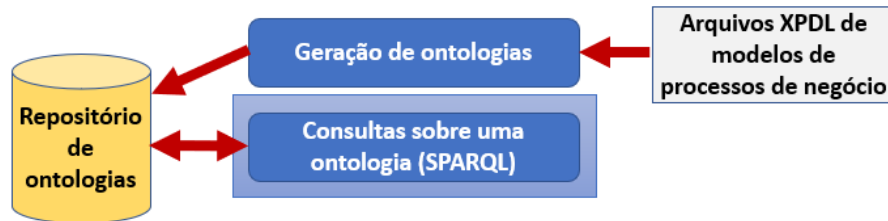
É importante ressaltar que, se os elementos de um modelo de processos de negócio forem atualizados e o modelo já tiver sido utilizado pelo processo sistemático anteriormente, então esse modelo deverá ser aplicado novamente no processo sistemático e uma nova ontologia será gerada. Observa-se que no *Capítulo 6*, como trabalhos futuros, será proposta uma pesquisa para permitir a atualização de ontologias já existentes quando se reutiliza um modelo de processos de negócio. A ideia é que se utilize um processo que otimize a atualização da ontologia com os novos elementos do modelo de processos de negócio, sem que haja o reprocessamento por completo desse modelo.

4.4 FERRAMENTA PM2ONTO

A ferramenta PM2ONTO, apresentada no APÊNDICE E (CD-R anexo), foi desenvolvida na linguagem Java v1.8 para ser utilizada a partir de um navegador Web. Além da geração de ontologias, PM2ONTO permite consultas sobre as ontologias geradas (**Figura 18**). As consultas são definidas na linguagem SPARQL e podem extrair informações básicas

dos elementos, bem como suas relações dentro de um modelo de processos de negócio. A **Figura 19** apresenta a tela da ferramenta PM2ONTO para geração das ontologias.

Figura 18 - Funcionalidades da ferramenta PM2ONTO.



Fonte: elaborado pelo autor.

Figura 19 - Tela da ferramenta PM2ONTO para a geração de ontologias.

Fonte: sistema PM2ONTO v1.0.

O *framework freeware* Apache JENA foi utilizado para a geração das ontologias e das consultas. O *framework* Hibernate (uma implementação da API Java Persistence) foi utilizado para persistência e mapeamento objeto-relacional do banco de dados; isso possibilitou o armazenamento das ontologias geradas em um banco de dados. O sistema de banco de dados utilizado foi o MySQL v. 5.7. A ferramenta *Protégé* (v. 5.0.0 ou superior) pode ser utilizada com a finalidade de visualizar e validar as ontologias geradas.

A *Subseção 4.4.1* apresenta como é o processo para mapear os elementos dos

arquivos XPDl para as classes da ontologia gerada. A *Subseção 4.4.2*, por sua vez, apresenta as consultas predefinidas na versão atual da ferramenta PM2ONTO.

4.4.1 Mapeamento das classes do metamodelo para as classes da ontologia

Um metamodelo, denominado Meta2Onto, foi definido para viabilizar o mapeamento dos elementos de BPMN para os conceitos da ontologia. A **Figura 20** mostra parte do metamodelo na notação UML, enquanto que a **Tabela 12** mostra as relações utilizadas no metamodelo.

Tabela 12 - Relações presentes no metamodelo Meta2Onto.

Relação	Elemento de origem	Elemento de destino
<i>Is succeeded by</i>	<i>Subprocess, Activity, Event, Gateway</i>	<i>Subprocess, Activity, Event, Gateway</i>
<i>Is executed by</i>	<i>Activity, Subprocess</i>	<i>Actor</i>
<i>Exchanges message with</i>	<i>SubProcess, Activity, Event, Gateway, Actor e Pool</i>	<i>SubProcess, Activity, Event, Gateway, Actor e Pool</i>
<i>Uses input</i>	<i>Activity</i>	<i>Artifact</i>
<i>Produces output</i>	<i>Activity, Event</i>	<i>Artifact</i>

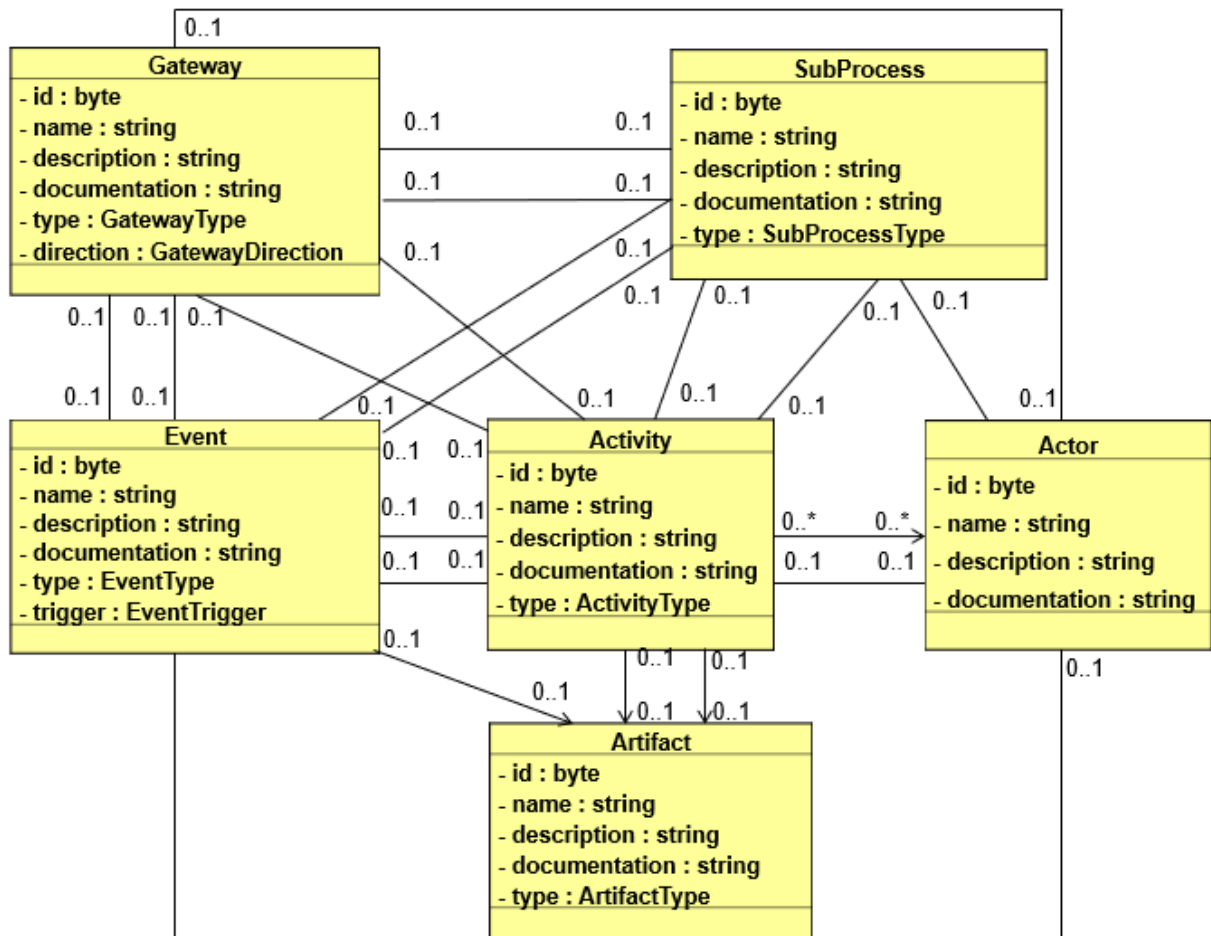
Fonte: elaborado pelo autor.

Como pode ser observado na **Figura 20**, todas as classes do metamodelo Meta2Onto, com exceção da classe “Actor”, têm pelo menos os seguintes atributos básicos: identificador, nome, descrição, documentação e tipo. A classe “Actor” não contém o atributo tipo, mas contém os demais atributos básicos mencionados.

O metamodelo Meta2Onto completo pode ser encontrado no APÊNDICE C. Ele é composto por 21 classes divididas em três categorias:

- **Classes principais**, que correspondem aos seguintes elementos de BPMN: “Process”, “Pool”, “Activity”, “SubProcess”, “Event”, “Gateway”, “Actor” e “Artifact”;
- **Classes auxiliares ou enumeradoras**, que definem os tipos e outras especificidades dos elementos de BPMN: “SubProcessType”, “ActivityType”, “GatewayType”, “EventType”, “ArtifactType”, “GatewayDirection”, “GatewayOutputs” e “EventTrigger”;
- **Classes de relação**, que definem a relação entre dois ou mais elementos de BPMN: “SucceededBy”, “UsesInput”, “ProducesOutput”, “ExchangesMessageWith” e “ExecutedBy”.

Figura 20 - Parte do metamodelo de classes Meta2Onto.



Fonte: elaborado pelo autor.

É importante mencionar que as informações definidas no arquivo XPDL sobre o posicionamento (coordenadas) relativo dos elementos no modelo de processos de negócio não foram consideradas neste trabalho. As informações consideradas relevantes para este trabalho nos arquivos XPDL dizem respeito aos elementos que indicam os processos, piscinas (pools), raias (lanes), subprocessos, atividades, gateways e eventos. As relações entre esses elementos, definidas como transições, associações, executores (de atividades), entre outras, também foram consideradas dos arquivos XPDL.

Quando os arquivos XPDL relacionados a um modelo de processos de negócio são inseridos como entrada na ferramenta PM2ONTO, os elementos de cada arquivo XPDL são analisados para a aplicação do metamodelo e geração da ontologia. Para isso, a **Tabela 13** mostra como foram feitas as correspondências entre os elementos XPDL e as classes do metamodelo. Os demais elementos dos arquivos XPDL não foram considerados para o mapeamento de classes neste trabalho. Os elementos XPDL selecionados são considerados básicos na modelagem de processos de negócio, ou seja, aparecem em praticamente todos

os modelos. As relações e interdependências desses elementos também foram incluídas no metamodelo, assim como os atributos textuais comuns dos elementos: identificador, nome, descrição, documentação e tipo. Exemplos de atributos que não foram considerados na versão do metamodelo neste trabalho: detalhes de implementação das atividades de um processo de negócio e atributos que referenciam informações gráficas dos elementos.

Tabela 13 - Correspondência entre os elementos XPDL e as classes do metamodelo Meta2Onto.

Elemento XPDL	Classe do metamodelo
<i>WorkflowProcess</i>	<i>Process</i>
<i>Pool</i>	<i>Pool</i>
<i>Performer</i>	<i>Actor</i>
<i>Activity</i> (com o elemento “Task” definido)	<i>Activity</i>
<i>Activity</i> (com o element “Event” definido)	<i>Event</i>
<i>Activity</i> (com o element “Route” definido)	<i>Gateway</i>
<i>ActivitySet</i>	<i>SubProcess</i>
<i>DataStore</i>	<i>Artifact</i>
<i>DataObject</i>	<i>Artifact</i>
<i>Artifact</i>	<i>Artifact</i>
<i>Artifact</i> (com o elemento “ArtifactType” definido como “Annotation”)	<i>Annotation</i>
<i>Artifact</i> (com o elemento “ArtifactType” definido como “Group”)	<i>Group</i>
<i>ExtendedAttribute</i>	<i>ExtendedAttribute</i>
<i>Transition</i>	<i>SucceededBy</i>
<i>MessageFlow</i>	<i>ExchangesMessageWith</i>
<i>Activity</i> (com o elemento “Performer” definido)	<i>ExecutedBy</i>
<i>Data Association</i>	<i>UsesInput/ProducesOutput</i>
<i>Data Store Reference</i>	<i>UsesInput/ProducesOutput</i>

Fonte: elaborado pelo autor.

Os elementos identificados como “Performer” foram mapeados para a classe “Actor” do metamodelo. Os elementos “DataStore”, “DataObject” e “Artifact” foram mapeados para a classe “Artifact” do metamodelo. Os elementos “Activity” de cada fluxo dos processos podem definir uma das seguintes classes do metamodelo: “Activity”, “Event” ou “Gateway”.

Os atores foram associados às atividades que executam por meio do mapeamento dos elementos “Performer”, quando encontrados dentro do elemento “Activity”; nesse caso, a relação entre os atores e suas atividades foram mapeadas para a classe “ExecutedBy” do metamodelo.

As ligações entre os objetos de dados e/ou depósitos de dados com as atividades ou com os eventos, definidas pelas associações “DataAssociation” e “DataStoreReference”, foram mapeadas para as classes “UsesInput” e “ProducesOutput”.

Os atores foram associados às atividades que executam por meio do mapeamento dos elementos “Performer”, quando encontrados dentro do elemento “Activity”; nesse caso, a relação entre os atores e suas atividades foram mapeadas para a classe “ExecutedBy” do metamodelo.

As ligações entre os objetos de dados e/ou depósitos de dados com as atividades ou com os eventos, definidas pelas associações “DataAssociation” e “DataStoreReference”, foram mapeadas para as classes “UsesInput” e “ProducesOutput”.

Quando as classes “Activity”, “Event” e “Gateway” apresentam armazenamento de atributos estendidos, eles devem ser mapeados para a classe “ExtendedAttribute”. Os atributos estendidos com nome “RN”, “RF” ou “RNF” foram definidos como regra de negócio, requisito funcional ou requisito não funcional, respectivamente. Na sequência, os elementos “ActivitySet” foram analisados e mapeados como subprocessos; para cada ocorrência de um “ActivitySet”, os passos anteriores de mapeamento de elementos de BPMN para as classes foram repetidos, visando definir todos os elementos pertencentes aos subprocessos.

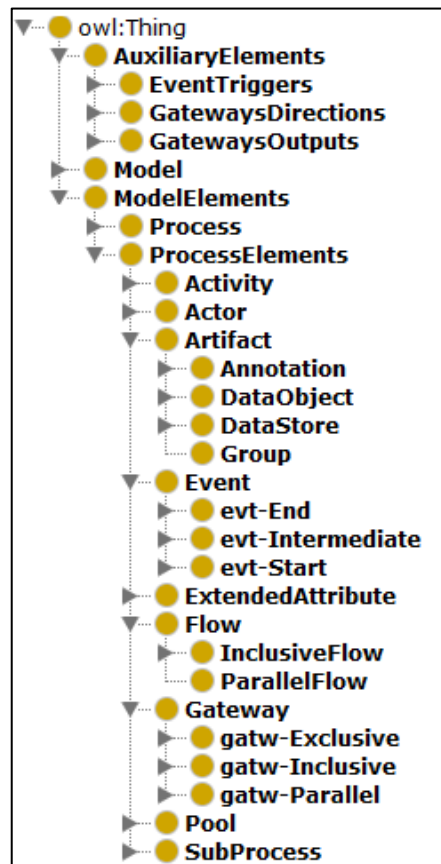
As transições (elemento “Transition”) que constam nos arquivos XPDL foram mapeadas para a classe “SucceededBy” do metamodelo, que é formada por dois atributos: elemento de origem e elemento de destino. Os fluxos de mensagem (elemento “MessageFlow”) foram mapeados para a classe “ExchangesMessageWith”, que também contém os atributos “elemento de origem” e “elemento de destino”.

As classes do metamodelo definem os conceitos e as relações que compõem a ontologia. As classes da ontologia são categorizadas em níveis de acordo com os elementos de BPMN a que se referem (**Figura 21**). Cada classe definida a partir do metamodelo é analisada e então incluída como uma nova classe na ontologia, sendo classificada de acordo com a categorização definida.

As classes do primeiro nível são apresentadas a seguir:

- **“AuxiliaryElements”**: classe composta pelas seguintes subclasses: “EventTriggers”, “GatewayDirections” e “GatewaysOutputs”. A classe “EventTriggers” define os gatilhos possíveis para disparo dos eventos. Já a classe “GatewayDirections” define as direções possíveis de um *gateway* (divergente ou convergente). A classe “GatewaysResponses” define os tipos de retornos possíveis dos *gateways*;

Figura 21 - Categorização das classes para a ontologia.



Fonte: elaborado pelo autor.

- **“Model”**: classe que define as informações principais do modelo de processos de negócio original;
- **“ModelElements”**: classe que contempla as classes dos processos de negócio propriamente dito (subclasses de “Process”) e de seus elementos (subclasses de “ProcessElements”). Esta classe é composta pelas classes de todos os elementos de um modelo. Está relacionada à classe “Model” por meio da propriedade de objeto “isComposedBy”, indicando que um modelo de processos de negócio é caracterizado por um conjunto de elementos.

A classe “ProcessElements” está presente no segundo nível e contém as subclasses: “Activity”, “Actor”, “Artifact”, “Event”, “ExtendedAttribute”, “Flow”, “Gateway”, “Pool” e “SubProcess”. É importante mencionar que as classes “Activity”, “Event”, “Flow”, “Gateway” e “SubProcess” são categorizados de acordo com o tipo a que pertencem. A classe “Gateway”, por exemplo, é categorizada em “Exclusive”, “Parallel” e “Inclusive”, pois são apenas considerados os *gateways* exclusivos, paralelos e inclusivos neste trabalho.

As seguintes propriedades de dados foram definidas para a ontologia com o objetivo de descrever melhor as classes: “id”, “name”, “description” e “documentation”. Também foram definidas as propriedades booleanas “isBusinessRule”, “isFunctionalRequirement” e “isNonFunctionalRequirement”, voltadas à definição de requisitos de software.

Convém observar que essas propriedades booleanas só serão viabilizadas se os modelos de processos de negócio estiverem em conformidade com as heurísticas de negócios e de requisitos, apresentadas nas *Subseções 4.2.1 e 4.2.2*, respectivamente. Essas propriedades são relacionadas aos atributos estendidos dos modelos de processos de negócio. Com a aplicação de algumas das heurísticas de negócios da **Tabela 10** (HNa6, HNa9, HNa12, HNg5, HNg9 e HNg11), além da aplicação das heurísticas de requisitos da **Tabela 11**, os atributos estendidos poderão ter a identificação de “RNF”, em alusão a Requisitos Não Funcionais, ou “NFR”, em alusão a *Non Functional Requirements*; também poderão ter a identificação “REGRA” ou “RULE”. Observa-se que a conformidade com as heurísticas de requisitos da **Tabela 11** identifica os requisitos funcionais com a informação sobre o tipo de determinada atividade: “Usuário”, “Serviço” ou “Script”.

Os atributos com a identificação “RNF” ou “NFR” serão representados na ontologia como requisitos não funcionais. Essa identificação será feita por meio da definição do valor booleano verdadeiro para a propriedade de dados “isNonFunctionalRequirement” das classes desses atributos estendidos na ontologia. É importante ressaltar que a definição do valor dessa propriedade será feita apenas se o atributo estendido de nome “RNF” ou “NFR” estiver associado a uma atividade.

Os atributos com a identificação “REGRA” ou “RULE” serão representados na ontologia como regras de negócio. A identificação ocorrerá por meio da definição do valor booleano verdadeiro para a propriedade de dados “isBusinessRule” das classes na ontologia dos atributos identificados.

Os atributos com a identificação “Usuário”, “Serviço” ou “Script” serão representados, na ontologia, por meio da definição do valor booleano verdadeiro para a propriedade de dados “isFunctionalRequirement” das classes das atividades que apresentarem esses tipos.

As classes da ontologia são relacionadas por meio de propriedades de objetos. Neste trabalho, essas propriedades relacionam duas ou mais classes da ontologia e são divididas em dois grupos:

- **Propriedades básicas:** definem as relações diretas que os elementos de BPMN podem assumir (**Tabela 14**). Elas são formadas a partir das transições, fluxos de mensagem e associações presentes nos arquivos XPD. Propriedades inversas foram definidas para algumas propriedades de objeto básicas, como é o caso de “isPerformedBy”, cuja propriedade inversa é “isExecutorOfActivity”;

Tabela 14 - Propriedades básicas de objeto definidas para a ontologia.

Propriedade	Domínio	Conjunto
<i>isComposedBy</i>	<i>Model</i>	<i>ModelElements</i>
<i>isPartOfProcess</i>	<i>Pool, SubProcess, Activity, Event, Gateway, Actor, Artifact</i>	<i>Process</i>
<i>isPartOfSubProcess</i>	<i>Activity, Event, Gateway, Actor, Artifact</i>	<i>SuProcess</i>
<i>isPartOfGroup</i>	<i>SubProcess, Activity, Event, Gateway, Actor, Artifact</i>	<i>Group</i>
<i>isSucceededBy</i>	<i>SubProcess, Activity, Event, Gateway</i>	<i>SubProcess, Activity, Event, Gateway</i>
<i>isPrecededBy</i>	<i>SubProcess, Activity, Event, Gateway</i>	<i>SubProcess, Activity, Event, Gateway</i>
<i>isPerformedBy</i>	<i>Activity</i>	<i>Actor</i>
<i>isExecutorOfActivity</i>	<i>Actor</i>	<i>Activity</i>
<i>usesInput</i>	<i>Activity</i>	<i>DataStore, DataObject</i>
<i>producesOutput</i>	<i>Activity, Event</i>	<i>DataStore, DataObject</i>
<i>hasExtendedAttribute</i>	<i>Activity, Gateway, Event</i>	<i>ExtendedAttribute</i>
<i>isAnnotatedBy</i>	<i>SubProcess, Activity, Event, Gateway</i>	<i>Annotation</i>
<i>communicatesWith</i>	<i>Pool, SubProcess, Activity, Event, Gateway</i>	<i>Pool, SubProcess, Activity, Event, Gateway</i>

Fonte: elaborado pelo autor.

- **Propriedades avançadas:** objetivam prover melhor entendimento das interdependências dos subprocessos, atividades, *gateways* e eventos presentes em um modelo de processos de negócio. A **Tabela 15** mostra as quatro propriedades avançadas definidas e que são explicadas a seguir.

Tabela 15 - Propriedades avançadas de objeto para a ontologia.

Propriedade	Domínio	Conjunto
<i>isActivateWhenEventIsTriggered</i>	<i>SubProcess, Activity, Event, Gateway</i>	<i>Event</i>
<i>isExecutedWhen[Nome da atividade antes do gateway]OutputIs</i>	<i>SubProcess, Activity</i>	<i>GatewaysOutputs</i>
<i>isExecutedAfterParallelExecutionOf</i>	<i>SubProcess, Activity</i>	<i>ParallelFlow</i>
<i>isExecutedAfterInclusiveExecutionOf</i>	<i>SubProcess, Activity</i>	<i>InclusiveFlow</i>

Fonte: elaborado pelo autor.

A propriedade avançada de objeto “isActivatedWhenEventIsTriggered” é definida para os elementos de BPMN que são precedidos por eventos. O objetivo esperado é indicar que esses elementos são invocados a partir do disparo ocasionado pelo evento relacionado. A propriedade de objeto “isExecutedWhen[Nome da Atividade]OutputIs” define a relação entre dois elementos BPMN interligados por um direcionamento de um *gateway* exclusivo. Essa relação tem por objetivo tornar mais clara a ligação entre o elemento que antecede o *gateway* exclusivo e o elemento após esse *gateway*. O elemento que sucede o *gateway* exclusivo é executado após o direcionamento do resultado da execução do primeiro elemento. É importante ressaltar que a parte “[Nome da Atividade]” do identificador da propriedade é referente ao nome da atividade que é executada antes do direcionamento realizado pelo *gateway* exclusivo. Dessa forma, a restrição para a aplicação dessa propriedade é a existência de uma atividade sucedida por um *gateway* exclusivo, desde que as saídas resultantes do direcionamento desse *gateway* estejam devidamente nomeadas.

A propriedade de objeto “isExecutedAfterParallelExecutionOf” define a relação entre um subprocesso, atividade, evento ou *gateway* com o fluxo paralelo que o antecede, caso isso ocorra. Observa-se que essa propriedade está relacionada com o critério C11, apresentado na Seção 4.1. Esse critério indica que todos os elementos existentes entre dois *gateways* paralelos definem fluxos paralelos. Todos os fluxos paralelos que devem ser executados antes de uma atividade serão incluídos como subclasse da classe “ParallelFlow” e associados à propriedade “isExecutedAfterParallelExecutionOf”.

A propriedade de objeto “isExecutedAfterInclusiveExecutionOf” apresenta praticamente as mesmas características da propriedade “isExecutedAfterParallelExecutionOf”. A única diferença é que sua restrição de uso é definida por dois *gateways* inclusivos. Todos os elementos existentes entre dois *gateways* inclusivos definem um fluxo inclusivo. Todos os fluxos inclusivos que devem ser executados antes de uma atividade serão incluídos como subclasse da classe “InclusiveFlow” e associados à propriedade “isExecutedAfterInclusiveExecutionOf”.

4.4.2 Consultas sobre as ontologias geradas

Algumas relações e interdependências não estão explícitas e não podem ser compreendidas apenas com a observação dos modelos de processos de negócio por meio das ferramentas de modelagem. Em um modelo extenso e de alta complexidade, a legibilidade de seus elementos e relações é afetada. As ferramentas de modelagem de processos de negócio também não apresentam recursos para a definição e execução de consultas sobre os modelos de processos de negócio.

Por outro lado, a representação por meio de uma ontologia possibilita a manipulação desses modelos computacionalmente. Consequentemente, possibilita consultas sobre os modelos de processos de negócio, explorando as classes e relações existentes na ontologia. Além disso, a representação por ontologias possibilita o uso de inferências, que podem ser obtidas inclusive com a execução de consultas definidas na linguagem SPARQL.

Assim, a ferramenta PM2ONTO provê um sistema de consultas predefinidas, estruturadas na linguagem SPARQL v1.1, para que os usuários obtenham informações a respeito da ontologia. Assim, os usuários não precisam construir uma consulta SPARQL manualmente.

As consultas são classificadas em dois grupos, de acordo com o tipo de informação que retornam: Básicas e Avançadas.

As consultas Básicas fornecem informações sobre os elementos de BPMN que foram mapeados como classes para a ontologia, sem levar em conta as relações definidas para essas classes. Essas consultas podem usar as seguintes propriedades de dados definidas para as classes como filtros: identificador, nome, descrição e documentação. Também é possível filtrar pelo atributo “tipo” as classes “SubProcess”, “Activity”, “Event”, “Gateway” e “Artifact”. As saídas esperadas para as consultas básicas são as informações principais sobre um ou mais elementos de BPMN, ou seja, identificador, nome, tipo, descrição e documentação. As consultas sobre qualquer elemento mapeado para a ontologia a partir do metamodelo podem conter filtros relacionados aos valores das suas propriedades de dados.

As consultas Avançadas foram inspiradas no trabalho de Fanesi, Cacciagrano e Hinkelmann (2015). Os autores apresentam alguns exemplos de consultas possíveis para a ontologia que definem em seu trabalho. Eles mencionam que essas consultas foram definidas e usadas para ajudar a entender os elementos necessários, bem como suas relações, em um modelo de processos de negócio. Entretanto, a maioria dessas consultas é feita somente sobre a relação entre as atividades e os executores delas no modelo. Outros elementos, como, por exemplo, *gateways* e eventos, não são considerados para a definição das

consultas. Ao contrário de Fanesi, Cacciagrano e Hinkelmann (2015), neste trabalho foram definidas consultas que relacionam, além de atividades e seus executores, os seguintes elementos de BPMN: piscinas, *gateways*, eventos, fluxos de sequência e de mensagem. Além disso, são explorados os atributos estendidos dos elementos de um modelo de processos de negócio.

As onze consultas Avançadas definidas neste trabalho fornecem informações sobre as relações e sobre as interdependências entre as classes geradas para os elementos de BPMN. As propriedades de objeto definidas para as classes da ontologia possibilitaram a definição das consultas Avançadas. Essas consultas estão descritas a seguir, mas também são apresentadas na **Tabela 16**, embora de uma forma mais resumida:

Tabela 16 - Síntese das consultas avançadas definidas no sistema PM2ONTO v1.0.

Nº da consulta	Descrição
1	Identifica quais atividades estão sobre a responsabilidade de um participante no processo.
2	Identifica o contexto em que cada elemento está inserido em um processo, com base em seu antecessor e sucessor.
3	Explicita a interação entre dois elementos de piscinas diferentes em um processo. A interação nesse caso é representada pela troca de mensagens.
4, 5 e 6	Identificam os elementos de um modelo de processos de negócio que representam requisitos de software, apoiando assim a especificação de requisitos de software.
7	Identifica os elementos que precisam de recursos para o seu devido funcionamento no contexto em que estão presentes no processo.
8	Identifica as atividades que geram recursos após a sua execução no contexto em que estão presentes no processo.
9, 10 e 11	Identificam as atividades de um processo que são executadas após uma tomada de decisão ou roteamento das ações realizadas no processo.

Fonte: elaborado pelo autor.

- 1. Atividades executadas por determinado ator:** atividades que estão associadas a um “Performer” no arquivo XPD, indicando que são executadas por um participante do processo;
- 2. Elemento antecessor/sucessor de um elemento filtrado na consulta:** consulta que retorna as informações do antecessor e do sucessor de um elemento no modelo de processos de negócio;
- 3. Elementos que realizam trocas de mensagens:** consulta que retorna os elementos que realizam troca de mensagens com elementos de outras piscinas (*pools*). Essa relação é verificada através do elemento de XPD “MessageFlow”;

4. **Elementos identificados como requisitos funcionais:** consulta que retorna os elementos identificados como requisitos funcionais na ontologia. Para isso são filtradas as classes da ontologia que apresentam a propriedade de dados booleana “isFunctionalRequirement” igual a verdadeiro;
5. **Elementos identificados como requisitos não funcionais:** consulta que retorna os elementos identificados como requisitos não funcionais na ontologia. Para isso são filtradas as classes da ontologia que apresentam a propriedade de dados booleana “isNonFunctionalRequirement” igual a verdadeiro;
6. **Elementos identificados como regras de negócios:** consulta que retorna os elementos identificados como regras de negócios na ontologia. Para isso são filtradas as classes da ontologia que apresentam a propriedade de dados booleana “isBusinessRule” igual a verdadeiro;
7. **Elementos que utilizam um artefato como entrada:** consulta que retorna os elementos que utilizam algum artefato como entrada. O artefato pode ser um objeto ou depósito de dados, enquanto que o elemento que o utiliza pode ser uma atividade ou evento. Essa relação é verificada através do elemento de XPDL “InputSet”, quando associado a um elemento “Activity” de XPDL;
8. **Atividades que produzem um artefato como saída:** consulta que retorna as atividades que produzem algum artefato como saída. O artefato pode ser um objeto ou depósito de dados. Essa relação é verificada através do elemento de XPDL “OutputSet”, quando associado a um elemento de XPDL “Activity”;
9. **Atividades precedidas por um *gateway* exclusivo:** consulta que retorna as atividades precedidas por uma decisão representada por um *gateway* exclusivo. Essa relação é verificada quando um elemento de XPDL “Transition” relaciona uma atividade e um *gateway* exclusivo;
10. **Atividades precedidas por um *gateway* inclusivo:** consulta que retorna as atividades precedidas por um fluxo inclusivo representado por um *gateway* inclusivo. Essa relação é verificada quando um elemento de XPDL “Transition” relaciona uma atividade e um *gateway* inclusivo;
11. **Atividades precedidas por um *gateway* paralelo:** consulta que retorna as atividades precedidas por um fluxo paralelo representado por um *gateway* paralelo. Essa relação é verificada quando um elemento de XPDL “Transition” relaciona uma atividade e um *gateway* paralelo.

Como exemplo do código desenvolvido para se efetivar as consultas avançadas, o código da consulta avançada “Atividades que produzem um artefato como saída”, identificada pelo número 8, está apresentado na **Figura 22**.

Figura 22 - Código SPARQL da consulta avançada nº 8.

```
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX owl: <http://www.w3.org/2002/07/owl#>
PREFIX xsd: <http://www.w3.org/2001/XMLSchema#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX j.0: <http://www.semanticweb.org/ontology/exampleOntology#>
PREFIX j.1: <http://www.semanticweb.org/ontology/exampleOntology#producesOutput#>

SELECT DISTINCT ?id ?name ?description ?documentation
WHERE
{
  ?x    rdfs:subClassOf    j.0:Activity ;
  |    j.0:id              ?id .
  |    j.0:name            ?name .
  |    j.0:description     ?description .
  |    j.0:documentation   ?documentation .
  ?activity rdfs:subClassOf ?activity_produces_output_restriction .
  ?activity_produces_output_restriction
    owl:onProperty j.1:
}
```

Fonte: elaborado pelo autor.

As consultas avançadas foram criadas com o objetivo de explorar as seguintes relações e interdependências entre os elementos de um modelo de processos de negócio, consideradas relevantes para esse trabalho: (1) relações das atividades com seus executores e o fluxo de decisão a que estão relacionadas; (2) atividades ou eventos que utilizam algum artefato como entrada para seu processamento; (3) atividades que produzem artefatos de saída após seu processamento; (4) elementos de diferentes piscinas (*pools*) que estão interligados através de trocas de mensagens; (5) antecessores e sucessores dos elementos no modelo de processos de negócio.

Além de melhorar a visualização das relações e interdependências, as consultas avançadas também podem auxiliar a identificação dos requisitos de software, pois permitem identificar os requisitos funcionais e não funcionais, bem como as regras de negócio (consultas nº 4, 5 e 6, respectivamente). Convém mencionar que as ontologias geradas em conjunto com o sistema de consultas apresentado podem auxiliar o levantamento e especificação de requisitos de uma solução de software.

Para que a interface do sistema seja melhor compreendida e utilizada pelo usuário, de acordo com o tipo de consulta são exibidos filtros na tela. Ao lado de cada um desses filtros há um símbolo de interrogação com explicações a respeito de cada campo que pode ser

informado para a realização das consultas. O resultado da consulta selecionada, juntamente com a combinação de filtros, é exibido em uma tabela.

O sistema de consultas deve ser utilizado em duas etapas: (1) seleção da ontologia; (2) escolha da consulta e dos filtros. Na primeira etapa, são listadas as ontologias geradas, que estão armazenadas em um banco de dados como arquivos em OWL. Na segunda etapa, é realizada a escolha da consulta pré-definida com os filtros, a fim de gerar uma instrução na linguagem SPARQL para a ontologia selecionada anteriormente. Nas **Figura 23** e **24** são apresentadas as telas de seleção da ontologia (Etapa 1) e dos filtros para uma consulta selecionada (Etapa 2), respectivamente.

Figura 23 - Tela do sistema PM2ONTO para seleção de uma ontologia para consultas.

Ontologias para consulta			
Exibindo 1-1 de 1 registros			
Id	Nome	Descrição	
1	AccountsPayable	Ontologia gerada a partir do modelo de processos de negócio "Contas a Pagar"	Remover

Fonte: sistema PM2ONTO v1.0.

Figura 24 - Tela do sistema PM2ONTO para seleção da consulta e respectivo filtro.

Os filtros marcados com (*) são obrigatórios

Selecione o Tipo de Consulta: * Avançada ?

Tipo Consulta Avançada: * Atividades precedidas por um gateway exclusivo ?

Atividade:

Pesquisar Limpar

Resultados da Pesquisa					
Exibindo 1 - 3 de 3 registros					
Atividade	Descrição	Tipo	Gateway	Descrição	Tipo
Check Invoice Products/Services	The financial assistant must check the products or services in the invoice in order to see if they require approval.	User Activity	Invoice match with PO	The gateway determines the next activity to be performed.	Exclusive Gateway
Return Products	The products are returned to the supplier's address.	Manual Activity	Products have to be returned	If previous steps in the process indicate that the products must be returned, the next task is Return Products. If not the process ends.	Exclusive Gateway

Fonte: sistema PM2ONTO v1.0.

4.5 CONSIDERAÇÕES FINAIS

Neste capítulo foi apresentado o processo sistemático para a geração automática de uma ontologia a partir de um modelo de processos de negócio, independente do seu nível de complexidade. Cada ontologia armazenada em um banco de dados pode ser utilizada para fornecer informações relevantes sobre o correspondente modelo de processos de negócio. Isso é feito por meio de um sistema de consultas, que conta com instruções pré-definidas na linguagem SPARQL, para exploração dos atributos, relações diretas e interdependências dos elementos do modelo de processos de negócio, os quais se encontram representados por conceitos na ontologia.

Os modelos de processos de negócio devem estar em conformidade com os doze critérios apresentados na *Seção 4.1*. Caso não estejam, devem ser ajustados, seja na parte textual ou gráfica. Conforme apresentado na *Seção 4.2*, recomenda-se que os modelos de processos de negócio estejam também em conformidade com algumas heurísticas de negócio apresentadas na seção, de modo a lhes proporcionar melhorias textuais e gráficas para melhor representatividade na ontologia. Caso seja de interesse que a ontologia represente requisitos funcionais, não funcionais e regras de negócio, recomenda-se a conformidade com as heurísticas de requisitos e com algumas heurísticas de negócio (HNa6, HNa9, HNa12, HNg5, HNg9 e HNg11). Esses requisitos e as regras de negócio serão representados por meio de propriedades de dados nas classes da ontologia.

O processo sistemático para a geração das ontologias é composto por três etapas, executadas na ordem definida, conforme apresentado na *Seção 4.3*. Na primeira etapa, é utilizada uma ferramenta de modelagem de processos de negócio para realizar a exportação de um modelo de processos de negócio em BPMN para arquivos na linguagem XPDL. Na segunda etapa, esses arquivos XPDL são utilizados como entrada no sistema PM2ONTO, que, utilizando o metamodelo Meta2Onto (*Subseção 4.4.1*), mapeia os elementos de XPDL para os elementos da ontologia em OWL e a armazena em um banco de dados. Na terceira etapa, opcional, a ferramenta Protégé, por exemplo, possibilita a visualização, validação e utilização do recurso de inferência sobre a ontologia. Entretanto, pode ser utilizada qualquer ferramenta de visualização e validação de ontologias, que apresente o recurso de inferência, com a aplicação dos *Reasoners* existentes na ferramenta escolhida.

5

AVALIAÇÃO DO PROCESSO SISTEMÁTICO

Este capítulo apresenta estudos de caso para a avaliação da aplicabilidade do processo sistemático apresentado no *Capítulo 4*, que possibilita a geração automática de uma ontologia correspondente a um modelo de processos de negócio em BPMN v2.0. Adicionalmente, este capítulo apresenta alguns procedimentos feitos para a avaliação do sistema de consultas sobre as ontologias geradas, também apresentado no *Capítulo 4*. Observa-se que outros testes foram realizados de forma incremental sobre o processo sistemático mencionado e sobre o sistema de consultas, de modo a avaliar partes específicas, de forma isolada e integrada.

Todos os modelos de processos de negócio considerados nos testes e estudos de caso satisfizeram os doze critérios apresentados na *Seção 4.1*. Adicionalmente, os modelos passaram por ajustes, quando necessário, para ficarem em conformidade com as heurísticas de negócio e de requisitos, apresentadas na *Seção 4.2*. Todos os modelos continham documentação textual associada aos elementos de BPMN. Contudo, essa documentação, assim como algumas relações entre os elementos do modelo, foi melhorada com os ajustes efetivados, o que impactou na representação das ontologias.

Seguindo a primeira etapa do processo sistemático, apresentadas na *Seção 4.3*, esses modelos foram exportados para XPDl com a utilização da ferramenta de modelagem de processos de negócio *Bizagi Modeler*, de uso gratuito. Como o sistema PM2ONTO foi desenvolvido para automatizar a segunda etapa do processo sistemático, a *Seção 5.1* apresenta uma visão geral da avaliação desse sistema.

Para verificar a aplicabilidade do sistema de consultas apresentado na *Subseção 4.4.2* sobre as ontologias geradas, foi elaborado um plano de avaliação, conforme apresentado na *Seção 5.2*. Esse plano leva em consideração um conjunto de propriedades (de dados e objetos) para cada consulta, que devem estar presentes nas ontologias para que a consulta possa ser realizada.

Seguindo a terceira etapa do processo sistemático definido neste trabalho, as ontologias geradas a partir dos modelos de processos de negócio considerados foram visualizadas e avaliadas com a ferramenta de domínio público *Protégé*. Adicionalmente, foram

utilizados os *Reasoners* disponíveis no *Protégé* para realizar inferências nas ontologias a partir de seus axiomas.

Os estudos de caso realizados estão apresentados nas Seções 5.3 a 5.5. Observa-se que todos os modelos de processos de negócio utilizados nesses estudos de caso apresentam mais de uma piscina, o que significa que as atividades são executadas por mais de um setor/departamento de uma organização. Algumas dessas atividades podem ser subdivididas em tarefas, que foram utilizadas na definição de subprocessos.

O primeiro estudo de caso, apresentado na Seção 5.3, foi realizado com um modelo da área de contas a pagar de uma organização, nomeado como “Contas a Pagar”. Esse modelo foi obtido do repositório público de modelos de processos de negócio da empresa de desenvolvimento de software Bizagi, disponível no endereço: <http://www.bizagi.com/en/community/process-xchange>. Esse repositório conta com modelos de diferentes áreas de negócio, apresentando complexidades diferentes também. O modelo “Contas a Pagar” foi escolhido por apresentar um subprocesso, além de outros tipos de elementos de BPMN, como atividades, *gateways*, eventos e artefatos. Dessa forma, foi possível avaliar o mapeamento desses elementos e de suas relações para a ontologia.

Na Seção 5.4, é apresentado o segundo estudo de caso, que considerou o modelo de processos de negócio denominado “Processo de Viagem”. Ele se refere às áreas de uma organização que providenciam viagens aos seus profissionais, conforme a necessidade. Esse modelo foi obtido a partir de um ambiente de treinamento do *site* da empresa especializada em gestão por processos iProcess, disponível no endereço: <http://iprocess.com.br>.

O terceiro estudo de caso, apresentado na Seção 5.5, tomou como entrada um modelo de processo de negócio denominado “Compra de Livros”. Esse modelo é resultado do mapeamento do processo de compra de livros da Biblioteca da Universidade Estadual Paulista Júlio de Mesquita Filho (Unesp) no Câmpus de Rio Claro (PUCCI, 2016). A autora, Maria Anatania Franco de Souza Pucci, pertenceu ao grupo de pesquisa coordenado pela orientadora deste trabalho: LesTIC-Laboratório de Engenharia de Software e Tecnologias da Informação e Comunicação. Esse modelo foi desenvolvido com boas práticas de modelagem e possui maior quantidade de subprocessos e piscinas que os modelos dos estudos de caso anteriores.

Uma avaliação geral dos resultados obtidos nos três estudos de caso encontra-se apresentada na Seção 5.6. Os três modelos de processos de negócio dos estudos de caso estão apresentados no APÊNDICE F, em formato digital (CD-R anexo), tanto no formato em BPMN v2.0 quanto em XPD L v2.2. As ontologias geradas, descritas na linguagem OWL, estão disponíveis no APÊNDICE G, também em formato digital (CD-R anexo).

5.1 AVALIAÇÃO DO PROCESSO DE GERAÇÃO DAS ONTOLOGIAS

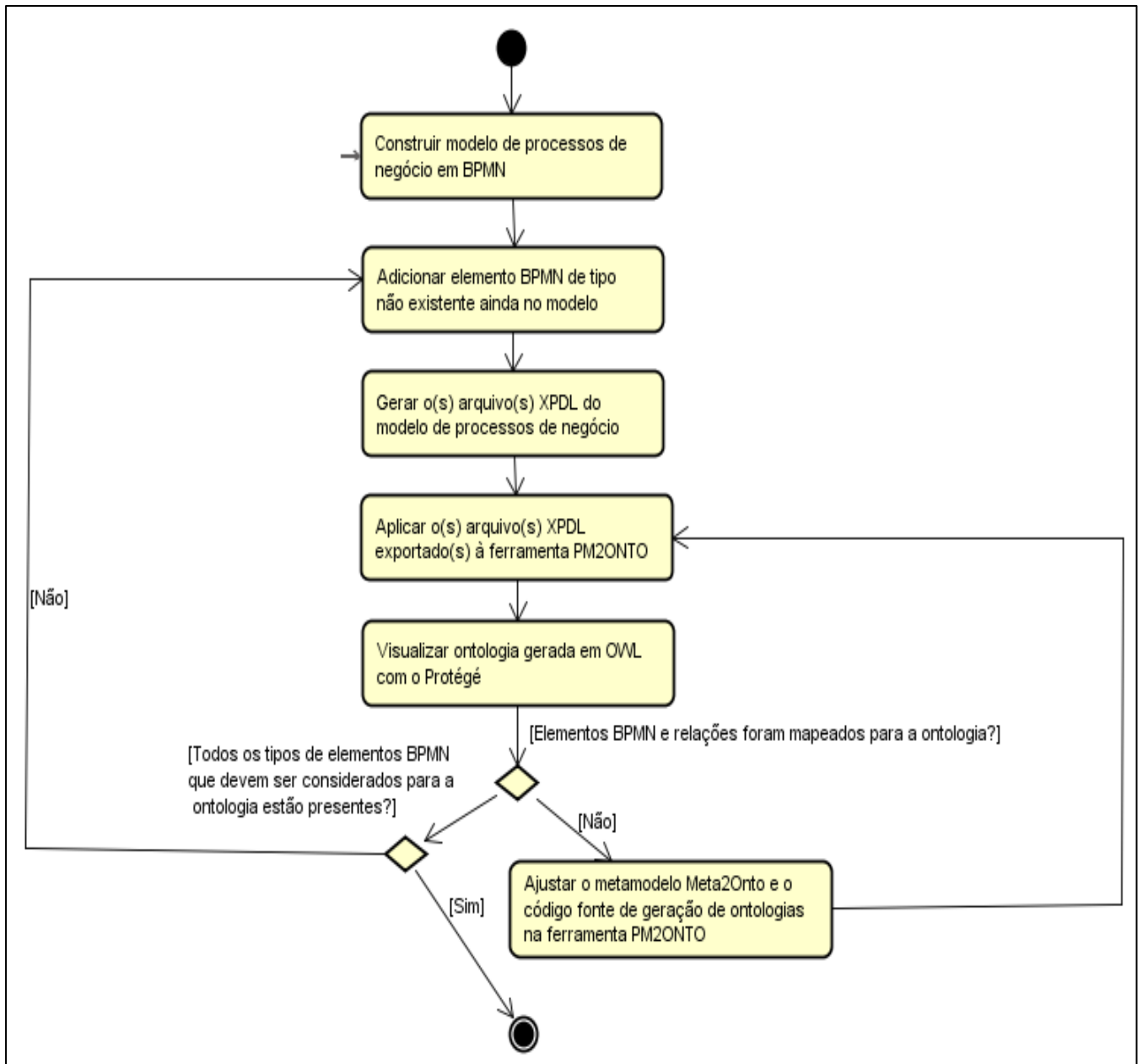
Durante o desenvolvimento da parte de geração de ontologias do sistema PM2ONTO, foram feitos vários testes como meio de analisar os resultados e então realizar possíveis correções no sistema.

Os testes realizados foram feitos com vários modelos de negócio, incluindo aqueles com o mínimo de elementos em BPMN a uns mais complexos, com um ou vários subprocessos. Esses testes foram utilizados como testes de validação do sistema PM2ONTO, organizados de maneira que a complexidade dos modelos fosse sendo aumentada gradualmente, com a inserção de novos elementos de BPMN (incluindo variações dos elementos) e atributos de documentação textual.

As falhas encontradas foram devidamente avaliadas, com a atenção de se verificar os pontos associados à ocorrência de cada falha, abrangendo desde o código ao metamodelo Meta2Onto. As correções no software resultaram na versão 1.0 do sistema PM2ONTO, apresentada neste trabalho no APÊNDICE E.

Na **Figura 25** é apresentada uma visão geral das atividades realizadas na aplicação dos testes, incluindo as atividades de correção do sistema. Na **Tabela 17** são mostrados os elementos de BPMN que foram avaliados nos testes realizados. As variações desses elementos estão dentro dos parênteses na frente de cada tipo de elemento na primeira coluna. De forma similar, são mostradas também os tipos de relações consideradas nos modelos de BPMN.

A cada modelo de processos de negócio utilizado como entrada no sistema PM2ONTO, a ontologia foi analisada por meio do sistema *Protégé*. Cada ontologia gerada foi validada como uma ontologia em OWL no sistema *Protégé* e visualizada. Em cada caso foi verificado se todos os elementos de BPMN do modelo de processos de negócio foram mapeados para a ontologia adequadamente (classes, relações e documentação). Essa verificação foi realizada com o auxílio de um mapa de itens que teriam que ser contemplados na ontologia. Os itens verificados seguiram a ideia do mapeamento apresentado na *Subseção 4.4.1*. Dessa forma, cada item foi associado a uma classe, atributo textual da classe ou relação existente no metamodelo Meta2Onto. Os itens foram verificados individualmente, sendo que todos eles deveriam constar na ontologia avaliada.

Figura 25 - Atividades para a geração de ontologias.

Fonte: elaborado pelo autor.

Tabela 17 - Elementos/relações dos modelos em BPMN testados nas ontologias geradas.

Elementos de BPMN	Classes mapeadas para a ontologia
Evento (Início, Intermediário, Fim)	<i>Event</i> (subclasses <i>Start</i> , <i>Intermediate</i> e <i>End</i>)
Atividade (Simples, Usuário, Script, Recebimento, Envio, Manual e Regra de Negócio)	<i>Activity</i> (subclasses <i>Simple</i> , <i>User</i> , <i>Script</i> , <i>Receive</i> , <i>Send</i> , <i>Manual</i> e <i>BusinessRule</i>)
Piscina	<i>Pool</i>
Gateway (Exclusivo, Paralelo, Inclusivo, Baseado em Evento, Exclusivo Baseado em Evento, Paralelo Baseado em Evento e Complexo)	<i>Gateway</i> (subclasses <i>Exclusive</i> , <i>Parallel</i> , <i>Inclusive</i> , <i>EventBased</i> , <i>ExclusiveEventBased</i> , <i>ParallelEventBased</i> e <i>Complex</i>)
Subprocesso (Reutilizável, Transacional, Adhoc, Loop, Múltiplas Instâncias de Sequência e Paralelas, Simples, Evento)	<i>SubProcess</i> (subclasses <i>Reusable</i> , <i>Transactional</i> , <i>Adhoc</i> , <i>Loop</i> , <i>ParallelMultipleInstances</i> , <i>SequenceMultipleInstances</i> , <i>Simple</i> e <i>Event</i>)
Artefato (Objeto de Dados, Repositório de Dados, Anotação)	<i>Artifact</i> (subclasses <i>DataObject</i> , <i>DataStore</i> e <i>Annotation</i>)
Participante do processo	<i>Actor</i>
Atributo Estendido	<i>ExtendedAttribute</i>
Relações entre os elementos BPMN	Relações mapeadas para a ontologia
Fluxo de sequência	<i>isSucceededBy</i> e <i>isPrecededBy</i>
Fluxo de sequência entre um <i>gateway</i> exclusivo e uma atividade	<i>isExecutedWhen</i> [Nome da atividade antes do <i>gateway</i>] <i>Outputs</i>
Fluxo de sequência entre um <i>gateway</i> paralelo e uma atividade	<i>isExecutedAfterParallelExecutionOf</i>
Fluxo de sequência entre um <i>gateway</i> inclusivo e uma atividade	<i>isExecutedAfterInclusiveExecutionOf</i>
Fluxo de sequência entre um evento e um elemento	<i>isActivatedWhenEventIsTriggered</i>
Fluxo de mensagem	<i>communicatesWith</i>
Associação de artefatos com atividades/evento	<i>usesInput</i> e <i>producesOutput</i>
Associação de artefatos de anotação com atividades/subprocessos/eventos/ <i>gateways</i>	<i>isAnnotatedBy</i>
Associação de um grupo com seus elementos	<i>isPartOfGroup</i>
Associação de uma atividade com um participante	<i>isExecutedBy</i> e <i>isExecutorOfActivity</i> (relação inversa)

Fonte: elaborado pelo autor.

5.2 AVALIAÇÃO DAS CONSULTAS SOBRE AS ONTOLOGIAS GERADAS

A avaliação das consultas sobre as ontologias geradas, presentes no sistema PM2ONTO, requereu que as ontologias em OWL estivessem devidamente armazenadas no repositório das ontologias. Convém mencionar que para esse armazenamento e acesso foi utilizado o sistema de banco de dados MySQL v 5.7.20. Além disso, foi pressuposto que as ontologias consultadas refletissem exatamente o modelo de processos de negócio de entrada, levando em consideração os elementos BPMN e suas relações no modelo. Nessa fase de avaliação, as ontologias já haviam sido analisadas, seguindo os procedimentos mencionados na Seção 5.1.

Assim, foi elaborado um “Plano de avaliação” para verificar os resultados das consultas pré-definidas (Básicas e Avançadas), apresentadas na Subseção 4.4.2. Para isso foram consideradas ontologias com variados níveis de complexidade. Esse Plano está sumariamente apresentado na **Tabela 18**. A meta da avaliação utilizada foi que uma consulta retornasse as informações corretas em relação ao modelo de processo de negócios de origem, considerando que o mapeamento tenha sido feito adequadamente. Observa-se que os resultados das consultas, sejam Básicas ou Avançadas, consistem em um texto organizado com as informações textuais dos elementos do modelo de processos de negócio consultado. Essas informações são formadas pelos atributos textuais de identificação, nome, descrição e documentação dos elementos do modelo.

Tabela 18 - Plano de avaliação das consultas SPARQL (continua...).

Consulta	Propriedades de dados necessárias	Propriedades de objetos/axiomas necessários	Resultado esperado da consulta
Básica	- <i>id</i> - <i>name</i> - <i>description</i> - <i>documentation</i>	- <i>subClassOf</i>	- <i>id</i> - <i>name</i> - <i>description</i> - <i>documentation</i> - <i>type</i> - <i>supertype</i>
Atividades executadas por determinado ator	- <i>name</i> (da atividade e do ator) - <i>description</i> (da atividade e do ator)	- <i>isPerformedBy</i> - <i>subClassOf</i>	- <i>name</i> (da atividade e do ator) - <i>description</i> (da atividade e do ator) - <i>type</i> (da atividade e do ator) - <i>supertype</i> (da atividade e do ator)

(... continuação) **Tabela 18** - Plano de avaliação das consultas SPARQL (continua...).

Consulta	Propriedades de dados necessárias	Propriedades de objetos/axiomas necessários	Resultado esperado da consulta
Elemento antecessor/sucessor de um elemento filtrado na consulta	- <i>name</i> (do elemento principal e de seu successor e antecessor) - <i>description</i> (do elemento principal e de seu successor e antecessor)	- <i>isSucceededBy</i> - <i>isPrecededBy</i> - <i>subClassOf</i>	- <i>name</i> (do elemento principal e de seu successor e antecessor) - <i>description</i> (do elemento principal e de seu successor e antecessor) - <i>type</i> (do elemento principal e de seu successor e antecessor) - <i>supertype</i> (do elemento principal e de seu successor e antecessor)
Elementos que realizam trocas de mensagens	- <i>name</i> (do elemento de origem e do elemento de destino) - <i>description</i> (do elemento de origem e do elemento de destino)	- <i>communicatesWith</i> - <i>subClassOf</i>	- <i>name</i> (do elemento de origem e do elemento de destino) - <i>description</i> (do elemento de origem e do elemento de destino) - <i>type</i> (do elemento de origem e do elemento de destino) - <i>supertype</i> (do elemento de origem e do elemento de destino)
Elementos identificados como requisitos funcionais	- <i>name</i> - <i>description</i> - <i>is_functional_requirement</i>	- <i>subClassOf</i>	- <i>name</i> - <i>description</i> - <i>type</i> - <i>supertype</i>
Elementos identificados como requisitos não funcionais	- <i>name</i> - <i>description</i> - <i>is_non_functional_requirement</i>	- <i>subClassOf</i>	- <i>name</i> - <i>description</i> - <i>type</i> - <i>supertype</i>
Elementos identificados como regras de negócios	- <i>name</i> - <i>description</i> - <i>is_business_rule</i>	- <i>subClassOf</i>	- <i>name</i> - <i>description</i> - <i>type</i> - <i>supertype</i>
Elementos que utilizam um artefato como entrada	- <i>name</i> (do elemento e do artefato) - <i>description</i> (do elemento e do artefato)	- <i>subClassOf</i> - <i>usesInput</i>	- <i>name</i> (do elemento e do artefato) - <i>description</i> (do elemento e do artefato) - <i>type</i> (do elemento e do artefato) - <i>supertype</i> (do elemento e do artefato)

(... continuação) **Tabela 18** - Plano de avaliação das consultas SPARQL.

Consulta	Propriedades de dados necessárias	Propriedades de objetos/axiomas necessários	Resultado esperado da consulta
Atividades que produzem um artefato como saída	- <i>name</i> (da atividade e do artefato) - <i>description</i> (da atividade e do artefato)	- <i>subClassOf</i> - <i>producesOutput</i>	- <i>name</i> (da atividade e do artefato) - <i>description</i> (da atividade e do artefato) - <i>type</i> (da atividade e do artefato) - <i>supertype</i> (da atividade e do artefato)
Atividades precedidas por um gateway exclusivo/inclusivo/paralelo	- <i>name</i> (da atividade e do gateway) - <i>description</i> (da atividade e do gateway)	- <i>subClassOf</i> - <i>isExecutedWhen</i>[Nome da Atividade]<i>OutputIs</i> - <i>isExecutedAfterInclusiveExecutionOf</i> - <i>isExecutedAfterParallelExecutionOf</i>	- <i>name</i> (da atividade e do gateway) - <i>description</i> (da atividade e do gateway) - <i>type</i> (da atividade e do gateway) - <i>supertype</i> (da atividade e do gateway)

Fonte: elaborado pelo autor.

Para a aplicação do “Plano de avaliação”, as ontologias geradas tiveram que apresentar condições mínimas quanto às suas classes e relações. Nas classes, deveria estar correto o mapeamento dos atributos textuais dos elementos dos modelos de processos de negócio para as propriedades de dados “id”, “name”, “description” e “documentation” das classes das ontologias geradas. Por sua vez, nas relações, deveria estar correto o mapeamento da sequência dos elementos no modelo de processos de negócio para os axiomas “isSucceededBy” e “isPrecededBy”, relacionando as classes das ontologias geradas. Também foi necessário que as classes estivessem corretamente categorizadas e organizadas hierarquicamente, seja como superclasses ou subclasses. Ainda, para as consultas do tipo Avançadas, as ontologias tiveram que ter as relações e interdependências dos elementos de um modelo de processos de negócio corretamente mapeados para a ontologia. Esse tipo de mapeamento estava presente nas ontologias por meio das propriedades de objeto, relacionando duas classes.

Durante os testes, todas as falhas encontradas foram avaliadas, com identificação dos pontos a serem corrigidos no sistema. Todos os erros foram anotados, bem como as correções realizadas, de modo a se manter uma base histórica dos problemas ocorridos.

5.3 ESTUDO DE CASO 1: MODELO “CONTAS A PAGAR”

O estudo de caso apresentado considerou o modelo de processos de negócio “Contas a Pagar” de uma organização, extraído do seguinte repositório público da empresa Bizagi: <https://www.bizagi.com/en/community/process-xchange>. Esse modelo é composto basicamente por atividades de validação de documentos (ex.: faturas) enviados por fornecedores, para que haja a liberação do pagamento posteriormente.

Esse modelo conta com quatro raias: setor de contabilidade, áreas do gerente administrativo, do assistente financeiro e da recepção. A **Figura 26** apresenta o processo principal do modelo de processos de negócios em questão. O modelo completo está disponível no APÊNDICE F.

Esse modelo completo inclui um subprocesso: “*Return Invoice to Supplier*”. É importante destacar que esse subprocesso não é um processo tratado como subfluxo do processo principal, ou sejam não é invocado a partir do processo principal. Dessa forma, ao se exportar para XPDL o modelo “Contas a Pagar” (Etapa 1 do processo sistemático), foi gerado apenas um arquivo XPDL.

Ao todo, o modelo “Contas a Pagar” conta com os seguintes elementos:

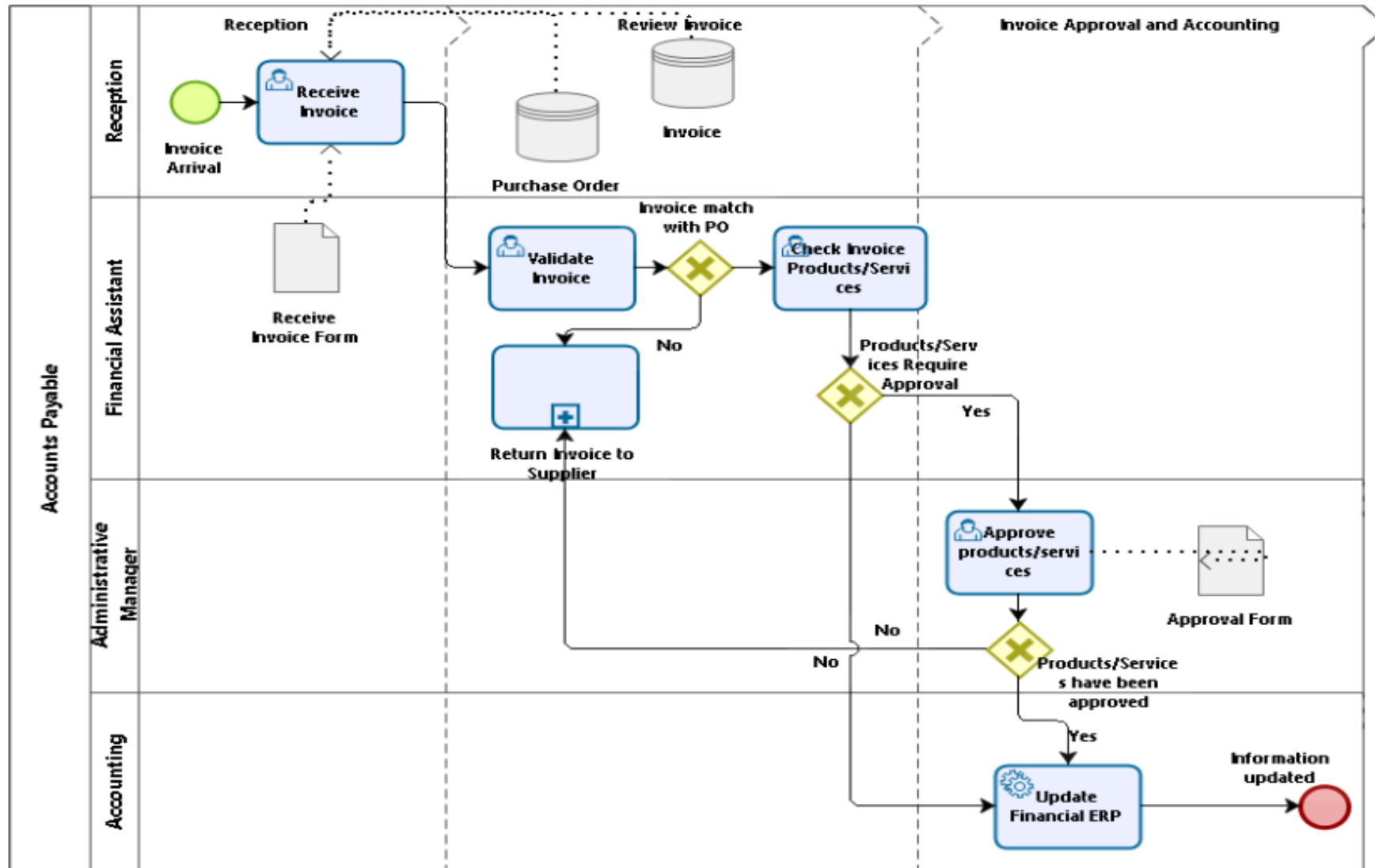
- 1 piscina;
- 4 participantes;
- 8 atividades (5 do tipo “Usuário”, 1 do tipo “Script”, 1 do tipo “Manual” e 1 do tipo “Serviço”);
- 5 eventos (2 do tipo “Início” e 3 do tipo “Fim”);
- 4 *gateways* (4 do tipo “Exclusivo”);
- 4 objetos de dados;
- 2 repositórios de dados;
- nenhuma anotação;
- 4 atributos estendidos.

Análise da conformidade com os critérios definidos

O modelo “Contas a Pagar” requereu poucas modificações para ficar em conformidade com os doze critérios definidos na *Seção 4.1*. Nesse modelo foram necessários os seguintes ajustes, relativamente aos critérios indicados:

- **C2:** foi necessário definir os participantes do processo “Contas a Pagar”;
- **C3:** alguns elementos, como, por exemplo, os “eventos de fim” do processo principal e do subprocesso, foram ajustados com definição de nome e documentação;

Figura 26 - Modelo de processos de negócio "Contas a Pagar".



Fonte: repositório público: <https://www.bizagi.com/en/community/process-xchange>.

- **C5:** as atividades “*Receive Invoice*” e “*Justify the rejection*” foram ajustadas com a definição dos tipos de cada uma delas;
- **C7:** todas as atividades foram associadas a um determinado ator para execução dessas atividades.

Aplicação das heurísticas de negócio

Quando as heurísticas de negócio da **Tabela 10** foram aplicadas ao modelo, evidenciaram a necessidade de algumas alterações. As heurísticas de negócio do tipo HNa evidenciaram a necessidade de se identificar os tipos das seguintes atividades: “*Receive Invoice*”, “*Validate Invoice*” e “*Approve products/services*”. Essas atividades foram marcadas com o tipo “Usuário”, após a análise da documentação de cada uma delas. Essas heurísticas HNa também permitiram a identificação de requisitos não funcionais, com base na documentação das atividades do modelo.

Aplicação das heurísticas de requisitos

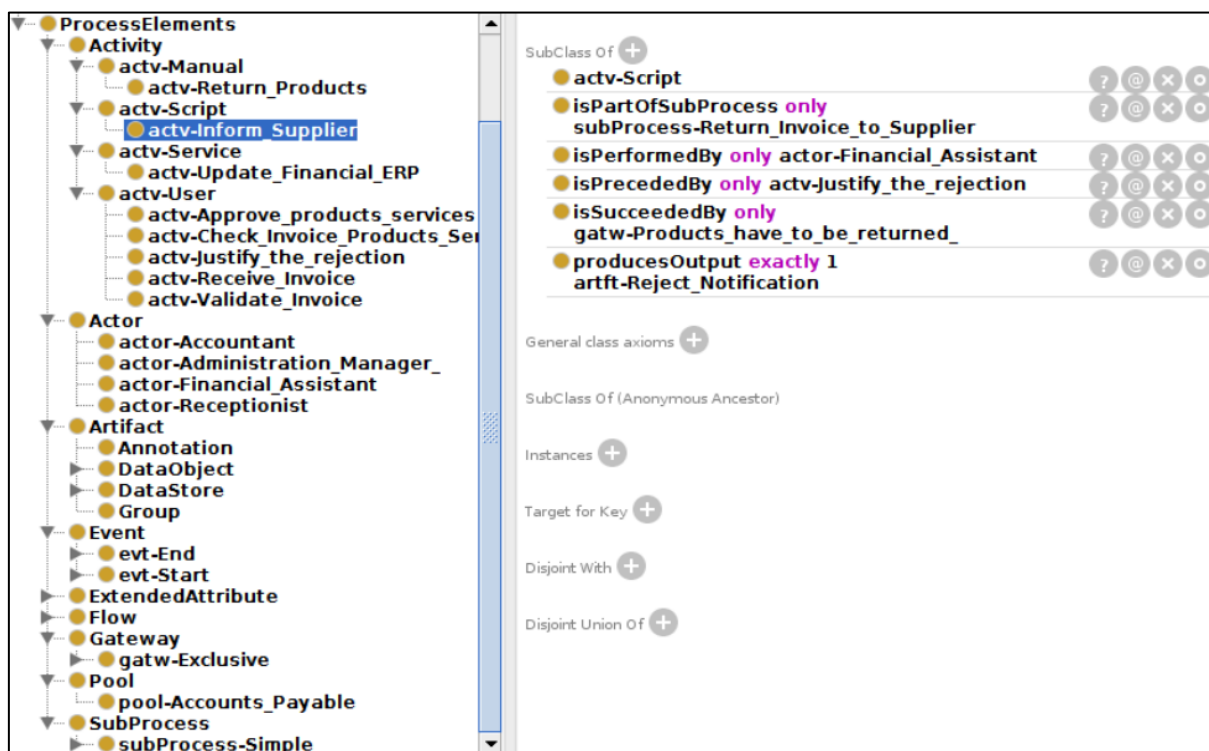
As heurísticas de requisitos da **Tabela 11**, quando aplicadas ao modelo, permitiram a identificação dos requisitos de software por meio das atividades (8) e dos *gateways* (4). Para as atividades do tipo “Usuário”, “Serviço” ou “Script”, foram analisados seus atributos estendidos. Esses atributos, quando nomeados como “RNF” (ou “NFR”, em alusão ao termo *Non Functional Requirements*) foram, então, identificados como requisitos não funcionais para a ontologia gerada. Quando apresentaram o nome “REGRA” ou “RULE” (em alusão ao termo em inglês), então foram identificados como regras de negócio. Para identificação dos requisitos funcionais, foi analisada a documentação textual das oito atividades do modelo. A documentação textual dos quatro *gateways* foi analisada para a identificação de regras de negócio a partir desses elementos.

Discussão dos resultados

Após a aplicação das heurísticas de negócio e de requisitos, o modelo foi exportado para XPDL, gerando um arquivo em XPDL v2.2. Parte da ontologia gerada é apresentada na **Figura 27**, enquanto a ontologia completa está apresentada no APÊNDICE G. A ontologia gerada apresentou os seguintes elementos:

- 38 classes;
- 245 axiomas;
- 3 regras de negócio;
- 7 requisitos funcionais;
- 4 requisitos não funcionais.

Figura 27 - Parte da ontologia gerada para o modelo “Contas a Pagar”.



Fonte: elaborado pelo autor.

O estudo de caso realizado para o modelo de processos de negócio “Contas a Pagar” evidenciou a aplicabilidade do processo sistemático apresentado neste trabalho em um modelo de processos de negócio real. A ontologia gerada a partir do modelo permitiu identificar melhor as interdependências desse modelo. Exemplo: a relação entre duas atividades que estão entre um *gateway*, indicando que o direcionamento desse fluxo de decisão, após a execução da atividade que o antecede, invocou determinada atividade. Essa interdependência é observada na ontologia por meio dos axiomas definidos à partir da propriedade de objeto “isExecutedWhen[Nome da Atividade]OutputIs”.

A ontologia gerada contemplou todos os elementos e relações do modelo “Contas a Pagar”. As heurísticas de requisitos possibilitaram a identificação de requisitos de software, representados pelas seguintes propriedades de dados booleanas associadas aos conceitos da ontologia: “is_business_rule”, “is_functional_requirement” e “is_non_functional_requirement”.

Na **Tabela 19**, apresenta-se uma comparação entre o número de objetos de fluxo do modelo “Contas a Pagar” e o número de classes da ontologia gerada. Na **Tabela 20**, a comparação é feita entre o número de objetos de conexão do modelo e o número de axiomas gerados a partir das propriedades de objeto da ontologia.

Tabela 19 - Modelo “Contas a pagar”: número de fluxos *versus* classes geradas.

Elemento BPMN (objetos de fluxo)	Quantidade no modelo	Classe da ontologia	Quantidade mapeada para a ontologia
Processo	1	<i>Process</i>	1
Piscina	1	<i>Pool</i>	1
Participante	4	<i>Actor</i>	4
Subprocesso	1	<i>SubProcess</i>	1
Atividade	8	<i>Activity</i>	8
Evento	5	<i>Event</i>	5
<i>Gateway</i>	4	<i>Gateway</i>	4
Depósito de dados	2	<i>DataStore</i>	2
Objeto de dados	4	<i>DataObject</i>	4
Anotação	0	<i>Annotation</i>	0
Atributo estendido	4	<i>ExtendedAttribute</i>	4

Fonte: elaborado pelo autor.

Tabela 20 - Modelo “Contas a pagar”: elementos de conexão *versus* axiomas gerados.

Elemento BPMN (objetos de conexão)	Quantidade no modelo	Propriedade de objeto da ontologia	Quantidade de axiomas com a propriedade
Fluxo de sequência comum	10	- <i>isSucceededBy</i> - <i>isPrecededBy</i>	28
Fluxo de mensagem	0	<i>communicatesWith</i>	0
Fluxo sequência após <i>gateway</i> exclusivo	8	<i>isExecutedWhen</i> [Nome da Atividade] <i>Outputs</i>	4
Fluxo sequência após <i>gateway</i> paralelo	0	<i>isExecutedAfterParallelExecutionOf</i>	0
Fluxo sequência após <i>gateway</i> inclusivo	0	<i>isExecutedAfterInclusiveExecution Of</i>	0
Fluxo sequência após um evento	0	<i>isActivatedWhenEventIsTriggered</i>	2
Associação entre um artefato e uma atividade, subprocesso, evento ou <i>gateway</i>	6	- <i>usesInput</i> - <i>producesInput</i> - <i>isAnnotatedBy</i>	6
Performances (associação de atividade com participante)	9	- <i>isPerformedBy</i> - <i>isExecutorOfActivity</i>	18

Fonte: elaborado pelo autor.

5.4 ESTUDO DE CASO 2: MODELO “PROCESSO DE VIAGEM”

Este estudo de caso considerou o modelo de processos de negócio “Processo de Viagem” de uma organização, extraído do site da empresa *iProcess*: <http://iprocess.com.br>. O modelo é composto por atividades relacionadas aos procedimentos necessários para que uma empresa tenha controle sobre uma viagem de um funcionário. Essas atividades incluem a solicitação de viagem, cotação e compra de passagens, prestação de contas após a pagamentos referentes à viagem, entre outras.

Esse processo conta com cinco raias, referentes aos seguintes setores: Assistente Financeiro, Assistente de Recursos Humanos (RH), Aprovador, Secretária e Solicitante. Esse modelo completo inclui três subprocessos de negócio: (1) “Aprovação de viagem”; (2) “Substituição de viajante”; (3) “Interrupção do processo”. O subprocesso “Aprovação de Viagem” está apresentado na **Figura 28**, com três raias: Gestor indicado, Aprovador e Solicitante. Os Subprocessos “Substituição de viajante” e “Interrupção do processo” possuem uma raia cada um. O modelo completo está disponível no APÊNDICE F.

Ao contrário dos outros dois subprocessos, o subprocesso “Aprovação de viagem” é chamado como um subfluxo a partir do processo principal. Dessa forma, ao exportar o modelo “Processo de Viagem” para o formato XPD (Etapa 1 do processo sistemático), foram gerados dois arquivos XPD: um para o processo principal com os outros dois subprocessos e outro para o subprocesso “Aprovação de viagem”.

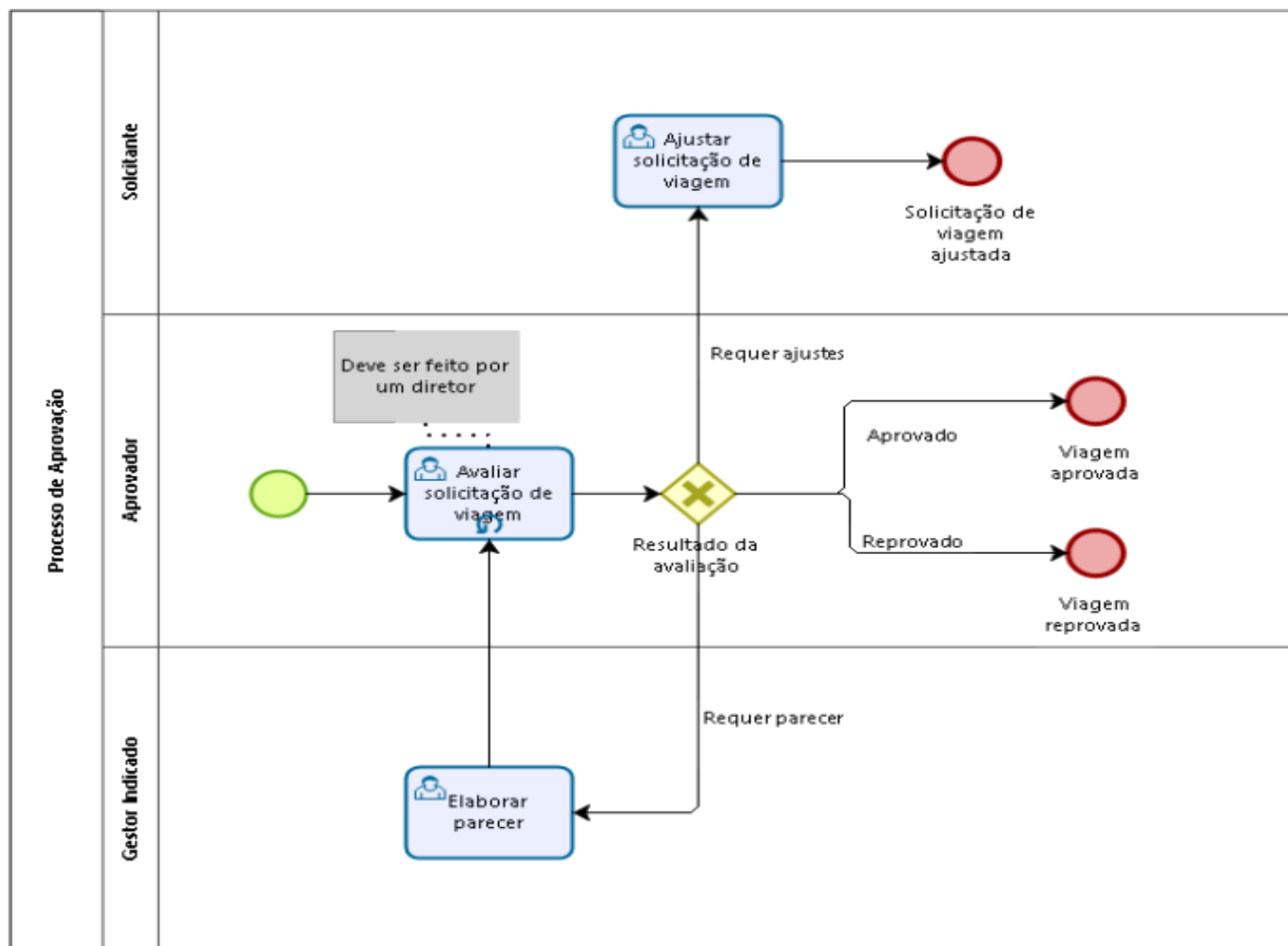
Ao todo, o modelo “Processo de Viagem” conta com os seguintes elementos:

- 8 piscinas;
- 6 participantes;
- 19 atividades (1 sem tipo definido, 1 do tipo “Script”, 3 do tipo “Serviço”, 7 do tipo “Simples” e 7 do tipo “Usuário”);
- 23 eventos (5 do tipo “Início”, 8 do tipo “Intermediário” e 10 do tipo “Fim”);
- 11 *gateways* (7 do tipo “Exclusivo”, 2 do tipo “Inclusivo” e 2 do tipo “Paralelo”);
- 3 objetos de dados;
- 1 repositórios de dados;
- 1 anotação;
- nenhum atributo estendido.

Análise da conformidade com os critérios definidos

Dentre os demais modelos de processo de negócio apresentados nos estudos de caso neste capítulo, o modelo “Processo de Viagem” foi o que mais requereu modificações para ficar em conformidade com os doze critérios definidos na *Seção 4.1*. Nesse modelo foram necessários os seguintes ajustes, com os respectivos critérios indicados:

Figura 28 - Subprocesso “Aprovação de Viagem” do modelo “Processo de Viagem”.



Fonte: <http://iprocess.com.br>.

- **C2:** as atividades do subprocesso “Aprovação de Viagem” foram associadas a um determinado ator para execução de cada uma das atividades;
- **C3:** alguns elementos, como, por exemplo, o *gateway* após a atividade “Avaliar prestação de contas”, pertencente ao processo principal, foram ajustados com definição de nome e documentação;
- **C6:** alguns elementos, como, por exemplo, o evento “Retomada de Cotação”, apresentaram o mesmo nome, mesmo sendo de tipos diferentes. Dessa forma, o nome desses elementos foi ajustado;
- **C7:** algumas atividades, como, por exemplo, a atividade “Ajustar solicitação de viagem”, pertencente ao subprocesso “Aprovação de Viagem”, foram associadas a um determinado ator para execução dessas tarefas;
- **C10:** alguns fluxos de sequência após *gateways*, como, por exemplo, o fluxo de sequência após o *gateway* que sucede a atividade “Realizar aquisição da viagem”, foram devidamente nomeados.

Aplicação das heurísticas de negócio

Como já mencionado anteriormente, o modelo “Processo de Viagem” foi o que mais necessitou de ajustes para satisfazer os critérios C1 a C12, da Seção 4.1, visando o mapeamento para uma ontologia.

Esse modelo é considerado de alta complexidade alta por apresentar muitos elementos e três subprocessos. Assim, a aplicação das heurísticas de negócio apresentadas na **Tabela 10** requereu muitos ajustes no modelo. Apesar desse esforço, os resultados foram significativos para a geração da ontologia.

Os ajustes foram efetuados na documentação textual dos elementos. Como exemplo, pode-se mencionar o *gateway* após a atividade “Reservar orçamento” do processo principal, pois ele não possuía nenhum atributo textual. Em relação à parte gráfica, não foram realizados ajustes no modelo.

Aplicação das heurísticas de requisitos

Considerando as heurísticas de requisitos da **Tabela 11**, a aplicação das heurísticas HR4, HR5, HR6 e HR8 possibilitou a identificação de requisitos funcionais, a partir das atividades e dos *gateways* existentes no modelo “Processo de Viagem”. Como esse modelo conta com sete atividades do tipo “Usuário”, foi possível aplicar as heurísticas HR4 e HR6,

analisando os atributos estendidos dessas atividades, a fim de identificar requisitos não funcionais e regras de negócio a partir dessas atividades. A heurística HR8, por sua vez, foi aplicada para a análise da documentação textual dessas atividades, visando a identificação de requisitos funcionais. Em relação aos onze *gateways* existentes no modelo, foi aplicada a heurística HR5, para a análise da documentação textual e identificação de regras de negócio a partir desses *gateways*.

Discussão dos resultados

Após a aplicação das heurísticas de negócio e de requisitos, o modelo foi exportado para XPDL, gerando dois arquivos em XPDL v2.2 (APÊNDICE G). Ambos os arquivos foram aplicados à ferramenta P2MONTTO para a geração da ontologia. A ontologia gerada completa encontra-se também no APÊNDICE G. A ontologia gerada apresentou os seguintes elementos:

- 137 classes;
- 414 axiomas;
- 0 regras de negócio;
- 11 requisitos funcionais;
- 0 requisitos não funcionais.

A ontologia gerada contemplou todos os elementos do modelo “Processo de Viagem”, com apenas uma observação a ser destacada. O processo principal desse modelo comunica-se com piscinas que indicam organizações externas à envolvida com o processo de viagem. Essas piscinas são: Processo RH, Agência de viagens e Banco. Embora haja comunicação com as piscinas em questão, as mesmas não foram mapeadas para a ontologia como elementos da classe *Pool*. Trata-se de uma restrição adotada na Etapa 2 do processo sistemático apresentado na Seção 4.3. A restrição é que organizações externas ao processo principal não devem ser mapeadas para a ontologia.

As heurísticas de requisitos possibilitaram a identificação de requisitos de software, representados pelas seguintes propriedades de dados booleanas associadas aos conceitos da ontologia: “isBusinessRule”, “isFunctionRequirement” e “isNonFunctionalRequirement”. Observa-se que para o caso desse modelo de processos de negócio foram obtidos apenas requisitos funcionais.

Na **Tabela 21**, apresenta-se uma comparação entre o número de objetos de fluxo do modelo “Processo de Viagem” e o número de classes da ontologia. Na **Tabela 22**, a comparação é feita entre o número de objetos de conexão do modelo e o número de axiomas gerados a partir das propriedades de objeto da ontologia.

Tabela 21 - Modelo “Processo de Viagem”: número de fluxos *versus* classes geradas.

Elemento BPMN (objetos de fluxo)	Quantidade no modelo	Classe da ontologia	Quantidade mapeada para a ontologia
Processo	1	<i>Process</i>	1
Piscina	8	<i>Pool</i>	5
Participante	6	<i>Actor</i>	6
Subprocesso	3	<i>SubProcess</i>	3
Atividade	19	<i>Activity</i>	19
Evento	23	<i>Event</i>	23
<i>Gateway</i>	11	<i>Gateway</i>	11
Depósito de dados	1	<i>DataStore</i>	1
Objeto de dados	3	<i>DataObject</i>	3
Anotação	1	<i>Annotation</i>	1
Atributo estendido	0	<i>ExtendedAttribute</i>	0

Fonte: elaborado pelo autor.

Tabela 22 - Modelo “Processo de Viagem”: elementos de conexão *versus* axiomas gerados.

Elemento BPMN (objetos de conexão)	Quantidade no modelo	Propriedade de objeto da ontologia	Quantidade de axiomas com a propriedade
Fluxo de sequência comum	19	- <i>isSucceededBy</i> - <i>isPrecededBy</i>	38
Fluxo de mensagem	3	<i>communicatesWith</i>	2
Fluxo sequência após <i>gateway</i> exclusivo	14	<i>isExecutedWhen</i> [Nome da Atividade] <i>Outputs</i>	2
Fluxo sequência após <i>gateway</i> paralelo	0	<i>isExecutedAfterParallelExecution Of</i>	0
Fluxo sequência após <i>gateway</i> inclusivo	1	<i>isExecutedAfterInclusiveExecutio nOf</i>	1
Fluxo sequência após um evento	10	<i>isActivatedWhenEventIsTriggered</i>	10
Associação entre um artefato e uma atividade, subprocesso, evento ou <i>gateway</i>	5	- <i>usesInput</i> - <i>producesInput</i> - <i>isAnnotatedBy</i>	5
Performances (associação de atividade com participante)	19	- <i>isPerformedBy</i> - <i>isExecutorOfActivity</i>	38

Fonte: elaborado pelo autor.

5.5 ESTUDO DE CASO 3: MODELO “COMPRA DE LIVROS”

Este estudo de caso considerou o modelo de processos de negócio “Compra de Livros” da Biblioteca da Unesp, no Câmpus de Rio Claro, desenvolvido por Pucci (2016). Esse processo envolve o cadastro dos pedidos de livros em um sistema de software específico da Biblioteca (sistema Aleph), assim como o controle desses pedidos. Também envolve a verificação da existência dos livros no acervo antes de realizar o pedido, controla o recebimento dos livros, entre outros procedimentos relacionados.

O processo “Compra de Livros” conta com seis raias, referentes aos seguintes setores: STATI (Seção Técnica de Aquisição e Tratamento da Informação), Supervisora técnica, Comissão avaliadora de pedido, Diretora, Assistente de suporte e Bibliotecários. O processo principal comunica-se também com outras duas piscinas: Instituto de Biociências (IB) e Docentes.

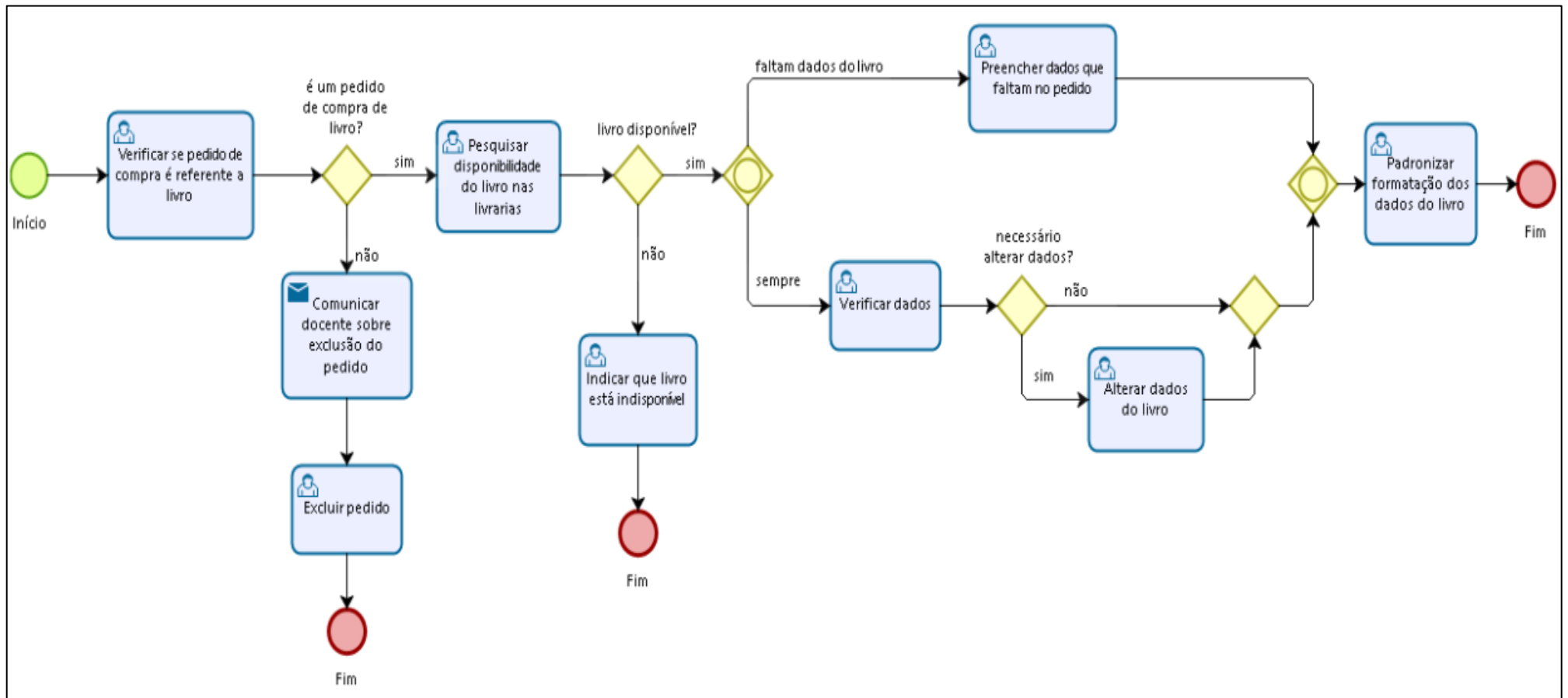
Esse modelo inclui nove subprocessos: (1) Atualizar dados dos pedidos ainda não comprados; (2) Completar dados dos novos pedidos; (3) Verificar existência do livro no acervo; (4) Avaliar pedido; (5) Cadastrar pedidos no Sistema Aleph; (6) Receber livros; (7) Entrar em contato com livraria; (8) Completar item e colocar etiquetas; (9) Catalogar livros. A **Figura 29** apresenta o subprocesso “Completar dados dos novos pedidos”. O modelo completo está disponível no APÊNDICE F.

Embora o modelo “Compra de Livros” apresente nove subprocessos, nenhum deles é invocado a partir do processo principal. Dessa forma, apenas um arquivo XPDL foi gerado após a exportação do modelo (Etapa 1 do processo sistemático).

Ao todo, o modelo “Compra de Livros” conta com os seguintes elementos:

- 3 piscinas;
- 4 participantes;
- 90 atividades (2 do tipo “Serviço”, 3 do tipo “Simples”, 46 do tipo “Usuário”, 10 do tipo “Envio”, 1 do tipo “Recebimento”, 2 do tipo “Regra de Negócio” e 26 do tipo “Manual”);
- 32 eventos (11 do tipo “Início”, 8 do tipo “Intermediário” e 13 do tipo “Fim”);
- 32 *gateways* (17 do tipo “Exclusivo”, 7 do tipo “Inclusivo”, 5 do tipo “Paralelo”, 2 do tipo “Baseado em Evento” e 1 do tipo “Complexo”);
- 9 objetos de dados;
- repositórios de dados;
- 2 anotações;
- 1 atributo estendido.

Figura 29 - Subprocesso "Completar dados dos novos pedidos" do modelo "Compra de Livros".



Fonte: Pucci (2016).

Análise da conformidade com os critérios definidos

Antes da aplicação das heurísticas de negócios, foi verificado se o modelo requeria ajustes para ficar em conformidade com os critérios C1 a C12, da *Seção 4.1*, visando o mapeamento para uma ontologia. É importante observar que o modelo “Compra de Livros” necessitou de poucos ajustes. Isso ocorreu porque esse modelo foi criado com base em boas práticas de modelagem. Além disso, o modelo foi ajustado manualmente pela autora para que houvesse melhoria na documentação textual de seus elementos (PUCCI, 2016).

Esse modelo é considerado de alta complexidade por apresentar muitos elementos e três subprocessos. Contudo, como mencionado anteriormente, precisou de poucos ajustes por meio dos critérios C1 a C12. Um deles foi a modificação da documentação textual referente ao nome repetido de alguns elementos, como é o caso da atividade “Adicionar no Sistema 3S observação sobre fracasso de compra”. Havia duas atividades com esse nome: uma no processo principal e outra no subprocesso “Entrar em contato com livraria”. Para esse caso e outros similares, onde elementos do mesmo tipo apresentam o mesmo nome, foi aplicado o critério C6.

Outro exemplo de ajuste foi quanto aos executores das atividades do modelo “Compra de Livros”. As atividades dos subprocessos não apresentaram inicialmente executores associados a elas. Por exemplo, a atividade “Excluir pedido” do subprocesso “Completar dados dos novos pedidos” não apresentava um dos quatro participantes do processo como seu executor (“Bibliotecários”, “Diretora”, “Assistente de Suporte” e “Supervisora técnica”). Dessa forma, o critério C7 foi aplicado para atividades sem esse vínculo com algum participante do processo.

Aplicação das heurísticas de negócio

Apesar da complexidade do modelo, a aplicação das heurísticas de negócio apresentadas na **Tabela 10** requereu poucos ajustes no modelo. As heurísticas HNd2 e HNd7, por exemplo, não precisaram ser aplicadas, pois os artefatos e repositórios de dados encontrados no modelo já se encontram relacionados às atividades que manipulam suas informações.

Aplicação das heurísticas de requisitos

Considerando as heurísticas de requisitos da **Tabela 11**, a aplicação das heurísticas. a aplicação das heurísticas de requisitos HR4, HR5, H6 e HR8 possibilitou a identificação de requisitos funcionais a partir das atividades e dos *gateways* existentes no modelo de

processos de negócio “Compra de Livros”.

Discussão dos resultados

Após a aplicação das heurísticas de negócio e de requisitos, o modelo foi exportado para XPDL, gerando um arquivo em XPDL v2.2 (APÊNDICE F). O arquivo foi aplicado à ferramenta P2MONTTO para a geração da ontologia. A ontologia gerada completa encontra-se também no APÊNDICE G. A ontologia gerada apresentou os seguintes elementos:

- 256 classes;
- 1097 axiomas;
- 0 regras de negócio;
- 48 requisitos funcionais;
- 0 requisitos não funcionais.

A ontologia gerada contemplou todos os elementos do modelo “Compra de Livros”. A aplicação dos critérios mencionados, C6 e C7, foi essencial para evitar a sobreposição de classes, no caso dos elementos de mesmo tipo e mesmo nome encontrados no modelo original. Essa sobreposição causaria outras inconsistências, como a definição incorreta dos axiomas para essas classes. A associação das atividades pertencentes a subprocessos com pelo menos um participante como executor, possibilitou a definição de novos axiomas, que relacionam as classes *Activity* e *Actor*.

Para o modelo de processos de negócio “Compra de Livros” foram obtidos apenas requisitos funcionais, assim como no modelo “Processo de Viagem”, na seção anterior. As classes da ontologia que representam esses requisitos funcionais apresentam a propriedade de dados booleana “isFunctionalRequirement” definida como verdadeira.

Na **Tabela 23**, apresenta-se uma comparação entre o número de objetos de fluxo do modelo “Compra de Livros” e o número de classes da ontologia. Na **Tabela 24**, a comparação é feita entre o número de objetos de conexão do modelo e o número de axiomas gerados a partir das propriedades de objeto da ontologia.

5.6 AVALIAÇÃO GERAL DOS RESULTADOS DOS ESTUDOS DE CASO

Considerando os estudos de caso apresentados nas Seções 5.3, 5.4 e 5.5, foi possível constatar que o processo sistemático, com o apoio do sistema PM2ONTTO, possibilitou a geração de ontologias representativas para os modelos de processos de negócio. A **Tabela 25** apresenta uma comparação dos elementos das três ontologias geradas.

Tabela 23 - Modelo “Compra de Livros”: número de fluxos *versus* classes geradas.

Elemento BPMN (objetos de fluxo)	Quantidade no modelo	Classe da ontologia	Quantidade mapeada para a ontologia
Processo	1	<i>Process</i>	1
Piscina	3	<i>Pool</i>	3
Participante	4	<i>Actor</i>	4
Subprocesso	9	<i>SubProcess</i>	9
Atividade	90	<i>Activity</i>	90
Evento	32	<i>Event</i>	32
<i>Gateway</i>	32	<i>Gateway</i>	32
Depósito de dados	5	<i>DataStore</i>	5
Objeto de dados	9	<i>DataObject</i>	9
Anotação	2	<i>Annotation</i>	2
Atributo estendido	1	<i>ExtendedAttribute</i>	1

Fonte: elaborado pelo autor.

Tabela 24 - Modelo “Compra de Livros”: elementos de conexão *versus* axiomas gerados.

Elemento BPMN (objetos de conexão)	Quantidade no modelo	Propriedade de objeto da ontologia	Quantidade de axiomas com a propriedade
Fluxo de sequência comum	105	- <i>isSucceededBy</i> - <i>isPrecededBy</i>	38
Fluxo de mensagem	10	<i>communicatesWith</i>	10
Fluxo sequência após <i>gateway</i> exclusivo	29	<i>isExecutedWhen</i> [Nome da Atividade] <i>OutputIs</i>	9
Fluxo sequência após <i>gateway</i> paralelo	4	<i>isExecutedAfterParallelExecutionOf</i>	0
Fluxo sequência após <i>gateway</i> inclusivo	12	<i>isExecutedAfterInclusiveExecutionOf</i>	1
Fluxo sequência após um evento	20	<i>isActivatedWhenEventIsTriggered</i>	20
Associação entre um artefato e uma atividade, subprocesso, evento ou <i>gateway</i>	38	- <i>usesInput</i> - <i>producesInput</i> - <i>isAnnotatedBy</i>	38
Performances (associação de atividade com participante)	99	- <i>isPerformedBy</i> - <i>isExecutorOfActivity</i>	103

Fonte: elaborado pelo autor.

Tabela 25 - Visão geral dos resultados dos três estudos de caso.

Modelo de processos de negócio	Classes	Axiomas	Regras de negócio	Requisitos funcionais	Requisitos não funcionais
Contas a Pagar	38	245	3	7	4
Processo de Viagem	137	414	0	11	0
Compra de Livros	256	1097	0	48	0

Fonte: elaborado pelo autor.

A aplicação dos critérios C1 a C12 indicou a necessidade de ajustes em todos os modelos utilizados. Esses ajustes permitiram a identificação de mais axiomas para as ontologias geradas, assim como evitaram inconsistências nessas ontologias (ex.: repetição de nome de classes). Os ajustes também possibilitaram o mapeamento adequado dos elementos de BPMN, com a documentação associada, para as ontologias.

A aplicação das heurísticas de negócio e de requisitos definidas por Nogueira (2017) também auxiliou na melhoria de documentação textual e na parte gráfica, com a inclusão de novos elementos conforme a necessidade. Ressalta-se, porém, a grande contribuição das heurísticas de requisitos para a extração de requisitos de software. Essas heurísticas permitiram a marcação dos conceitos da ontologia que dizem respeito aos requisitos funcionais, não funcionais e regras de negócio.

Entre os modelos de processos de negócio utilizados para os estudos de caso, o que mais permitiu a identificação de requisitos foi o modelo “Contas a Pagar”. Por meio dos outros modelos, “Processo de Viagem” e “Compra de Livros”, foi possível verificar apenas requisitos funcionais. Esses requisitos puderam ser identificados com base no tipo das atividades e *gateways* dos modelos de processos de negócio utilizados. Uma das razões para a ausência de requisitos não funcionais e regras de negócio nos modelos “Processo de Viagem” e “Compra de Livros” está na existência de poucos atributos estendidos definidos para esses modelos. Os atributos estendidos tendem a apresentar documentação textual referente a um requisito de software.

A ontologia que apresentou maior expressividade em relação à quantidade de conceitos e axiomas foi a gerada para o modelo de processos de negócio “Compra de Livros”. Isso ocorreu devido à complexidade desse modelo, formado por nove subprocessos. Por meio dos ajustes necessários após a aplicação dos doze critérios de mapeamento foi possível obter novos axiomas para a ontologia desse modelo, como por exemplo a associação de atividades dos subprocessos com os seus executores.

6

CONCLUSÃO E CONSIDERAÇÕES FINAIS

Este trabalho apresentou um processo sistemático, formado por três etapas, para a geração automática de ontologias a partir de modelos de processos de negócio na notação BPMN v2.0. Alguns trabalhos encontrados na literatura, apresentados no *Capítulo 3*, também são voltados à construção de ontologias a partir de modelos de processos de negócio. Entretanto, esses trabalhos não buscam explorar uma quantidade relevante de elementos de BPMN e de suas interdependências no modelo de processos de negócio. Também é importante destacar que alguns desses trabalhos não apresentam um mecanismo automatizado para a geração das ontologias.

A forma alternativa de representação de um modelo de processos de negócio por meio de uma ontologia traz inúmeros benefícios. As ontologias melhoram a visualização dos elementos e da documentação dos modelos, bem como das interdependências contidas nesses modelos. Por meio das ontologias, podem ser feitas inferências e consultas sobre os modelos de processos de negócio. As ontologias possibilitam a representação do conhecimento que se encontra implícito nos modelos de processos de negócio. Além disso, a legibilidade de modelos complexos (incluindo subprocessos e documentação) é beneficiada. Isso porque a leitura das ontologias não é afetada à medida em que elas se tornam mais extensas, ao contrário dos modelos de processos de negócio. A legibilidade das ontologias está diretamente relacionada à categorização das classes pertencentes às ontologias, melhorando a forma de identificação e a navegabilidade entre esses conceitos.

As ontologias também possibilitam, de certa forma, que os modelos de processos de negócio sejam utilizados por diferentes sistemas de software. As ontologias são estruturas manipuláveis por máquina e podem ser visualizados de diversas formas.

A navegação de um modelo de processos de negócio por meio de uma ontologia melhora o entendimento das atividades e dos recursos existentes nesse modelo. É possível inserir conhecimento nas ontologias e compartilhá-lo entre as equipes de negócio e demais equipes de uma organização, inclusive as de Tecnologia da Informação (TI). Assim, as visões das equipes de negócio e de TI podem ser mais alinhadas ao domínio de interesse.

Para ontologias mais completas, recomenda-se que os modelos apresentem

documentação textual coerente e sejam construídos utilizando boas práticas de modelagem. Para isso, visando a aplicação do processo sistemático apresentado, é necessário que os modelos de processos de negócio satisfaçam doze critérios (C1 a C12) definidos na Seção 4.1, baseados em boas práticas de modelagem. Além disso, conforme apresentado na Seção 4.2, recomenda-se que os modelos estejam em conformidade com determinadas heurísticas de negócio e de requisitos definidas por Nogueira (2017). Em casos de não conformidade com os critérios ou heurísticas, os ajustes requerem esforço manual. No entanto, esses ajustes agregam valor ao conhecimento dos processos de negócio de um ambiente organizacional e contribuem para a geração de uma ontologia mais representativa em termo de classes, propriedades e axiomas.

Para automatizar o processo sistemático apresentado neste trabalho, foi desenvolvida a ferramenta P2MONTTO (*Process Model To Ontology*), que toma como entrada os arquivos XPDL para os quais os modelos de processos de negócio foram exportados. Para cada modelo de processos de negócio é então gerada uma ontologia representativa do modelo, descrita na linguagem OWL.

Adicionalmente, a ferramenta conta com um sistema de consultas definidas na linguagem SPARQL. Essas consultas estão direcionadas às ontologias geradas e são divididas em dois grupos: Básicas e Avançadas. As consultas Básicas retornam informações como identificação, nome, descrição, documentação e tipo, referentes a uma ou mais classes de uma ontologia. As consultas Avançadas, por sua vez, estão direcionadas aos axiomas existentes nas ontologias, retornando informações de classes presentes em determinadas relações das ontologias.

Para o desenvolvimento do sistema PM2ONTO foi utilizado o *framework* de licença gratuita Apache JENA. Entre os recursos disponíveis nesse *framework* está a geração e manipulação de ontologias descritas em OWL. O *framework* também apresenta recursos para a realização de consultas em SPARQL sobre ontologias. Como apoio à geração das ontologias, foi definido também um metamodelo (Meta2Onto), que apoia o mapeamento dos elementos de BPMN para as ontologias.

Neste trabalho é mencionada a ferramenta *Protégé* para a visualização e validação das ontologias geradas, mas outras ferramentas podem ser utilizadas para essa finalidade. A ferramenta *Protégé* apresenta recursos para a realização de inferências em ontologias, permitindo identificar relações implícitas nessas ontologias.

De modo geral, com a geração de uma ontologia associada a um modelo de processos de negócios, este trabalho contribui, primeiramente, para melhorar as condições de navegabilidade entre as atividades e recursos do modelo, evidenciando a documentação

textual e as interdependências entre seus elementos. As inferências que podem ser utilizadas nas ontologias permitem explicitar conhecimento antes implícito nos modelos. Além disso, novas informações podem ser inseridas nas ontologias, aumentando o grau de documentação e compreensão dos modelos. É importante mencionar que as ontologias possibilitam consultas abrangentes sobre os modelos de processos de negócio, que são relevantes às equipes de negócio e de TI e que não poderiam, muitas vezes, serem respondidas apenas pela representação do modelo em BPMN. Ainda, as ferramentas de modelagem de processos de negócio, por si só, não apresentam recursos para consultas envolvendo elementos e interdependências no modelo; já com as ferramentas associadas às ontologias, isso é possível. Por fim, observa-se que todo o conteúdo pode ser compartilhado entre as equipes de negócio e de TI, apoiando a comunicação entre essas duas áreas.

Trabalhos futuros

Este trabalho considerou elementos de BPMN v2.0 comumente utilizados na modelagem de processos de negócio, assim como boa parte dos atributos desses elementos. Contudo, é interessante que se amplie o conjunto de elementos da notação BPMN e, consequentemente, de atributos. Então, o processo de geração da ontologia deverá ser ajustado para definir também novas relações, definidas pelas propriedades de objetos e axiomas. Por outro lado, é possível definir novos axiomas para as ontologias. Para isso, deve-se analisar os elementos dos modelos para verificar novas e relevantes interdependências. Após essa análise de viabilidade da interdependência, então a nova relação é mapeada para o metamodelo Meta2Onto.

Ainda em relação a melhorias no processo de geração de ontologias, pode-se verificar a viabilidade em utilizar classes disponíveis pela comunidade “Schema.org”, no seguinte endereço: <http://schema.org/docs/full.html>. As classes servem para a descrição de conceitos de diferentes áreas. Essas classes estão estruturadas hierarquicamente, de maneira que existem superclasses, descrevendo conceitos genéricos, e subclasses, descrevendo conceitos mais específicos de uma área de interesse. Assim, como trabalho futuro, serão adotadas algumas dessas classes para a ontologia de processos gerada pelo processo sistemático, com o objetivo de padronizar ainda mais os conceitos que compõem a ontologia.

Atualmente, os requisitos funcionais e não funcionais, assim como as regras de negócio, são obtidos por meio de consultas sobre as ontologias. Como trabalho futuro, uma nova etapa pode ser inserida no processo sistemático, visando criar ontologias de domínio para representar especificamente os requisitos de software. A ideia é utilizar as ontologias geradas a partir dos modelos de processos de negócio e então compor uma outra ontologia,

de domínio, com base em um mapeamento dos conceitos e relações da ontologia base.

Uma outra linha de trabalho relevante para as organizações que utilizam modelos de processos de negócio, é incluir a funcionalidade de integração de ontologias na ferramenta PM2ONTO. Isso poderia permitir maior variedade de consultas e observação de mais interdependências entre os modelos de processos de negócio. Essa integração poderia utilizar técnicas de PLN e de mineração de textos nas propriedades dos conceitos das ontologias geradas. Assim, uma nova ontologia, de domínio, poderia ser construída, contemplando os elementos comuns às ontologias integradas.

O processo sistemático não prevê a atualização de uma ontologia após o reprocessamento de um modelo de processos de negócio que sofreu alterações. Assim, todos os modelos de processos de negócio aplicados na ferramenta PM2ONTO geram uma nova ontologia. Futuramente, pode ser realizada uma alteração na etapa de geração de ontologias do processo sistemático. Essa alteração realizaria a verificação da existência de uma ontologia para o modelo de processos de negócio reprocessado e todas as modificações feitas no modelo seriam então repassadas à ontologia, sem que haja a necessidade em criar uma nova ontologia. As novas classes e relações referentes às alterações no modelo de processos de negócio constariam na ontologia atualizada.

O sistema de consultas na ferramenta PM2ONTO também pode ser melhorado com a utilização do Apache Jena Fuseki, um recurso presente no *framework* Apache JENA, que disponibiliza um protocolo HTTP (*Hypertext Transfer Protocol*), permitindo a execução de consultas em SPARQL para dados de ontologias. Dessa forma, o usuário não ficaria mais limitado às consultas manuais pré-definidas nesse trabalho, podendo criar suas consultas conforme a necessidade. Além disso, não haveria mais a necessidade em armazenar as ontologias em um banco de dados, uma vez que esse armazenamento existe com o objetivo de recuperar as ontologias geradas para a realização de consultas.

Em outra direção, é importante avaliar a usabilidade das ontologias e como profissionais de negócios e de TI as consideram em termos de recurso complementar aos modelos de processos de negócio. Para a avaliação da usabilidade das ontologias, pode ser utilizada a metodologia proposta por Pizzoleto e Oliveira (2016). Esses autores definem uma sistemática para avaliação de ontologias empresariais, baseada em técnicas de testes de usabilidade de software. A intenção é verificar, com a participação de seres humanos, a navegabilidade, facilidade/flexibilidade de uso e apreensibilidade das ontologias, assim como detectar possíveis inconsistências. Dessa forma, o processo de validação das ontologias geradas neste trabalho seria melhorado e essas ontologias poderiam ser ajustadas de acordo com os resultados dos testes. De modo similar, testes de usabilidade poderiam ser aplicados

também sobre a ferramenta PM2ONTO, para avaliar a interação com usuários em potencial do sistema. Esses testes poderiam ser baseados em heurísticas ou com a participação de usuários.

REFERÊNCIAS

ABPMP - ASSOCIATION OF BUSINESS PROCESS MANAGEMENT PROFESSIONALS. **BPM CBOK V3.0**: Guia para o Gerenciamento de Processos de Negócio – Corpo Comum de Conhecimento. 1. ed. São Paulo: ABPMP Brasil, 2013.

AFONSO, R., A.; PEREIRA, C., F. MaTUTO: adaptação de metodologia de aprendizagem baseada em problemas aplicada ao ensino de ontologias. **AtoZ: novas práticas em informação e conhecimento**, Curitiba, v. 2, n. 1, p. 34-43, 2013. Disponível em: <<http://www.atoz.ufpr.br>>. Acesso em: 11 mar. 2018.

ALMEIDA, M. B.; BAX, M. P. Uma visão geral sobre ontologias: pesquisa sobre definições, tipos, aplicações, métodos de avaliação e de construção. **Ciência da Informação**, Brasília, v. 32, n. 3, p. 7-20, 2003.

ASF - APACHE SOFTWARE FOUNDATION. **Apache Jena - a free and open source java framework for building Semantic Web and Linked Data applications**, 2017. Disponível em: <<https://jena.apache.org>>. Acesso em: 10 jul. 2017.

ARENAS, M.; PÉREZ, J. Querying semantic web data with SPARQL. In: ACM SIGMOD-SIGACT-SIGART SYMPOSIUM ON PRINCIPLES OF DATABASE SYSTEMS, 2011, Athens. **Proceedings...** Athens, 2011, p. 305-316.

BOC GROUP. **Business Process Management with Adonis**, 2013. Disponível em: <<https://uk.boc-group.com/adonis/>>. Acesso em: 13 dez. 2016.

BORGER, E. Approaches to modeling business processes: a critical analysis of BPMN, workflow patterns and YAWL. **Software & Systems Modeling**, v. 3, p. 305-318, 2012.

BRAUN, R.; ESSWEIN, W. Semantics in the Context of BPMN Extensions – State of Affairs and Research Challenges. In: MULTIKONFERENZ WIRTSCHAFTSINFORMATIK, 2016, Ilmenau. **Proceedings...** Ilmenau, 2016, p. 1-12.

CATARINO, M. E.; SOUZA, T. B. A representação descritiva no contexto da web semântica. **Transinformação**, Campinas, v. 24, n. 2, p. 77-90, maio/ago., 2012.

CHINOSI, M.; TROMBETTA, A. BPMN: An introduction to the standard. **Computer Standards & Interfaces**, Netherlands, v. 34, n. 1, p. 124-134, 2012.

CORREIA, A. Elements of style of BPMN language. **The Computing Research Repository (CoRR)**, Saarbrücken, 2015, p. 1-103.

CORREIA, A.; ABREU, F. B. Enhancing the correctness of BPMN models. In: VARAJÃO, J. E.; CRUZ-CUNHA, M. M.; MARTINHO, R. **Improving organizational effectiveness with Enterprise Information Systems**. Hershey: IGI Global, 2015. v.1, p. 241-261.

CRUZ, E. F.; MACHADO, R. J.; SANTOS, M. Y. From business process models to use case models: A systematic approach. In: ENTERPRISE ENGINEERING WORKING CONFERENCE, 4., 2014, Funchal. **Proceedings...** Funchal, 2014. p. 167-181.

DIAS, F.; MORGADO, G.; OSCAR, P., SILVEIRA, D.; ALENCAR A.; LIMA, P.; SCHMITZ, E. Uma abordagem para a transformação automática do modelo de negócio em modelo de requisitos. In: WORKSHOP EM ENGENHARIA DE REQUISITOS, 6., 2006, Rio de Janeiro. **Anais...** Rio de Janeiro, 2006. p. 51-60.

DIAS, F. **Modelagem de Processos de Negócio**: Diferenças entre diagrama, mapa e modelo, 2014. Disponível em: <<http://blog.iprocess.com.br/2014/02/modelagem-de-processos-de-negocio-diferencas-entre-diagrama-mapa-e-modelo-de-processos/>>. Acesso em: 11 jan. 2017.

DI MARINO, B.; ESPOSITO, A.; NACCHIA, E.; MAISTO, S., A. Semantic annotation of BPMN: current approaches and new methodologies. In: INTERNATIONAL CONFERENCE ON INFORMATION INTEGRATION AND WEB-BASED APPLICATIONS AND SERVICES, 17., 2015, Brussels. **Proceedings...** Brussels, 2015. p. 1-14.

FALBO, R., A. **Aplicações de Ontologias em Engenharia de Software**, 2016. Disponível em: <http://www.inf.ufes.br/~falbo/files/OES/OES3-Aplicacoes_Ontologias_ES.pdf>. Acesso em: 15 dez. 2016.

FALBO, R., A.; BRAGA, C., E., C.; MACHADO, B.; N. Semantic Documentation in Requirements Engineering. In: WORKSHOP ON REQUIREMENTS ENGINEERING (WER), 17., 2014, Pucón. **Proceedings...** Pucón, 2014. p. 1-14.

FANESI, D.; CACCIAGRANO, D. R.; HINKELMANN, K. Semantic Business Process Representation to Enhance the Degree of BPM Mechanization-An Ontology. In: IEEE INTERNATIONAL CONFERENCE ON ENTERPRISE SYSTEMS (ES), 2015, Basel. **Proceedings...** Basel, 2015. p. 21-32.

FILOGRANA, A.; GIUNTA, G.; INDRAFFIA, N.; LOIACONO, L. An integrated approach for modeling business processes using BPMN and XPD L standards. In: KMO CONFERENCE, KNOWLEDGE MANAGEMENT IN ORGANIZATION, 2., 2007, Lecce. **Proceedings...** Lecce, 2007. p. 1-5.

FRANCESCOMARINO, C. Di.; ROSPOCHER, M., G.; VALERIO, A. (2014). The Role of Semantic Annotations in Business Process Modelling. In: IEEE INTERNATIONAL ENTERPRISE DISTRIBUTED OBJECT COMPUTING CONFERENCE, 18., 2014, Ulm. **Proceedings...** Ulm, 2014. p. 181-189.

GÁBOR, A.; KÖ, A. **Corporate Knowledge Discovery and Organizational Learning**: The Role, Importance, and Application of Semantic Business Process Management. Springer International Publishing, 2016. (Knowledge Management and Organizational Learning, v. 2).

GÁBOR, A.; SZABÓ, Z. Semantic Technologies in business process management. In: FATHI, M. Integration of practice-oriented knowledge technology: Trends and perspectives. Berlin: Springer, 2013.

GAGNE, D. **BPMN-CMMN-DMN**: an intro to the triple crown of process improvement standards, 2015. Disponível em: <<http://pt.slideshare.net/dgagne/bpmncmmndmn-an-intro-to-the-triple-crown-of-process-improvement-standards-denis-gagne>>. Acesso em: 22 jun. 2016.

GILES, T. R. **Introdução à filosofia**. São Paulo: EDUSP, 1979.

GOLDBERG, L.; HALLE, B. V. **The OMG Decision Model and Notation Spec (DMN) and The Decision Model (TDM)**, 2015. Disponível em: <<http://www.modernanalyst.com/Resources/Articles/tabid/115/ID/2757/The-OMG-Decision-Model-and-Notation-Spec-DMN-and-The-Decision-Model-TDM.aspx>>. Acesso em: 01 jun. 2016.

GÓMEZ-PÉREZ, A. Towards a framework to verify knowledge sharing technology. **Expert Systems with Applications**, New York, v. 11, n. 4, p. 519-529, 1996.

GÓMEZ-PÉREZ, J., M. **Studies on the Semantic Web**: Acquisition and Understanding of Process Knowledge Using Problem Solving Methods. Amsterdam: IOS Press, 2010.

GRUNINGER, M.; LEE, J. Ontology: applications and design. **Association for Computing Machinery**, New York, v. 45, n. 2, p. 39-41, fev. 2012.

GUARINO, N. **Formal ontology in information systems**. Amsterdam: IOS Press, 1998.

_____. Understanding, building and using ontologies. **International Journal of Human-Computer Studies**, Padova, v. 46, n. 2-3, p. 293-310, 1997.

GUIDO, A.; PANDURINO, A.; PAIANO, R. An Ontological Meta-Model for Business Process Model

and Notation. **International Journal of Business Research and Management (IJBRM)**, v. 7, p. 1-13, 2016.

GUIZZARDI, G. **Desenvolvimento para e com reuso**: um estudo de caso no domínio de vídeo sob demanda. Dissertação (Mestrado) - Universidade Federal do Espírito Santo, 2000.

HALLER, A.; MARMOLOWSKI, M.; OREN, E.; GAALOUL, W. oXPDL: a Process Model Exchange Ontology. **DERI-Digital Enterprise Research Institute**, Galway, p. 1-18, 2007.

HAPPEL, H. J.; SEEDORF, S. Applications of Ontologies in Software Engineering. In: SEMANTIC WEB ENABLED SOFTWARE ENGINEERING, 2., 2006. **Proceedings...** 2006. p. 5-9.

HARMON, P.; WOLF, C. **Business Process Modeling Survey**, 2011. Disponível em: http://www.bptrends.com/reports/Process_Modeling_Survey-Dec_11_FINAL.pdf/. Acesso: 01 out. 2017.

HE GROUP – UNIVERSITY OF MICHIGAN MEDICAL SCHOOL. **Ontobee – Tutorial: How to use Ontobee SPARQL to Query RDF triple store?**, 2013. Disponível em: <http://www.ontobee.org/tutorial/sparql/>>. Acesso em 01 nov. 2017.

HEPP, M.; ROMAN, D. An ontology framework for semantic business process management. In: INTERNATIONALE TAGUNG WIRTSCHAFTSINFORMATIK, 8., 2007, Karlsruhe. **Proceedings...** Karlsruhe, 2007. p. 423-440.

HINKELMANN, K. **CMMN - Case Management and Case Modeling**, 2015. Disponível em: http://knut.hinkelmann.ch/lectures/bpm2015-16/04_CMMN.pdf>. Acesso em 04 jul. 2016.

ISOTANI, S.; BITTENCOURT, I., I. Linguagens de ontologia da Web. In: ISOTANI, S.; BITTENCOURT, I., I. (Eds.). **Dados Abertos Conectados**: Em busca da Web do Conhecimento. Novatec, 2015. Cap. 3, p. 108-123.

JUNG, M.; KIM, H. S.; JO, M. H.; TAK, K. H.; CHA, H. S.; SON, J. H. Mapping from BPMN-formed business processes to XPDL business processes. In: INTERNATIONAL CONFERENCE ON ELECTRONIC BUSINESS, 4., 2004, Beijing. **Proceedings...** Beijing, 2004. p. 422-427.

KIM, T.; PARK, I.; HYUN, S., J.; LEE, D. MiRE4OWL: Mobile Rule Engine for OWL. In: IEEE COMPUTER SOFTWARE AND APPLICATIONS CONFERENCE WORKSHOPS (COMPSACW), 34., 2010, Seoul. **Proceedings...** Seoul, 2010. p. 317-322.

KOSCHMIDER, A.; OBERWEIS, A. Ontology based business process description. In: INTERNATIONAL CONFERENCE, CAISE, 17., 2005, Porto. **Proceedings...** Porto, 2005. p. 321-333.

LAUTENBACHER, F.; BAUER, B.; SEITZ, C. Semantic Business Process Modeling-Benefits and Capability. In: AAAI SPRING SYMPOSIUM: AL MEETS BUSINESS RULES AND PROCESS MANAGEMENT, 2008, Palo Alto. **Proceedings...** Palo Alto, 2008. p. 71-76.

LEOPOLD, H.; MENDLING, J.; GÜNTHER, O. Learning from Quality Issues of BPMN Models from Industry. **IEEE Software**, v. 33, n. 4, p. 26-33, 2016.

MANOLA, F.; MILLER, E. RDF Primer, 2004. Disponível em: <http://www.w3.org/TR/2004/REC-rdf-primer-20040210/>>. Acesso em: 22 jun. 2016.

MISSIKOFF, M.; PROIETTI, M.; SMITH, F. Linking ontologies to business process schemas. **IASI-CNR Tech**, v. 20, p. 1-20, 2010.

NARDI, J. C.; FALBO, R. A. Uma Ontologia de Requisitos de Software. In: WORKSHOP IBEROAMERICANO DE INGENIERÍA DE REQUISITOS Y AMBIENTES DE SOFTWARE, 9., 2006, La Plata. **Anais...** La Plata, 2006. p. 111-124.

NOGUEIRA, F. A. **Levantamento e especificação de requisitos de software utilizando modelos**

de processos de negócio. Dissertação (Mestrado) - Universidade Estadual Paulista, 2017.

NOGUEIRA, F., A.; OLIVEIRA, H., C. A heuristic-based approach to improve the quality of BPMN business process models. In: IEEE INTERNATIONAL CONFERENCE ON APPLICATION OF INFORMATION AND COMMUNICATION TECHNOLOGIES (AICT), 17., 2017a, Moscow. **Proceedings...** Moscow, 2017a. p. 1-6.

_____. Application of heuristics in business process models to support software requirements specification. In: INTERNATIONAL CONFERENCE ON ENTERPRISE INFORMATION SYSTEMS (ICEIS), 19., 2017b, Funchal. **Proceedings...** Funchal, 2017b. p. 1-12.

OLIVEIRA, M. A. B.; VIEIRA, S. R. C.; SANTOS, D. V.; MARCZAK, S.; CONTE, T. Elicitação de requisitos a partir de modelos de processos de negócio e modelos organizacionais: uma pesquisa para definição de técnicas baseadas em heurísticas. In: REQUIREMENTS ENGINEERING BRAZIL, 2013, Rio de Janeiro. **Anais...** Rio de Janeiro, 2013. p. 86-91.

OMG - OBJECT MANAGEMENT GROUP. **Business Process Model and Notation (BPMN) Version 2.0**, 2011. Disponível em: <<http://www.bpmn.org/>>. Acesso em: 23 maio 2016.

_____. **Object Management Group**, 2016. Disponível em: <<http://www.omg.org/>>. Acesso em: 16 maio 2016.

PARREIRA, P. A. J.; PENTEADO, R. A. D. Domain Ontologies in the Context of Requirements Engineering. In: INTERNATIONAL CONFERENCE OF COMPUTER SYSTEMS AND APPLICATIONS, 12., 2015, Marrakech. **Proceedings...** Marrakech, 2015. p. 1-8.

PEDRINACI, C.; DOMINGUE, J.; BRELADGE, C.; VAN LESSEN, T.; KARASTOYANOVA, D.; LEYMANN, F. Semantic business process management: Scaling up the management of business processes. In: IEEE INTERNATIONAL CONFERENCE ON SEMANTIC COMPUTING, 2008, Santa Clara. **Proceedings...** Santa Clara, 2008. p. 546-553.

PIZZOLETO, A., V.; OLIVEIRA, H., C. A systematic approach to evaluate enterprise ontologies using testing techniques of software usability. In: INTERNATIONAL CONFERENCE ON SOFTWARE ENGINEERING RESEARCH AND PRACTICE (SERP), 2016. Athens. **Proceedings...** Athens, 2016. p. 125-131.

PUCCI, M. A. F. S. **Gerenciamento de processos de negócio voltado à Engenharia de Software, com ênfase em modelagem de processos.** Trabalho de Conclusão de Curso (Graduação em Ciência da Computação). Universidade Estadual Paulista Júlio de Mesquita Filho, Rio Claro, 2016.

RDF WORKING GROUP. **Resource Description Framework (RDF)**, 2014. Disponível em: <https://www.w3.org/RDF/>. Acesso em: 15 dez. 2016.

SATELI, B.; ANGIUS, E.; RAJIVELU, S., S.; WITTE, R. ReqWiki: A Semantic System for Collaborative Software Requirements Engineering. In: INTERNATIONAL SYMPOSIUM ON WIKIS AND OPEN COLLABORATION (WIKISYM), 8., 2012, Linz. **Proceedings...** Linz, 2012, p. 1-27.

SCHAFFERT, S.; BISCHOF, D.; BUERGER, T.; GRUBER, A.; HILZENS AUER, W. Learning with semantic wikis. In: WORKSHOP ON SEMANTIC WIKIS – FROM WIKI TO SEMANTICS (SEMWIKI2006), 1., 2006, Budva. **Proceedings...** Budva, 2006, p. 109-123.

SGANDERLA, K. **Um BPMN para cada propósito de modelagem de processos**, 2012. Disponível em: <<http://blog.iprocess.com.br/2012/04/um-bpmn-para-cada-proposito-de-modelagem-de-processos/>>. Acesso em: jul. 2016.

TERNAI, K.; TÖRÖK, M.; VARGA, K. Corporate Semantic Business Process Management. In: GÁBOR, A.; KÖ, A. **Corporate Knowledge Discovery and Organizational Learning: The Role, Importance, and Application of Semantic Business Process Management.** Springer International Publishing, 2016. p. 33-57. (Knowledge Management and Organizational Learning, v. 2).

TESSARI, R. **Gestão de processos de negócio**: um estudo de caso da BPMN em uma empresa do setor moveleiro. Dissertação (Mestrado) - Universidade de Caxias do Sul, 2008.

THOMAS, O.; FELLMANN, M. Semantic EPC: enhancing process modeling using ontology languages. In: WORKSHOP ON SEMANTIC BUSINESS PROCESS AND PRODUCT LIFECYCLE MANAGEMENT (SBPM 2007), 2007, Innsbruck. **Proceedings...** Innsbruck, 2007. p. 64-75.

VAN DER AALST, W. M. P. Patterns and XPD: a critical evaluation of the XML Process Definition Language. **BPM Center Report BPM-03-09**, p. 1-30, 2003.

WAND, Y.; WEBER, R. Research commentary: information systems and conceptual modeling - a research agenda. **Information Systems Research**, v. 13, n. 4, p. 363-376, 2002.

WETZSTEIN, B.; MA, Z.; FILIPOWSKA, A.; KACZMAREK, M.; BHIRI, S.; LOSADA, S.; LOPEZ-COBO, J.; CIRCUREL, L. Semantic business process management: A lifecycle based requirements analysis. In: WORKSHOP ON SEMANTIC BUSINESS PROCESS AND PRODUCT LIFECYCLE MANAGEMENT (SBPM 2007), 2007, Innsbruck. **Proceedings...** Innsbruck, 2007. p. 1-11.

W3C – WORLD WIDE WEB CONSORTIUM. **OWL Web Ontology Language Guide**, 2004a. Disponível em: <<https://www.w3.org/TR/2004/REC-owl-guide-20040210/#StructureOfDoc/>>. Acesso em: 16 dez 2016.

_____. **OWL Web Ontology Language Overview**, 2004b. Disponível em: <<https://www.w3.org/TR/2004/REC-owl-features-20040210/#s1.3/>>. Acesso em: 16 dez 2016.

_____. **PROV - A Framework for Provenance Interchange**, 2013a. Disponível em: <<https://www.w3.org/blog/SW/2013/03/21/prov-a-framework-for-provenance-interchange/>>. Acesso em: 20 jun. 2017.

_____. **RDF Schema 1.1**, 2014. Disponível em: <<https://www.w3.org/TR/rdf-schema/>>. Acesso em: 16 dez. 2016.

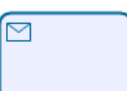
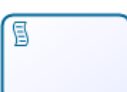
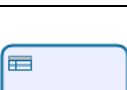
_____. **SPARQL 1.1 Overview**, 2013b. Disponível em: <<https://www.w3.org/TR/sparql11-overview/>>. Acesso em 09 jun. 2017.

WfMC – WORKFLOW MANAGEMENT COALITION. **Process Definition Interface**: XML Process Definition Language. WfMC-TC-1025. Version 2.2. 2012. Disponível em: <<http://www.xpdl.org>>. Acesso em: 12 dez. 2016..

APÊNDICE A – NOTAÇÃO BPMN

As **Tabelas 26 a 31** apresentam os elementos da notação BPMN v2.0, agrupados como elementos para representação de: atividades, eventos, *gateways*, estruturas básicas, artefatos e subprocessos, respectivamente.

Tabela 26 - Representações para as atividades na notação BPMN.

ATIVIDADES	DESCRIÇÃO
	Simples: tipo genérico de atividade, empregado normalmente nos estágios iniciais do desenvolvimento do processo, sem que haja a necessidade de discriminação em detalhes.
	Serviço: atividade associada a algum tipo de serviço ou sistema, e que não necessita de interferência humana.
	Usuário: atividade realizada por uma pessoa auxiliada por um sistema.
	Recebimento de mensagem: atividade alimentada por participantes externos.
	Envio de mensagem: atividade relacionada ao envio de mensagem a um participante externo.
	Script: atividade executada por meio de um mecanismo de processos de negócio. A definição do <i>script</i> ocorre a partir de um modelador, em uma linguagem que possa ser interpretada pelo motor de processos.
	Manual: atividade exclusivamente realizada por humanos.
	Regra de negócio: atividade que fornece um mecanismo para que o processo defina informações para a execução de um motor de regras de negócio para a obtenção dos cálculos realizados por esse motor de regras de negócio.

Fonte: elaborado pelo autor.

Tabela 27 - Representações de eventos na notação BPMN.

EVENTOS DE INÍCIO	EVENTOS INTERMEDIÁRIOS	EVENTOS DE FIM	DESCRIÇÃO
			Nenhum: elemento que representa o início, a continuidade ou o fim de um processo.
	 		Mensagem: Indica o recebimento ou envio de uma mensagem
			Tempo: indica uma parada definida por um tempo determinado para iniciar/continuar o processo
			Condição: elemento que determina o aguardo de uma condição ou expressão até que a mesma seja verdadeira para dar início/continuidade ao processo.
	 		Sinal: demonstra que em determinado ponto do fluxo haverá o recebimento ou o envio de um sinal.
	 		Múltiplo: determina a existência de diversos eventos que disparam ou continuam o processo. Todavia, somente um é necessário. Para o caso de evento de fim, indica a existência de sequências ao finalizar o processo, sendo que todas ocorrerão.
			Múltiplo paralelo: indica que existem vários eventos e que todos precisam ocorrer em paralelo para que se inicie ou continue o processo.
	 		Link: interliga atividades de um mesmo processo, indicando recebimento ou envio do <i>link</i> , deixando o diagrama mais limpo.
			Compensação: indica a necessidade de uma compensação, ou seja, alguma atividade que será desfeita no processo.
			Escalar: indica a necessidade de escalabilidade de um processo, comunicando uma falha de execução de alguma atividade do subprocesso ao seu processo pai.
			Exceção: denota a criação de um erro no fim de um processo.
			Cancelamento: é utilizado dentro de um subprocesso de transação, indicando que a mesma deve ser cancelada. Efetua o disparo de um evento intermediário, receptor de cancelamento, na fronteira do subprocesso.

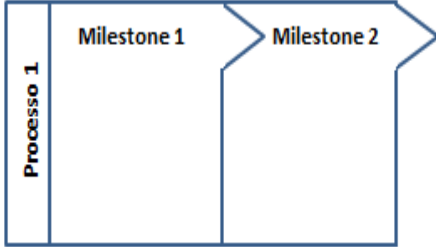
Fonte: elaborado pelo autor.

Tabela 28 - Representações de *gateways* na notação BPMN.

GATEWAYS	DESCRIÇÃO
	Gateway exclusivo: é utilizado para a situação em que existe apenas um caminho a ser tomado após uma decisão. Pode ser usado como ponto de convergência ou divergência.
	Gateway paralelo: é utilizado para indicar que todos os caminhos devem ser seguidos simultaneamente, sem que haja a necessidade de se tomar uma decisão.
	Gateway inclusivo: utilizado quando há a possibilidade do seguimento de vários caminhos para uma decisão ser tomada.
	Gateway baseado em eventos: só existe um caminho a ser seguido, e isso depende do evento intermediário que ocorrer primeiro.
	Gateway exclusivo baseado em eventos: semelhante ao <i>gateway</i> baseado em eventos, mas nesse caso cada ocorrência de um evento posterior inicia uma nova instância do processo.
	Gateway paralelo baseado em eventos: semelhante ao <i>gateway</i> baseado em eventos, porém só há o seguimento do fluxo quando ocorre um evento.
	Gateway complexo: utilizado para decisões com base em expressões complexas e situações que os outros gateways não conseguem contemplar.

Fonte: elaborado pelo autor.

Tabela 29 - Estruturas básicas da notação BPMN.

ESTRUTURAS BÁSICAS	DESCRIÇÃO
	Piscina ou <i>Pool</i>: utilizado para representação de um ator em um processo.
	Raia ou <i>Lane</i>: subdivisão de uma piscina, empregada para organizar e categorizar os participantes e suas atividades dentro de um processo.
	Etapa ou <i>Milestone</i>: utilizado para indicar as diferentes etapas de um processo, exibindo assim uma mudança de fase.

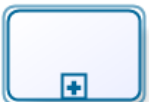
Fonte: elaborado pelo autor.

Tabela 30 - Artefatos presentes na notação BPMN.

ARTEFATOS	DESCRIÇÃO
	Objeto de dados: disponibiliza informações a respeito de documentos, dados e outros objetos. É usado e atualizado conforme necessidade durante o processo.
	Depósito de dados: fornece um mecanismo de resgate ou atualização de dados e/ou informações que serão persistidos além do escopo do processo.
	Grupo: agrupamento de elementos do diagrama que não afeta o mesmo. É empregado para documentação ou análise.
	Anotação: possibilita a inserção de informações adicionais no diagrama, com o objetivo de facilitar a leitura do diagrama pelo usuário.

Fonte: elaborado pelo autor.

Tabela 31 - Representações de subprocessos na notação BPMN.

SUBPROCESSOS	DESCRIÇÃO
	Subprocesso: empregado em uma atividade que contém um conjunto de outras atividades. Os subprocessos são dependentes de um processo, mas possuem fluxo próprio.
	Subprocesso reutilizável: realiza o encapsulamento da chamada a um processo global dentro do processo de negócio. A ativação desse subprocesso transfere o controle processual para o processo global.
	Transacional: o comportamento do subprocesso é controlado por meio de um protocolo de transação. Permite que todas as suas atividades sejam concluídas com sucesso ou canceladas e compensadas.
	Ad hoc: subprocesso que é formado por um grupo de atividades que não têm obrigatoriamente relações sequenciais. Esse subprocesso é concluído após a execução de todas as atividades.
	Loop: indica a repetição de um subprocesso até que se atinja a condição estabelecida anteriormente. Nesse caso não se sabe o número de vezes em que será repetido o processo.
  (Paralelo e Sequencial, respectivamente)	Múltiplas instâncias: os subprocessos podem ser repetidos de maneira paralela ou sequencial. A quantidade de vezes em que esses subprocessos serão repetidos é conhecida antes do início dos mesmos.

Fonte: elaborado pelo autor.

APÊNDICE B - NOTAÇÕES RELACIONADAS: DMN, CMMN E BPMN

Este Apêndice apresenta os dois padrões do Consórcio OMG que podem ser integrados à BPMN na modelagem de processos de negócio. Para isso, primeiramente será apresentado um panorama da notação DMN, seguido de uma visão geral da notação CMMN. Assim, a última parte deste Apêndice apresentará alguns comentários sobre o uso integrado dessas três notações e um quadro comparativo entre elas.

Notação DMN

A notação DMN (*Decision Model Notation*) foi introduzida pelo consórcio OMG em 2013. O objetivo é providenciar uma notação comum de fácil entendimento para todos os usuários de negócios envolvidos nas decisões (regras de negócio). Entre esses usuários estão os analistas de negócio, responsáveis por criar os requisitos iniciais de decisão, bem como modelos de decisão mais detalhados. Os desenvolvedores técnicos também são beneficiados, uma vez que são responsáveis pela automatização das decisões nos processos. Esse benefício se estende a toda a equipe de negócios, responsável por gerenciar e monitorar essas decisões. De certa forma, a notação DMN busca minimizar o distanciamento (*gap*) entre o projeto e a implementação da decisão de negócio.

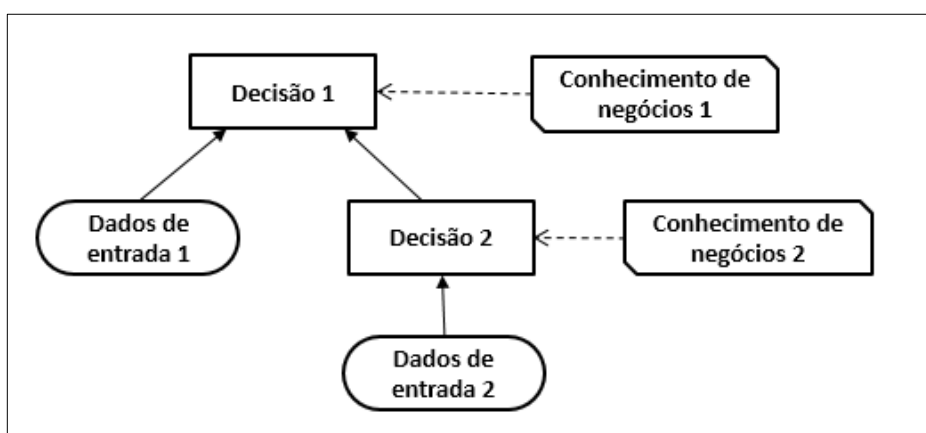
O objetivo da notação DMN é fornecer os construtores necessários para modelar as decisões, de modo que a tomada de decisão possa ser facilmente representada por diagramas gráficos. DMN é composta por cinco componentes principais (GOLDBERG; HALLE, 2015; OMG, 2016):

1. **Notação em nível de requisitos:** apresenta os requisitos de decisão e seus relacionamentos por meio de diagramas de requisitos de decisão;
2. **Notação em nível de decisão lógica:** representa os fluxos de tomadas de decisão utilizando somente elementos de negócios;
3. **Linguagem de expressão FEEL (*Friendly Enough Expression Language*):** define um padrão semântico de execução para expressões presentes nos modelos de decisões, com tipos de dados e sintaxe próprios;
4. **Metamodelo de suporte:** representação dos elementos presentes na notação DMN, por meio de diagramas de classes UML;
5. **Níveis específicos de conformidade:** requisitos de conformidade estabelecidos

para padronizar a comunicação entre aplicações que usam a especificação DMN.

Segundo Goldberg e Halle (2015), os principais elementos do diagrama de requisitos de decisões DMN são: decisões, conhecimento de negócios, fontes de conhecimento de negócios e dados de entrada. Na **Figura 30** é mostrado um exemplo de diagrama de requisitos de decisão, onde cada caixa titulada como “Decisão” corresponde a uma conclusão a partir de uma família de regras de negócio.

Figura 30 - Diagrama de requisitos de decisão proposto na notação DMN.



Fonte: Goldberg e Halle (2015).

Segundo o Consórcio OMG (2016), a abordagem DMN define três níveis de conformidade. O primeiro nível inclui conformidades em relação à notação e consistência das interpretações, no que diz respeito às semânticas das expressões especificadas na lógica de decisão. Já o segundo nível inclui conformidades tanto do nível um quanto suporte ao S-FEEL, subconjunto de linguagem de expressões FEEL. E por fim, o terceiro nível inclui conformidade ao nível dois e suporte à linguagem FEEL.

Notação CMMN

Para que a modelagem de processos de negócio pudesse ser mais informativa semanticamente, o consórcio OMG introduziu a notação CMMN (*Case Management Model and Notation*), com o objetivo de propiciar a modelagem e representação gráfica de casos de uso. CMMN foi projetada para complementar a notação BPMN e sua versão 1.0 foi disponibilizada em maio de 2014.

Segundo Hinkelmann (2015), qualquer caso de uso individual pode ser representado

através da aplicação e utilização de tarefas *ad-hoc*. Essas tarefas, fora de uma linha temporal, visam atingir um objetivo, resolver algum problema ou demanda específica do usuário.

Um caso de uso apresenta duas fases distintas: a fase de projeto e a de execução (OMG, 2016). Na fase de projeto, os analistas de negócios estão envolvidos diretamente com a modelagem. Essa modelagem inclui a definição de tarefas, que formam os segmentos predefinidos de um caso de uso. Dentre essas tarefas estão as tarefas aplicadas em situações frequentes, mas que não pertencem necessariamente a uma parte do caso, denominadas discricionários. As tarefas discricionárias podem ser aplicadas adicionalmente ao caso de uso, a critério dos responsáveis pelo caso de uso. A fase de execução compreende a realização das tarefas definidas, conforme o que planejado na modelagem. Segundo o consórcio OMG (2016), o plano do caso de uso pode evoluir de maneira contínua, com a adição de tarefas discricionárias na instância em execução desse plano.

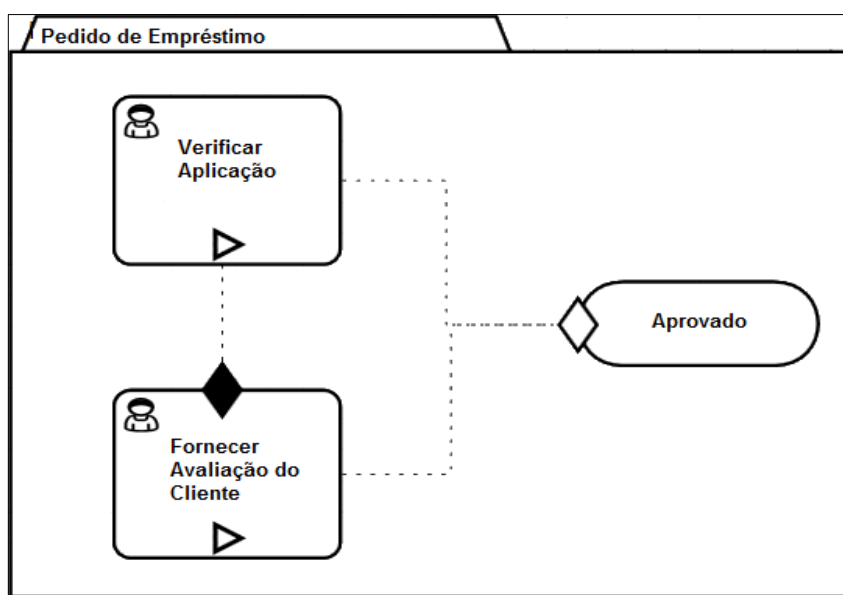
Segundo Hinkelmann (2015), um modelo de caso de uso definido na notação CMMN é constituído por três componentes: (1) um arquivo de caso de uso; (2) um modelo de plano do caso de uso; (3) um conjunto de funções do caso de uso.

Um arquivo de caso contém toda a informação ou referência de informação necessária para o gerenciamento do caso de uso (OMG, 2016). As informações presentes em um arquivo de caso de uso servem, por exemplo, para criação de eventos e avaliação de expressões. Um evento é algo que pode ocorrer durante o curso de um caso de uso, podendo ativar gatilhos para conclusão de etapas e tarefas, por exemplo. Esses eventos podem ser subdivididos em eventos de tempo e de usuário. Hinkelmann (2015) destaca que um arquivo de caso de uso pode ser formado por vários itens e que esses itens podem representar uma parte da informação de qualquer natureza, seja ela simples ou complexa, desestruturada ou estruturada.

O plano do caso de uso pode ser composto por itens de quatro tipos: (1) tarefas (discricionárias ou não); (2) fragmentos de plano e etapas; (3) escutadores de eventos; (4) marcos (HINKELMANN, 2015). Uma tarefa é uma unidade atômica de um de trabalho (OMG, 2016). Há três categorias de tarefas: humanas, de processo e de caso de uso. As tarefas humanas são executadas pelos envolvidos no trabalho do caso de uso e podem ser classificadas como bloqueadoras e não bloqueadoras. As tarefas humanas bloqueadoras são aquelas que ficam em espera até que o trabalho relacionado à tarefa seja completado. As tarefas humanas não bloqueadoras são completadas imediatamente à instanciação, não ficando em espera. As tarefas de processo, por sua vez, são utilizadas pelos casos de uso para a chamada a um processo de negócio. Por fim, as tarefas de caso de uso são utilizadas para que um caso de uso chame outro caso de uso, conforme a necessidade.

Na **Figura 31** apresenta-se um exemplo de modelo de plano de caso de uso formado por tarefas humanas bloqueadoras. As tarefas comuns são representadas por um retângulo de bordas levemente arredondadas. A forma retangular de extremidades semi-arredondadas, por sua vez, representa um *Milestone*, que define uma meta a ser atingida. Para o controle do fluxo do caso de uso em questão são empregadas as sentinelas, representadas pelo diamante cheio (preenchidos pela cor preta) e vazio (na cor branca). Os triângulos presentes na figura identificam que a tarefa onde os mesmos estão inseridos depende de ativação manual.

Figura 31 - Exemplo de modelo de plano de caso de uso.



Fonte: Hinkelmann (2015).

Hinkelmann (2015) afirma que existem dois tipos de elementos para o controle do fluxo de um modelo de plano de caso de uso: sentinelas e tabelas de planejamento. As sentinelas definem os critérios segundo os quais os itens de plano são habilitados (entrada) ou encerrados (saída). As sentinelas de critério de entrada são representadas graficamente por um diamante raso, enquanto que sentinelas de critério de saída são representadas por diamantes cheios, preenchidos com a cor preta. As tabelas de planejamento, por sua vez, definem o escopo do planejamento e podem ser atribuídas a tarefas humanas ou etapas. Esses elementos são representados graficamente por tabelas e podem apresentar diversos itens discricionários. As tabelas de planejamento apresentam marcadores de expansão e de contração. Quando o marcador de expansão é utilizado para uma tabela de planejamento em uma etapa são exibidos os itens discricionários armazenados pela etapa em questão. Já a utilização do marcador de expansão nos elementos identificados como tarefas humanas exibe os itens discricionários fora da tarefa em questão.

De modo geral, um fragmento do plano de caso de uso armazena itens do plano e sentinelas. É representado graficamente por um retângulo com bordas tracejadas e de cantos suavemente arredondados, juntamente com um marcador de expansão e de contração (HINKELMANN, 2015). As etapas de um caso de uso são consideradas fragmentos de plano que podem ser rastreados. As etapas também podem ser consideradas como subcasos ou “episódios” de um caso de uso. Os *Milestones*, por sua vez, são definidos graficamente por formas retangulares de extremidades semi-arredondadas e representam uma meta que deve ser atingida (OMG, 2016). Esses *Milestones* permitem a avaliação dos processos ocorridos em um caso de uso. Um *Milestone* pode conter zero ou mais critérios de entrada que definem quando essa milha é atingida.

Integração BPMN, DMN e CMMN

Gagne (2015) apresenta um comparativo entre as notações BPMN, DMN e CMMN. Segundo ele, os processos estão para BPMN assim como as decisões estão para DMN e os casos de uso estão para CMMN. A notação BPMN trabalha com atividades, enquanto a notação DMN trabalha com regras. Já a notação CMMN trabalha com eventos. Em relação à linguagem de cada uma dessas notações, tem-se BPMN como uma linguagem procedural, enquanto DMN é uma linguagem de conhecimento e CMMN é uma linguagem declarativa. A comparação entre essas notações pode ser vista na **Tabela 30**.

Tabela 32 - Comparação entre as notações BPMN, CMMN e DMN.

BPMN	CMMN	DMN
Processos	Casos	Decisões
Atividades	Eventos	Regras
Transitiva	Contextual	Aplicada
Dado	Informação	Conhecimento
Procedural	Declarativa	Funcional
Token	Evento-Condição-Ação	Primeira Ordem Lógica (POL)

Fonte: Gagne (2015).

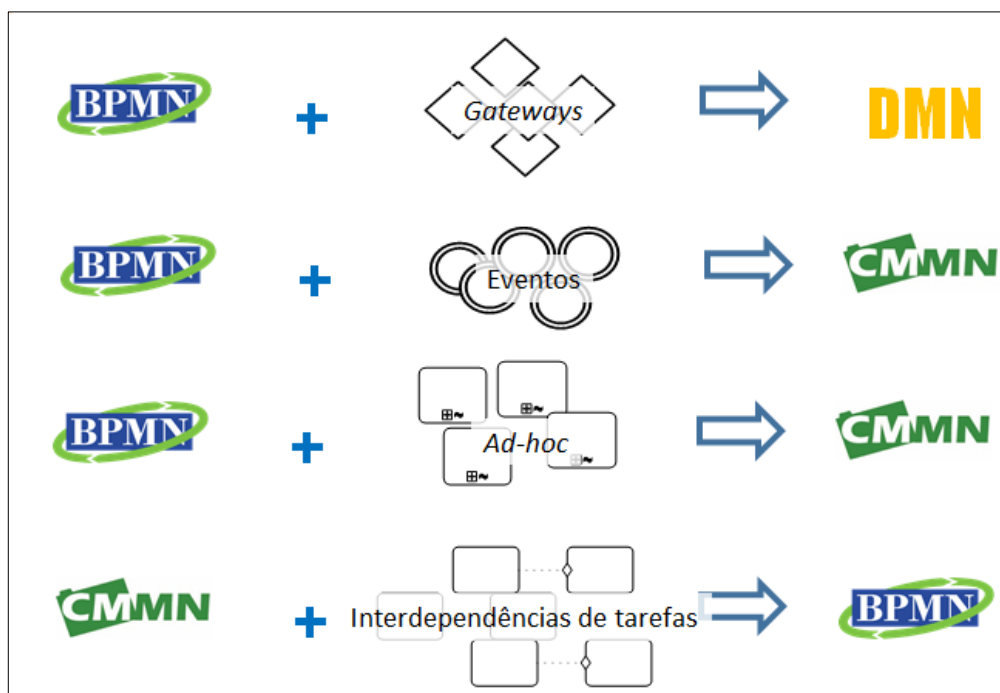
Gagne (2015) também apresenta uma abordagem de aplicação de melhores práticas para o uso dessas notações. O uso de *gateways* em um modelo notado em BPMN pode ser substituído por um modelo em DMN. A utilização de eventos em um modelo notado em BPMN pode ser substituída por um modelo de caso de uso em CMMN. Para o cenário de um modelo

notado em BPMN que apresente muitas tarefas *ad-hoc*, a criação de um modelo CMMN torna-se interessante. Em um modelo de caso de uso em CMMN que apresenta tarefas interdependentes há a possibilidade em se empregar um modelo notado em BPMN.

Assim, embora cada uma dessas notações seja destinada a uma finalidade, em determinado momento da modelagem de uma cadeia de processos de negócio, a integração dessas notações é de interesse. A **Figura 32** apresenta uma esquematização das melhores práticas, onde a aplicação constante de um elemento em determinada notação pode ser substituída pela inclusão de um modelo em outra notação mais adequada para a situação.

Destaca-se que alguns modeladores oferecem já oferecem suporte à integração das notações BPMN, DMN e CMMN, como é o caso do Camunda, que em sua documentação, em <https://docs.camunda.org/manual/develop/user-guide/process-engine/decisions/#decisions-inside-bpmn>, aborda os recursos disponibilizados no modelador para a integração entre as notações DMN e BPMN. O modelador Camunda também apresenta, em sua documentação, suporte à integração entre as notações DMN e CMMN.

Figura 32 - Melhores práticas para as notações BPMN, DMN e CMMN.



Fonte: Gagne (2015).

APÊNDICE C – METAMODELO META2ONTO (CD-R *anexo*)

Este Apêndice apresenta o metamodelo de classes criado para utilização no processo sistemático apresentada neste trabalho. O metamodelo, definido na linguagem UML, foi criado para auxiliar o processo de mapeamento dos elementos XPDL para os elementos da ontologia, conforme apresentado no *Capítulo 4*.

O conteúdo deste Apêndice encontra-se em formato digital, em CD-R anexo a esta dissertação, em pasta denominada “APENDICE C – Metamodelo Meta2Onto”.

APÊNDICE D – *FRAMEWORK* APACHE JENA

Este Apêndice contempla algumas informações básicas sobre o *framework* Apache JENA, utilizado para o desenvolvimento da ferramenta PM2ONTO, que visa automatizar o processo sistemático apresentado neste trabalho. Trata-se de um *framework* de licença gratuita, disponível no endereço <http://jena.apache.org>.

Ele apresenta os seguintes recursos para a construção de aplicações para Web Semântica:

- **API RDF:** API destinada à criação e leitura de grafos expressos em RDF. Esses grafos podem ser serializados na linguagem RDF/XML ou *Turtle*;
- **ARQ:** uma *engine* compatível com a linguagem SPARQL. Essa *engine* é destinada a consultas em grafos expressos em RDF;
- **TDB:** utilizado para a persistência de dados. Suporta todas as API's existentes no Apache JENA;
- **Fuseki:** fornece um ponto acessível através do protocolo HTTP para execução de consultas em SPARQL para dados em RDF;
- **Ontology API:** API destinada à criação e manipulação de ontologias em RDFS e OWL;
- **Inference API:** API destinada à recursos para realização de inferências em ontologias. Com essa API, é possível definir uma série de regras de inferência que serão utilizados pelo *reasoner* para as ontologias

APÊNDICE E – SISTEMA PM2ONTO: USO E CÓDIGO (CD-R *anexo*)

Este Apêndice contém o código fonte em Java da ferramenta P2MONTTO v1.0, desenvolvida para automatizar o processo sistemático apresentado. Além disso, este Apêndice contém instruções para a configuração de todos os recursos necessários para execução da ferramenta.

O conteúdo deste Apêndice encontra-se em formato digital, em CD-R anexo a esta dissertação, em pasta denominada “APENDICE E – Sistema PM2ONTO-Uso&Codigo”.

APÊNDICE F – MODELOS EM BPMN E XPD L (CD-R *anexo*)

Este Apêndice contém os modelos de processos de negócio em BPMN v2.0 utilizados nos três estudos de caso apresentados nas Seções 5.3 a 5.5, visando avaliar o processo sistemático apresentado neste trabalho. Esses arquivos foram exportados por meio da ferramenta de modelagem *Bizagi Modeler* para arquivos XPD L v2.2, os quais também se encontram neste Apêndice.

O conteúdo deste Apêndice encontra-se em formato digital, em CD-R anexo a esta dissertação, em pasta denominada “APENDICE F – Modelos BPMN&XPD L”.

APÊNDICE G – ONTOLOGIAS GERADAS (CD-R *anexo*)

Este Apêndice contém todas as ontologias geradas nos três estudos de caso apresentados nas Seções 5.3 a 5.5, visando avaliar o processo sistemático apresentado neste trabalho. Essas ontologias estão descritas na linguagem OWL, sendo possível visualizá-las por meio da ferramenta *Protégé*.

O conteúdo deste apêndice encontra-se em formato digital, em CD-R anexo a esta dissertação, em pasta denominada “APENDICE G – Ontologias geradas”.

Autorizo a reprodução xerográfica para fins de pesquisa

Rio Claro, 04/06/2018

Luiz R. F.

Assinatura