



COPPE/UFRJ

REPOSITÓRIO DE CONHECIMENTO DE UM AMBIENTE DE
APOIO A EXPERIMENTAÇÃO EM ENGENHARIA DE SOFTWARE

Vitor Pires Lopes

Dissertação de Mestrado apresentada ao Programa de Pós-graduação em Engenharia de Sistemas e Computação, COPPE, da Universidade Federal do Rio de Janeiro, como parte dos requisitos necessários à obtenção do título de Mestre em Engenharia de Sistemas e Computação.

Orientador: Guilherme Horta Travassos

Rio de Janeiro

Maio de 2010

REPOSITÓRIO DE CONHECIMENTO DE UM AMBIENTE DE
APOIO A EXPERIMENTAÇÃO EM ENGENHARIA DE SOFTWARE

Vitor Pires Lopes

DISSERTAÇÃO SUBMETIDA AO CORPO DOCENTE DO INSTITUTO ALBERTO
LUIZ COIMBRA DE PÓS-GRADUAÇÃO E PESQUISA DE ENGENHARIA
(COPPE) DA UNIVERSIDADE FEDERAL DO RIO DE JANEIRO COMO PARTE
DOS REQUISITOS NECESSÁRIOS PARA A OBTENÇÃO DO GRAU DE MESTRE
EM CIÊNCIAS EM ENGENHARIA DE SISTEMAS E COMPUTAÇÃO.

Examinada por:

Prof. Guilherme Horta Travassos, D.Sc.

Prof. Toacy Cavalcante de Oliveira, D.Sc.

Prof. Márcio de Oliveira Barros, D.Sc.

Prof. Jorge Calmon de Almeida Barros, D.Sc.

RIO DE JANEIRO, RJ – BRASIL

MAIO DE 2010

Lopes, Vitor Pires

Repositório de Conhecimento de um ambiente de Experimentação em Engenharia de Software / Vitor Pires Lopes. – Rio de Janeiro: UFRJ/COPPE, 2010.

XIX, 163 p.: il.; 29,7 cm.

Orientador: Guilherme Horta Travassos

Dissertação (mestrado) – UFRJ/ COPPE/ Programa de Engenharia de Sistemas e Computação, 2010.

Referências Bibliográficas: p. 106-113.

1. Engenharia de Software Experimental 2. Repositórios de Conhecimento 3. Ontologias 4. Estudos Experimentais em Engenharia de Software. I. Travassos, Guilherme Horta. II. Universidade Federal do Rio de Janeiro, COPPE, Programa de Engenharia de Sistemas e Computação. III. Título.

*À minha mãe Angelica,
ao meu pai Luis Cassio,
ao meu irmão Thales,
aos meus queridos vovôs e vovós,
à minha Tia Paula,
aos amigos,*

Agradecimentos

À Deus pela dádiva da vida e inúmeras bênçãos que derrama sobre mim e minha família.

À minha família amada, razão fundamental da minha vida e a qual devo tudo que sou. À minha mãe Angelica e meu pai Luis Cassio, por todo amor, carinho e dedicação ao longo deste percurso. Devo tudo aos senhores, meus pais, e serei eternamente grato. Em especial à minha mãe, e também minha grande amiga, por ter sido a única luz em vários de meus dias escuros, me fazendo perceber que a vida é muito maior que um simples momento de adversidade. Ao meu querido irmão Thales, companheiro e grande amigo. Não há nenhuma distância física capaz de diminuir nossos laços de fraternidade. Eles só se fazem aumentar dia após dia, reafirmados no amor e na amizade.

Ao meu amado vovô Willian, que está sob a luz da face de nosso Senhor, olhando por mim e minha família, exemplo de homem e que representa minha meta maior de vida como ser humano. Ao meu querido vovô Cassio, outro grande exemplo de homem e que amo intensamente. Às minhas amadas avós Aurelina e Rosária, minhas jóias raras. A presença das senhoras em minha vida reforça a existência de Deus. À minha tia Paula, a quem amo muito e representa muito na minha vida.

À Suelen, uma pessoa de extrema importância em minha vida, por me trazer a perspectiva e desejo de ser e fazê-la feliz.

Ao meu orientador e amigo Guilherme, fundamental para meu desenvolvimento. Frente a um novo contexto, sua confiança, seu apoio e dedicação na minha trajetória transformaram momentos difíceis e penosos em ocasiões de ensinamento e aprendizagem contínuos.

Aos meus amigos, velhos e novos, em especial: Paulo Sérgio, Jobson, Arilo, Rodrigo Spínola, Rodrigo Pereira, Rodrigo Murta, Marcos (Kali), Marco Antônio, Wallace Pereira, Priscila, Taisa, Beto e Breno, pelas risadas e conversas descontraídas.

Aos professores Toacy, Márcio e Jorge, por terem aceitado participar desta banca.

A todos os amigos e colegas dos Grupos de Experimentação e Reutilização da COPPE/UFRJ, por todas as contribuições pessoais e intelectuais.

A CAPES pelo apoio financeiro.

Resumo da Dissertação apresentada à COPPE/UFRJ como parte dos requisitos necessários para a obtenção do grau de Mestre em Ciências (M.Sc.)

REPOSITÓRIO DE CONHECIMENTO DE UM AMBIENTE DE
APOIO A EXPERIMENTAÇÃO EM ENGENHARIA DE SOFTWARE

Vitor Pires Lopes

Maio/2010

Orientador: Guilherme Horta Travassos

Programa: Engenharia de Sistemas e Computação

A execução de estudos experimentais em Engenharia de Software consome tempo e produz grande volume de informação e conhecimento, cuja gestão é complexa. Conseqüentemente, é importante ter uma infra-estrutura computadorizada para apoiar a execução do estudo experimental. Esta infra-estrutura deve prover mecanismos para representar e gerenciar eficientemente o conhecimento sobre experimentação. Nesta dissertação nós discutimos a construção de um repositório de conhecimento para ambiente de apoio à experimentação em Engenharia de Software (eSEE – experimental Software Engineering Environment), uma infra-estrutura capaz de instanciar ambientes de Engenharia de Software para apoiar a condução do processo de experimentação em Engenharia de Software.

Este repositório é fundamentado em dois mecanismos de representação do conhecimento: glossário e ontologias. O primeiro visa estabelecer uma terminologia única na área de Engenharia de Software Experimental. As ontologias constituem a formalização do conhecimento expresso na lista de termos do glossário, expressando os diferentes relacionamentos entre os conceitos. Por fim, este trabalho apresenta um protótipo de ferramenta que ilustra como o repositório pode ser utilizado como núcleo de recuperação de conhecimento no eSEE.

Abstract of Dissertation presented to COPPE/UFRJ as a partial fulfillment of the requirements for the degree of Master of Science (M.Sc.)

KNOWLEDGE REPOSITORY OF AN EXPERIMENTATION
ENVIRONMENT IN SOFTWARE ENGINEERING

Vitor Pires Lopes

May/2010

Advisor: Guilherme Horta Travassos

Department: Computer and Systems Engineering

Experimental studies execution in Software Engineering is time consuming e produces great volume of information and knowledge with complex management. Hence, it is important to have a computerized infrastructure to support the experimental study execution. This infrastructure should provide efficient knowledge representation and management mechanisms regarding Experimentation. In this dissertation we discuss the construction of a knowledge repository for an experimentation environment in Software Engineering (eSEE) – an infrastructure capable of instantiating software engineering environments to manage knowledge about the definition, planning, execution and packaging of experimental studies in software engineering.

This repository is structured under two knowledge representation mechanisms: glossary and ontologies. The first aims at establish a common terminology in Experimental Software Engineering area. The ontologies represent the formalization of the knowledge expressed in the glossary's list of terms. These formalized models express the various relationships between concepts. Finally, this work shows the prototype of a tool that illustrates how the repository can be used as the knowledge retrieve core in eSEE.

Índice

Índice de Figuras	xiii
Índice de Tabelas	xvi
Capítulo 1 Introdução	1
1.1. Motivação	1
1.2. Objetivos.....	3
1.3. Contexto: eSEE (<i>experimental Software Engineering Enviroment</i>)	5
1.5. Metodologia e Atividades.....	6
1.6. Síntese de resultados.....	6
1.7. Organização	7
Capítulo 2 Experimentação em Engenharia de Software	8
2.1. Introdução	8
2.2. Conceitos Básicos.....	9
2.3. Técnicas de Investigação Científica	11
2.4. Classificação de Estudos Experimentais	12
2.4.1. Estudos Primários	13
2.4.1.1. Classificação quanto à Estratégia de Estudo	13
2.4.1.2. Classificação quanto ao Método de Pesquisa Empregado.....	15
2.4.1.3. Classificação quanto ao Ambiente do Estudo	16
2.4.2. Estudos Secundários	17
2.4.3. eSEE (<i>experimental Software Engineering Environment</i>)	18
2.5. Conclusão	22
Capítulo 3 Glossário de Termos de Engenharia de Software Experimental	23
3.1. Motivação	23

3.2. Escopo	25
3.3. Representação Formal do Conhecimento	26
3.4. Proposta de Estrutura do Repositório de Conhecimento	28
3.5. Glossário de Termos do eSEE	30
3.5.1. Histórico da Construção do Glossário de Termos	30
3.5.2. Características do Glossário de Termos	32
3.6. Integração do Glossário ao Repositório de Conhecimento	35
3.7. Revisão do Glossário de Termos	36
3.8. Conclusões.....	43
Capítulo 4 Ontologias do Repositório de Conhecimento do eSEE	44
4.1. Introdução	44
4.2. Processo de Construção das Ontologias	45
4.2.1. Planejamento	48
4.2.2. Especificação	48
4.2.3. Conceitualização.....	48
4.2.4. Formalização	49
4.2.4.1. Formalização da Ontologia Pesquisa Científica.....	51
4.2.4.2. Construção da Subontologia de Estratégia de Estudo	54
4.2.4.3. Construção da Subontologia de Método de Pesquisa.....	58
4.2.4.4. Construção da Subontologia de Ambiente de Estudo	59
4.2.4.5. Formalização da Ontologia do Pacote do Experimento	60
4.2.4.6. Formalização da Associação entre as Ontologias de Pesquisa Científica e Pacote do Experimento	63
4.2.5. Avaliação	66
4.2.6. Implementação.....	66

4.2.7. Integração	67
4.2.8. Documentação	67
4.3. Revisão das Ontologias do Repositório.....	67
4.3.1. Inspeção dos Modelos de Pesquisa-ação	69
4.3.2. Inspeção dos Modelos de <i>Survey</i>	70
4.3.3. Inspeção dos Modelos de Estudo de Caso.....	71
4.3.4. Inspeção dos Modelos de Estudo de Observação	72
4.3.5. Inspeção dos Modelos de Estudo Controlado.....	73
4.3.6. Inspeção do Modelo de Método Qualitativo	74
4.3.7. Inspeção do Modelo de Método Quantitativo	75
4.3.8. Síntese dos resultados.....	76
4.4. Conclusões.....	77
Capítulo 5 Ferramenta para Configuração de Modelos de Pacote de Experimentos em Engenharia de Software.....	79
5.1. Introdução.....	79
5.2. Protótipo da Ferramenta para Configuração de Modelos de Pacote de Experimento	80
5.5. Implementação da Ontologia.....	82
5.5.1. Classes com estereótipos “conceito” e “documento”	83
5.5.2. Atributo cujo tipo é primitivo e string	83
5.5.3. Generalização/Especialização entre classes com estereótipo “conceito”.....	85
5.5.4. Agregação/ Composição entre classes com estereótipo “conceito”	86
5.5.5. Agregação/Composição entre classes com estereótipo “documento”.....	87
5.5.6. Atributo de classes com estereótipo “documento”	89
5.5.7. Associação.....	90

5.6. Estratégia para Extração do Conhecimento da Ontologia	91
5.7. Exemplo de Utilização do Protótipo.....	93
5.8. Conclusões.....	101
Capítulo 6 Conclusão	102
6.1. Conclusões.....	102
6.2. Contribuições.....	103
6.3. Limitações	104
6.4. Perspectivas Futuras	104
Referências Bibliográficas.....	106
Anexo I Modelos Ontológicos do Repositório de Conhecimento do eSEE.....	114
1.1. Introdução	114
1.2. Ontologia de Pesquisa Científica.....	114
1.3. Subontologia de Estratégia de Estudo.....	116
1.3.1. Subontologia de Pesquisa-ação	117
1.3.2. Subontologia de Survey.....	120
1.3.3. Subontologia de Estudo de Caso	120
1.3.4. Subontologia de Estudo de Observação	124
1.3.5. Subontologia de Estudo Controlado	124
1.4. Subontologia de Método de Pesquisa	130
1.4.1. Subontologia de Método Qualitativo.....	130
1.4.2. Subontologia de Método Quantitativo.....	137
1.5. Subontologia de Ambiente de Estudo.....	145
1.5.1. Subontologia de Estudo <i>In Vivo</i>	145
1.5.2. Subontologia de Estudo <i>In Vitro</i>	146
1.5.3. Subontologia de Estudo <i>In Virtuo</i>	147

1.5.4. Subontologia de Estudo <i>In Silico</i>	149
1.1. Referências bibliográficas.....	150
Anexo II Processo de Construção de Ontologias - <i>Methodology</i>	152
1.1. Conceitos Básicos	152
1.2. Processo de Construção de Ontologias <i>Methodology</i>	152
1.2.1. Planejamento	152
1.2.2. Especificação	153
1.2.3. Conceitualização.....	153
1.2.4. Formalização	153
1.2.5. Integração	153
1.2.6. Implementação.....	153
1.2.7. Documentação	153
1.2.8. Avaliação.....	153
1.2. Referências bibliográficas.....	154
Anexo III Modelo de Pacote de Estudos Experimentais do eSEE	155
1.1. Modelo de Pacote de Estudos Experimentais.....	155
1.1.1 Comunidade de Engenharia de Software	156
1.1.2. Documentos de Organização do Experimento	157
1.1.3. Documentos de Instrumentos	159
1.1.4. Documentos de Resultados.....	161
1.2 Referências bibliográficas.....	162

Índice de Figuras

Figura 1-1. Arquitetura do eSEE	19
Figura 3-1. Estrutura do Repositório de Conhecimento	29
Figura 2-1. Fragmento do documento distribuído para registrar sugestões de inclusão e exclusão de termos.....	32
Figura 2-2. Fragmento da Lista de Termos do Glossário	33
Figura 2-3. Hipergrafo do Glossário de Termos.....	34
Figura 2-4. Definições em diferentes línguas do termo selecionado.....	34
Figura 4-1. Relação taxonômica entre os domínios (BIOLCHINI <i>et al.</i> , 2006)	46
Figura 4-2. Relações entre subcategorias do conceito Medição (BIOLCHINI <i>et al.</i> , 2006).....	46
Figura 4-3. Processo de construção de ontologias adotado (FERNANDÉZ <i>et al.</i> , 1997)	47
Figura 4-4. Exemplo de representação de conhecimento utilizando a linguagem LINGO (MIAN, 2003).....	50
Figura 4-5. Ontologia de Pesquisa Científica do repositório de conhecimento do eSEE53	
Figura 4-6. Subontologia de Estratégia de Estudo	54
Figura 4-7 - Subontologia de Survey.....	56
Figura 4-8. Subontologia de Método de Pesquisa	58
Figura 4-9. Subontologia de Tipo de Estudo.....	59
Figura 4-10 - Ontologia do Pacote do Experimento.....	61
Figura 4-11. Subontologia de Planejamento do Experimento	64
Figura 4-12 - Subontologia de Documentos de Survey	66
Figura 5-1. Classe UML com estereótipo "conceito" transformada em classe OWL	83
Figura 5-2. Data property “hasFocoDaQualidade” e seu domínio e alcance	84

Figura 5-3. Superclasse da classe “QualidadeDoEstudo”	85
Figura 5-4. Relacionamento hierárquico entre os tipos de abordagem de pesquisa com a classe AbordagemDePesquisaCientifica	86
Figura 5-5. Classe “Estudo Primário” que contém a classe “Estratégia do Estudo”. A direita na figura está a hierarquia de classes do <i>Protégé</i> ilustrando as respectivas classes OWL.....	86
Figura 5-6. Superclasse da classe “EstudoPrimario”.....	87
Figura 5-7. Subontologia de Planejamento do Estudo	88
Figura 5-8. Regra de superclasse para a classe OWL "PlanejamentoDoExperimento" .	89
Figura 5-9. Subontologia de Documentos de Estudo de Caso	89
Figura 5-10. Regra de equivalent class para a classe "PlanejamentoDoEstudoDeCaso"	90
Figura 5-11. Metadados para a classe OWL “AbordagemDePesquisaCientifica”.....	91
Figura 5-12. Conceitos associados a tomadas de decisão na Ontologia de Pesquisa Científica	94
Figura 5-13 - Interface inicial da ferramenta.....	95
Figura 5-14 - Decisão sobre o tipo de abordagem de pesquisa científica	96
Figura 5-15 - Conceitos associados a tomadas de decisão na Subontologia Estratégia de Estudo	96
Figura 5-16 - Decisão sobre o tipo de survey.....	97
Figura 5-17 - Conceitos associados a tomadas de decisão na Subontologia de Survey. .	97
Figura 5-18 - Decisão sobre o tipo de survey	98
Figura 5-19 - Decisão sobre os tipos de design de survey, apoio ao participante e forma de distribuição de questionário	99
Figura 5-20 - Interface para seleção do documento do pacote de experimento que pode ser gerado pela ferramenta.....	99
Figura 5-21 - Interface para download do documento selecionado para geração	100
Figura 5-22 - Subontologia de Documento de Survey	100

Figura 5-23 - Exemplo de documento de plano de estudo que pode ser gerado a partir da escolha da estratégia de survey	101
Figura I-1. Ontologia de Pesquisa Científica.....	115
Figura I-2. Subontologia de Estratégia de Estudo	117
Figura I-3. Subontologia de Pesquisa-ação	118
Figura I-4. Subontologia de survey	120
Figura I-5. Subontologia de estudo de caso.....	123
Figura I-6. Subontologia de estudo de observação.....	124
Figura I-7. Subontologia de estudo controlado	129
Figura I-8. Subontologia de Método de Pesquisa.....	130
Figura I-9. Subontologia de Método Qualitativo (Parte 1).....	131
Figura I-10. Subontologia de Método Qualitativo (Parte 2).....	132
Figura I-11. Subontologia de Método Quantitativo (Parte 1).....	139
Figura I-12. Subontologia de Método Quantitativo (Parte 2).....	140
Figura I-13. Subontologia de Ambiente de Estudo	145
Figura I-14. Subontologia de estudo in vivo	146
Figura I-15. Subontologia de estudo in vitro.....	146
Figura I-16. Subontologia de Estudo in virtuo	147
Figura I-17. Subontologia de estudo in silico.....	149

Índice de Tabelas

Tabela 2-1. Principais conceitos do contexto de estudos experimentais.....	9
Tabela 3-1. Lista de termos do glossário antes da integração ao repositório do eSEE ..	35
Tabela 3-2. Bibliografia selecionada para pesquisa pelas equipes	37
Tabela 3-3. Taxonomia de defeitos e ação adotada para cada defeito identificado	38
Tabela 3-4. Síntese dos resultados da inspeção do glossário de termos.....	40
Tabela 3-5. Lista de termos incluídos ou alterados na revisão do glossário	40
Tabela 4-1. Tabela das subontologias inspecionadas por equipe	69
Tabela 4-2. Número de defeitos identificados após discriminação para os modelos de pesquisa-ação (Parte 1).....	70
Tabela 4-3. Número de defeitos identificados após discriminação para os modelos de pesquisa-ação (Parte 2).....	70
Tabela 4-4. Número de defeitos identificados após discriminação para os modelos de survey (Parte 1).....	71
Tabela 4-5. Número de defeitos identificados após discriminação para os modelos de survey (Parte 2).....	71
Tabela 4-6. Número de defeitos identificados após discriminação para os modelos de estudo de caso (Parte 1).....	72
Tabela 4-7. Número de defeitos identificados após discriminação para os modelos de estudo de caso (Parte 2).....	72
Tabela 4-8. Número de defeitos identificados após discriminação para os modelos de estudo de observação (Parte 1).....	72
Tabela 4-9. Número de defeitos identificados após discriminação para os modelos de estudo de observação (Parte 2).....	73
Tabela 4-10. Número de defeitos identificados após discriminação para os modelos de estudo controlado (Parte 1).....	73

Tabela 4-11. Número de defeitos identificados após discriminação para os modelos de estudo controlado (Parte 2).....	74
Tabela 4-12. Número de defeitos identificados após discriminação para o modelo de método qualitativo (Parte 1)	74
Tabela 4-13. Número de defeitos identificados após discriminação para o modelo de método qualitativo (Parte 2)	75
Tabela 4-14. Número de defeitos identificados após discriminação para o modelo de método quantitativo (Parte 1)	76
Tabela 4-15. Número de defeitos identificados após discriminação para o modelo de método quantitativo (Parte 2)	76
Tabela 4-16. Síntese do resultado (número de defeitos) da inspeção dos modelos conceituais	77
Tabela 5-4 - Transformação de elementos LINGO (UML) em elementos OWL	82
Tabela II-1. Conjunto de Documentos de Organização do Experimento.....	158
Tabela II-2. Informações registradas no documento de Plano do Experimento.....	158
Tabela II-3. Documentos de Instrumentos	160
Tabela II-4. Documentos de Resultados.....	162

Capítulo 1

Introdução

1. Motivação

O progresso em qualquer disciplina científica envolve a construção de modelos que possam ser testados através de estudos experimentais (BASILI *et al.*, 1999). Deste modo, realizar estudos experimentais provê um meio de testar, refinar e refutar a compreensão sobre um determinado fenômeno. Esta compreensão nos permite determinar a influência dos vários fatores sobre os efeitos de tecnologias e métodos. (JURISTO e MORENO, 2001).

Em Engenharia de Software, entretanto, muitas vezes os resultados das pesquisas ainda são baseados em opiniões pessoais e credices e não em dados objetivos (JURISTO e MORENO, 2001). O desenvolvimento de software é dependente de diversas tecnologias, muitas das quais não possuem evidências suficientes sobre potenciais benefícios, limitações, custo de implantação e riscos associados. Caso a Engenharia de Software fizesse um uso intenso e sistemático de uma abordagem baseada em evidência, seria possível caracterizar uma determinada tecnologia em uso. Através dessa caracterização seria possível indicar, com níveis razoáveis de segurança, o que funciona, o que não funciona e sob quais circunstâncias (TRAVASSOS *et al.*, 2008).

Para atender a essa finalidade, a Engenharia de Software Baseada em Evidência deve prover meios pelos quais melhores evidências provenientes da pesquisa possam ser integradas com experiência prática e valores humanos no processo de tomada de decisão, considerando o desenvolvimento e a manutenção do software. A Engenharia de Software Baseada em Evidência faz uso, essencialmente, de estudos experimentais (KITCHENHAM *et al.*, 2004).

Segundo Wöhlin *et al.* (2000), a experimentação provê de forma sistemática, disciplinada e controlada a avaliação de processos e atividades humanas. Estudos experimentais são úteis para entender as influências nos vários fatores que determinam os efeitos das tecnologias e métodos. Os resultados de pesquisas experimentais são informações chaves para muitos tipos de atividades de melhoria de software. O

conhecimento sobre o comportamento de técnicas, métodos e processos auxilia a antecipar seus impactos no processo de software e na qualidade do produto. Estudos experimentais são importantes nas pesquisas em Engenharia de Software, pois permitem avaliar e comparar novas teorias, métodos, técnicas e linguagens (JURISTO e MORENO, 2001).

Além da avaliação da tecnologia em desenvolvimento ou em uso, pesquisadores também podem aplicar a metodologia científica para identificar futuros tópicos de interesse de pesquisa em Engenharia de Software e para avaliar e interpretar a pesquisa já disponível sobre um determinado fenômeno de forma a criar generalizações.

Na área de Engenharia de Software, experimentação visa desenvolver uma base de evidência para compreensão e intervenção científica nos processos envolvidos no desenvolvimento de tecnologias de software, a fim de apoiar o uso de métodos e tecnologias de engenharia no planejamento, projeto, construção, execução, validação, manutenção, e melhoria de produtos de software.

São inúmeros os desafios na aplicação do método científico em Engenharia de Software. SHULL *et al.* (2001) descreve a problemática central que atua como origem de todos os desafios na área: executar um estudo experimental, especialmente em Engenharia de Software é difícil, consome muito tempo, gera um volume grande de informação e conhecimento cujo controle é complexo. SJØBERG *et al.* (2007) conduziram uma análise atual da área de Engenharia de Software Experimental e apontaram inúmeros desafios, cada qual apresentado com o estado da prática previsto para os próximos 10-15 anos. Dois desafios apontados em (SJØBERG *et al.*, 2007) são de particular relevância por impactarem a qualidade dos estudos experimentais e a comparação de resultados entre estudos.

Um deles relaciona-se à utilização de estratégias de estudo, ou como também são denominados os diferentes métodos experimentais aplicáveis. A condução de estratégias de estudo demanda que diferentes questões de arranjo sejam consideradas, sendo todas específicas da estratégia escolhida. A não-observância de todos os detalhes de design pertinentes a uma dada estratégia pode introduzir ameaças à validade do estudo, comprometendo sua qualidade (SJØBERG *et al.*, 2007). Embora a qualidade de estudos baseados na estratégia de estudo controlado tenha melhorado nos últimos anos, as habilidades dos pesquisadores em conduzir estudos baseados em *survey*, estudo de caso

e pesquisa-ação ainda apresentam importantes oportunidades de melhoria (SJØBERG *et al.*, 2007).

Em um cenário futuro de aproximadamente 10 anos, SJØBERG *et al.* (2007) prevê que estudos experimentais sejam mais cuidadosamente planejados, levando em consideração todos os detalhes de arranjo das diferentes estratégias de estudo. Segundo (SJØBERG *et al.*, 2007), isto será possível a partir de um melhor entendimento de todas as características das estratégias, suas vantagens e desvantagens. Entretanto, para atingir este cenário futuro e viabilizar este entendimento pelos pesquisadores, é essencial que o conhecimento acerca das características das estratégias de estudo seja externalizado formalmente e disponibilizado para a comunidade. O conhecimento a respeito das diferentes estratégias é especialmente relevante na etapa de planejamento do estudo, na qual importantes ameaças ao estudo podem ser introduzidas caso o pesquisador não possua todo esse conhecimento e este não seja disponibilizado (LOPES e TRAVASSOS, 2008).

SJØBERG *et al.* (2007) apontam ainda outro importante desafio relacionado à carência de uma terminologia comum e descrição apropriada de palavras-chave da área. Divergências terminológicas dificultam a comparação entre resultados de estudos heterogêneos e de diferentes grupos de pesquisa, uma vez que cada grupo pode utilizar termos com nomes diferentes e significados aplicáveis em contextos distintos. Sobretudo, o diálogo e a troca de conhecimento entre os pesquisadores ficam prejudicados em função desta deficiência. A condução de estudos secundários também é prejudicada por ser fortemente dependente dos termos da área para pesquisa na literatura técnica. Portanto, assim como para o desafio relacionado às estratégias de estudo, é importante definir um corpo de conhecimento que também normalize a nomenclatura da área, estabelecendo um significado único e um contexto de aplicação bem definido.

1.2. Objetivos

A solução aos dois desafios de SJØBERG *et al.* (2007) visa o avanço da pesquisa em Engenharia de Software a partir da condução de estudos com maior qualidade e cujos resultados sejam efetivamente comparáveis. Há um forte ponto em comum na solução de ambos: há a necessidade de explicitar conhecimento sobre o domínio de Engenharia de Software Experimental. Para o primeiro desafio, o

conhecimento relevante é pertinente ao subdomínio de estratégia de estudo. Já para o segundo desafio, o conhecimento requerido é muito amplo, pois deve contemplar o domínio de Engenharia de Software Experimental como um todo, em especial os conceitos que constituem palavras-chave neste domínio.

Portanto, este trabalho visa construir um repositório para este conhecimento, constituindo assim um primeiro passo no sentido propor uma solução aos dois desafios de SJØBERG *et al.* (2007). Serão explorados dois mecanismos de representação do conhecimento: glossário de termos e ontologias. O primeiro é utilizado no estabelecimento da terminologia comum na área de Engenharia de Software Experimental. Como o objetivo principal do glossário é difundir a terminologia pela comunidade para incentivar sua adoção, o mesmo foi construído na plataforma *web*, tornando-o acessível pelos pesquisadores (LOPES e TRAVASSOS, 2009).

As ontologias são empregadas para explicitar o conhecimento formal a respeito do subdomínio de estratégia de estudo. Outros subdomínios, importantes no contexto de planejamento do estudo e intimamente relacionados às estratégias, serão considerados também na construção do repositório. Os subdomínios de método de pesquisa e ambiente de estudo também possuem características relevantes na etapa de planejamento e, caso não sejam observadas no planejamento do estudo, ameaças a validade podem ser introduzidas (LOPES e TRAVASSOS, 2008), assim como no caso da não-observância dos detalhes de uma estratégia de estudo (SJØBERG *et al.*, 2007).

Constitui também objetivo adicional deste trabalho desenvolver um protótipo de ferramenta que atue como uma prova de conceito para ilustrar como o conhecimento do repositório pode ser utilizado. O cenário de utilização está na etapa de planejamento, na qual o conhecimento dos subdomínios de estratégia e ambiente de estudo e método de pesquisa são relevantes (LOPES e TRAVASSOS, 2008). A ferramenta construída obtém do repositório o conhecimento necessário para apoiar as tomadas de decisão na etapa de planejamento, disponibilizando o significado dos principais conceitos envolvidos para o pesquisador. Além disto, o próprio conhecimento sobre os pontos de decisão também será explicitado no repositório e disponibilizado pela ferramenta em forma de perguntas (com suas respectivas opções de resposta). Uma das razões para a escolha de ontologias na construção do repositório está no fato de que o conhecimento externalizado é formal e, portanto, passível de processamento automático. Neste caso, a

ferramenta construída processa a ontologia, extraindo o conhecimento decisões comuns à etapa de planejamento, incluindo conceitos e definições.

Além de auxiliar o pesquisador no processo de tomada de decisão, a ferramenta gera um *template* de documento de planejamento pré-preenchido a partir das decisões tomadas. Embora a ferramenta seja um protótipo, ela poderá ser estendida para contemplar a geração do *template* de outros tipos de documentos. Os documentos gerados e consumidos durante a execução do estudo experimental constituem o pacote do experimento (COSTA *et al.*, 2004). AMARAL (2003) definiu o conjunto de artefatos presentes no pacote, explicitando quais informações são registradas em cada documento. As ontologias do repositório também foram construídas visando formalizar o conhecimento produzido em (AMARAL, 2003) sobre o pacote do experimento. Este conhecimento é utilizado pela ferramenta na geração do *template* do documento.

1.3. Contexto: eSEE (*experimental Software Engineering Environment*)

Este trabalho está dentro do contexto de um ambiente de apoio a Experimentação em Engenharia de Software, denominado eSEE (*experimental Software Engineering Environment*). Travassos *et al.* (2008) apresentaram o eSEE visando ampliar as possibilidades de utilização e apoiar efetivamente experimentação em larga-escala em Engenharia de Software. Este ambiente consiste em uma infra-estrutura computacional baseada em serviços *web* que provê facilidades para instanciação de ambientes de apoio às atividades de definição, planejamento, execução e empacotamento de estudos em Engenharia de Software. O capítulo 2 desta dissertação apresenta em maiores detalhes a arquitetura do eSEE.

A estrutura proposta para o repositório de conhecimento foi projetada para torná-lo utilizável de forma independente pelo pesquisador, para que possa consultá-lo, ou de forma integrada ao eSEE, apoiando a execução de estudos experimentais. Neste caso, ferramentas inteligentes podem ser implementadas e disponibilizadas como facilidades no eSEE, contribuindo para recuperar conhecimento do repositório e fornecê-lo ao pesquisador. O protótipo construído e apresentado neste trabalho constitui um exemplo destas ferramentas e atua como uma prova de conceito.

1.5. Metodologia e Atividades

Anteriormente a este trabalho, o Grupo de Engenharia de Software Experimental da COPPE/UFRJ (<http://ese.cos.ufrj.br>) já trabalhava na construção de um glossário de termos na área. Desenvolveu-se uma ferramenta web utilizando a infra-estrutura wiki (MEDIAWIKI, 2010) para disponibilizar a lista de termos para a comunidade. Entretanto, no domínio de conhecimento de Engenharia de Software Experimental, os conceitos pertinentes aos subdomínios de estratégia e ambiente de estudo e método de pesquisa ainda não haviam sido amplamente contemplados pelo glossário. Portanto, fez-se necessário conduzir uma pesquisa em referências para revisar a versão corrente do glossário e capturar um novo conjunto de conceitos relacionados a esses subdomínios. Esta é a primeira etapa da metodologia empregada. Abaixo, segue um resumo das principais etapas da metodologia, ressaltando algumas atividades envolvidas.

1. Adequação do glossário de termos aos subdomínios de estratégia de estudo, ambiente de estudo e método de pesquisa (revisão individual);
2. Construção da versão inicial das ontologias, tendo como base a Ontologia de Pesquisa Científica de BIOLCHINI *et al.* (2006). A construção seguiu o processo denominado *Methodology* (FERNANDÉZ *et al.*, 1997);
3. Construção da Ontologia de Pacote do Experimento, a fim de representar o conhecimento sobre os artefatos do experimento conforme definido em AMARAL (2003);
4. Revisão em grupos do glossário de termos (inspeção);
5. Revisão individual das ontologias (inspeção);
6. Adequação do glossário e das ontologias frente às oportunidades de melhoria apontadas na revisão dos mesmos;
7. Construção de protótipo de ferramenta como prova de conceito para apoiar a etapa de planejamento a partir da consulta do repositório de conhecimento. A ferramenta é disponibilizada como uma facilidade na infra-estrutura eSEE.

1.6. Síntese de resultados

Dentre os resultados alcançados neste trabalho, ressaltamos as seguintes contribuições:

- Formalização em ontologias do conhecimento de Engenharia de Software Experimental, em especial do conhecimento pertinente aos subdomínios;

- Adequação do glossário de termos aos conceitos dos subdomínios supracitados;
- Revisão do conteúdo do glossário e das ontologias;
- Construção de uma ferramenta que ilustra como o conhecimento pode ser extraído do repositório. Neste protótipo, o conhecimento é disponibilizado ao pesquisador para apoiar tomadas de decisão do pesquisador durante a etapa de planejamento. A ferramenta também gera um *template* do documento de plano do estudo.

1.7. Organização

Além do capítulo de introdução, este trabalho possui mais cinco capítulos. No próximo capítulo será discutida Experimentação em Engenharia de Software, enfatizando seus principais conceitos e detalhando a arquitetura do eSEE. No Capítulo 3 é apresentado em detalhes o glossário de termos de Engenharia de Software Experimental, assim como o planejamento e execução de sua avaliação. No Capítulo 4 são descritas as ontologias do repositório, o processo de avaliação do conteúdo inicial do repositório. Seu conteúdo foi avaliado a partir de inspeções. O Capítulo 5 apresenta um protótipo de ferramenta inteligente que recupera conhecimento do repositório, disponibilizando-o ao pesquisador durante a etapa de planejamento para apoiar tomadas de decisão típicas desta etapa. Além disto, a ferramenta gera um *template* do documento de plano do experimento devidamente preenchido com as respostas às decisões tomadas. Ao final é apresentada uma conclusão, com contribuições esperadas e trabalhos futuros.

O ANEXO I apresenta em detalhes os principais modelos ontológicos do repositório, descrevendo conceitos e relacionamentos. O ANEXO II descreve o processo adotado para construir as ontologias. O ANEXO III contém a descrição do modelo de pacote de experimento de AMARAL (2003), que foi explicitado nas ontologias para ser utilizado na geração do *template* do documento de plano de estudo na ferramenta desenvolvida.

Capítulo 2

Experimentação em Engenharia de Software

Este capítulo apresenta os conceitos básicos utilizados em Experimentação, e descreve as abordagens e estratégias experimentais. Discorre também sobre o eSEE, um ambiente computacional de apoio à Experimentação em Engenharia de Software no qual a proposta deste trabalho está inserida

2.1. Introdução

O progresso em qualquer disciplina envolve construir teorias que possam ser testadas, a fim de verificar se o entendimento atual do campo é correto (BASILI *et al.*, 1999). Experimentação e medição são indispensáveis para o avanço da ciência (PFLEEGER, 1999). Segundo TICHY (1998), os diferentes assuntos abordados por questões de pesquisa não representam obstáculos para aplicação do método científico. Seja o processo de desenvolvimento de software (como na Engenharia de Software), ou seja o processo de geração de energia (como na Física), ambos podem ser objeto de estudo em Experimentação. Experiência, intuição e especulação não são fontes confiáveis de conhecimento.

No contexto de Engenharia de Software, a adoção de experimentação como forma de gerar evidências é crucial no cenário de pesquisa atual, no qual grande parte das decisões ainda se pode observar ser baseada em opiniões pessoais e credices, e não em dados objetivos (JURISTO e MORENO, 2001). Assim, para contornar essa falta de evidências, Engenheiros de Software devem observar fenômenos, formular teorias e colocá-las à prova. Assim, é possível construir um arcabouço de conhecimento sobre tecnologias baseado em evidências concretas, e não em intuições, julgamentos ou opiniões.

Desta forma, cresce entre os pesquisadores na comunidade de Engenharia de Software a consciência de que estudos experimentais são cruciais no desenvolvimento e aprimoramento de processos, métodos e tecnologias para o desenvolvimento e manutenção de software (SJØBERG *et al.*, 2005). Em razão da forte influência de fatores humanos em Engenharia de Software (WÖHLIN *et al.*, 2000), a Experimentação

pode contribuir no sentido de avaliar esta influência em processos de software. Além disto, a pesquisa experimental em Engenharia de Software é capaz de elucidar o conhecimento sobre: (1) qual tecnologia é útil para realizar determinadas tarefas; (2) quem são os mais indicados para realizar essas tarefas; (3) quais são os ambientes nos quais esta tarefa pode ser conduzida (SJØBERG *et al.*, 2005).

Neste capítulo são apresentados conceitos básicos relacionados à Experimentação. São discutidos os métodos de pesquisa aplicados à Experimentação e apresentadas algumas taxonomias de classificação de estudos experimentais com seus respectivos exemplos. Ao final apresentamos uma visão geral da representação formal de conhecimento, a ser empregado no trabalho apresentado nesta dissertação.

2.2. Conceitos Básicos

Um **estudo experimental** é uma atividade com o propósito de descobrir algo desconhecido ou de testar uma hipótese envolvendo uma investigação de coleta de dados e de execução de uma análise para determinar o significado dos dados. Isto cobre várias formas de análise e estratégias de pesquisa (BASILI *et al.*, 1999).

Experimento são estudos experimentais que se colocam à prova uma determinada hipótese através da manipulação de variáveis independentes, que repercutem no valor de variáveis dependentes. São considerados experimentos os experimentos controlados e *quasi*-experimentos, que serão abordados nas próximas seções (AMARAL, 2003; SJØBERG *et al.*, 2005). No contexto desta dissertação, o significado de estudo experimental é tratado como sendo mais amplo que o de experimento, uma vez que pode contemplar diferentes estratégias de estudo. AMARAL (2003) e CHAPETTA (2006) reuniram os principais conceitos, com sua respectiva definição, empregados no contexto de experimentos científicos. Abaixo estão relacionados os conceitos mais importantes conforme os trabalhos de AMARAL (2003) e CHAPETTA (2006).

Tabela 2-1. Principais conceitos do contexto de estudos experimentais

Conceito	Significado
Variável independente	Denominada variável de entrada ou fator, é manipulada pelo pesquisador para que sejam observadas as repercussões desta manipulação no processo ou sistema.
Variável dependente	Também denominada resultado, é medida pelo pesquisador

	para avaliar o impacto das alterações nas variáveis independentes.
Variável de contexto	É um valor particular de um fator. O tratamento é aplicado à combinação de objetos e participantes (WÖHLIN <i>et al.</i> , 2000).
Tratamento	Para cada variável independente do experimento, o pesquisador define uma faixa de valores, que serão atribuídos à variável durante a execução do estudo. Cada combinação de valor e fator é chamada de tratamento.
Teoria	É uma possível explicação de algum fenômeno. Qualquer teoria é composta por um conjunto de hipóteses.
Hipótese	De acordo com PFLEEGER (2001), uma hipótese é uma tentativa teórica ou uma suposição do que se pensa, e que explica o comportamento do que se pretende explorar. Segundo PERRY <i>et al.</i> (2000) as hipóteses definem as questões de pesquisa que esta procura responder. Deve-se pensar no estudo como um procedimento para fazer uma comparação. Desta forma, inicia-se em alto nível com questões abstratas e refina-se até chegar a um nível de questões concretas. As hipóteses abstratas são declarações em linguagem natural, geralmente utilizadas nos termos diários, como por exemplo: “reunião é uma parte indispensável no processo de inspeção”. As hipóteses concretas são utilizadas em termos do projeto de estudo, como por exemplo: “equipes que aplicam inspeções com reuniões encontram mais defeitos do que equipes que aplicam inspeção sem reuniões”.
Hipótese nula	A hipótese principal de um experimento se chama hipótese nula e declara que não há nenhum relacionamento estatisticamente significativo entre a causa e o efeito. O objetivo principal do experimento é, então, rejeitar a hipótese nula a favor de uma ou algumas hipóteses alternativas. A decisão sobre rejeição da hipótese nula pode ser tomada

	baseada nos resultados da sua verificação utilizando um teste estatístico. É aceito que a hipótese nula, empregada por conveniência para enunciar a hipótese de um experimento, não possa ser aceita pelos dados obtidos; podendo apenas ser rejeitada ou deixar de ser rejeitada.
Objeto de estudo	Ferramenta usada para verificar o relacionamento causa-efeito numa teoria. Durante a execução do experimento, os tratamentos estão sendo aplicados a um conjunto de objetos de estudo e assim os resultados estão sendo avaliados.
Instrumentação	A instrumentação provê meios de executar o estudo experimental e monitorá-lo, sem afetar seu controle. Os resultados devem ser os mesmos independentemente de como o estudo experimental é instrumentado. Se a instrumentação afeta a saída do estudo experimental, então os resultados são inválidos (AMARAL, 2003).
Participantes	Selecionados a partir da população de interesse para conduzir o experimento. Para generalizar os resultados de um experimento a uma população desejada, o conjunto de participantes deve ser representativo para aquela população. Para atingir esta meta, parâmetros que influem no resultado do experimento, como o modo de seleção dos participantes e o tamanho do conjunto selecionado, devem ser considerados. A princípio, quanto maior é a variedade da população tanto maior deve ser o tamanho do conjunto de participantes
Teste experimental	A combinação dos objetos, participantes e tratamentos se chama teste experimental ou <i>trial</i> .

2.3. Técnicas de Investigação Científica

Com o propósito de validar teorias da ciência da computação, para que melhorias nos processos e produtos da Engenharia de Software sejam obtidas e disponibilizadas para a indústria, técnicas de investigação científica devem ser aplicadas. Essas técnicas podem ser agrupadas em quatro categorias (ZELKOWITZ *et al.*, 2003):

- **Método científico:** consiste em cientistas que desenvolvem uma teoria para explicar um fenômeno, propõem uma hipótese e testam variações alternativas da hipótese. Ao testar variações da hipótese, dados são coletados a fim de aceitar ou rejeitar a hipótese, como por exemplo, um modelo de simulação;
- **Método da engenharia:** engenheiros desenvolvem e testam uma solução para uma hipótese. Baseado nos resultados do teste, os engenheiros melhoram a solução até que não sejam mais necessárias melhorias;
- **Método experimental:** um método estatístico é proposto como sendo uma forma de validar uma determinada hipótese. Diferente do método científico, pode não haver um modelo formal ou uma teoria descrevendo as hipóteses. Dados são coletados a fim de verificar a hipótese, como por exemplo, estudos de caso ou experimentos;
- **Método analítico:** uma teoria formal é desenvolvida e resultados derivados desta teoria podem ser comparados com as observações experimentais.

O método de engenharia e o método experimental podem ser vistos como variações do método científico (AMARAL, 2003). Tradicionalmente o método analítico é usado nas áreas mais formais da engenharia elétrica e da ciência da computação. O método científico é usado em áreas aplicadas, tal como a simulação de uma rede de telecomunicações com o propósito de avaliar o seu desempenho.

O método de engenharia é amplamente empregado na indústria. Estudos experimentais tradicionalmente têm sido utilizados nas ciências sociais e na psicologia. Nestas áreas, eles estão relacionados com o comportamento humano. O ponto importante neste contexto é que a engenharia de software é bastante relacionada com o comportamento humano já que o desenvolvimento de software se dá através da participação ativa de desenvolvedores (WÖHLIN *et al.*, 2000). O propósito comum destes métodos é a coleta de dados do processo de desenvolvimento de software, do próprio produto, ou de ambos.

2.4. Classificação de Estudos Experimentais

Diferentes disciplinas e pesquisadores dentro de uma mesma disciplina atribuem diferentes nomes para um mesmo tipo de estudo. Em particular na Engenharia de Software, há diferentes perspectivas sob as quais estudos experimentais são classificados.

É apresentada a seguir a classificação de estudos experimentais utilizada nesta dissertação. Os estudos são classificados quanto à abordagem de pesquisa utilizada, sendo caracterizados como **estudos primários, secundários e terciários**.

2.4.1. Estudos Primários

Estudos primários em Engenharia de Software são aqueles que visam a caracterização de uma determinada tecnologia em uso dentro de um contexto específico (WÖHLIN *et al.*, 2000). Nessa categoria podem ser utilizadas diferentes perspectivas para classificar de maneira mais específica os estudos primários, estabelecendo assim outras taxonomias. São abordadas a seguir taxonomias que classificam estudos primários quanto à estratégia do estudo, ao método de pesquisa empregado e ao tipo de ambiente e participante do estudo.

2.4.1.1. Classificação quanto à Estratégia de Estudo

A literatura atribui nomes específicos aos estudos experimentais de acordo com seus propósitos e condições a serem investigadas. Visando organizar esta classificação, COSTA *et al.* (2004) propõe a classificação de estudos primários quanto à sua estratégia. Nesta taxonomia, os tipos de estudos são survey, estudo de caso e estudo controlado. Outras estratégias de estudo presentes na literatura podem ser acrescidas a esta classificação: pesquisa-ação (SANTOS e TRAVASSOS, 2008) e estudo de observação (DALY *et al.*, 1997; ROBINSON *et al.*, 2007).

- **Pesquisa-ação** (ou Action Research): objetiva-se resolver uma situação social de um domínio específico do mundo real, através da participação intervencionista e colaborativa de pesquisadores junto aos demais envolvidos no problema. Busca-se reunir a prática e a pesquisa, ação e reflexão para promover melhorias na situação e contribuir para o conhecimento científico (SANTOS e TRAVASSOS, 2008).
- **Survey** (ou Pesquisa de Opinião): é uma investigação realizada em retrospecto, quando, por exemplo, uma determinada tecnologia tem sido utilizada durante certo período de tempo (KITCHENHAM *et al.*, 2008). Um survey permite capturar um “retrato instantâneo” da situação atual. Através de questionários ou entrevistas, dados são coletados de uma amostra de pessoas que representam um grupo específico. Os resultados do survey são então analisados de forma a

extrair conclusões que possam ser generalizadas para a população da qual a amostra foi tomada (MENDONÇA, 2004).

- **Estudo de Caso:** utilizado para monitorar projetos ou atividades onde dados são coletados para um objetivo definido. Estudos de caso são usados, principalmente, para a monitoração de atributos presentes em projetos, atividades ou atribuições. Durante a condução de um estudo de caso, dados são coletados e, com base na coleta desses dados, análises estatísticas são conduzidas de forma a permitir-se avaliar um determinado atributo ou o relacionamento entre diferentes atributos (WÖHLIN *et al.*, 2000).
- **Estudo de observação ou observacional:** nestes estudos o investigador não expõe objetos ou participantes a nenhum tratamento. O propósito de um estudo de observação pode ser de caracterizar, definir um baseline e/ou identificar relacionamentos. Os dados podem ser coletados através de observação visual, entrevistas e formulários de coleta de dados, ou pode ainda utilizar dados históricos de um repositório (DALY *et al.*, 1997).
- **Estudo Controlado:** realizado, normalmente, em laboratórios, com o objetivo de manipular uma ou mais variáveis e controlar as demais a níveis fixos ou determinados. Em um estudo experimental, os participantes são atribuídos a diferentes tratamentos de forma aleatória. O objetivo é manipular uma ou mais variáveis e controlar todas as outras variáveis num valor fixo. O efeito da manipulação das variáveis é, então, medido e, com base nessa medição, análises estatísticas são conduzidas (WÖHLIN *et al.*, 2000). Uma variação de estudos controlados é o Quasi-experimento. Nestes estudos não há atribuição aleatória dos tratamentos aos participantes. Um ou mais tratamentos podem ser aplicados a dois ou mais grupos, e os grupos são comparados diretamente. A maior restrição é que os investigadores geralmente são incapazes de selecionar e atribuir de forma aleatória os participantes ou projetos às condições. Conseqüentemente, amostras representativas da população sob estudo não são prováveis de serem conseguidas e comparações não podem ser feitas nas bases de grupos equivalentes sem alguma forma de manipulação de estudo. Dependendo das restrições, um quasi-experimento pode ser conduzido em condições representativas, onde o nível de controle disponível é mínimo, ou ainda nas condições mais controladas de um laboratório (DALY *et al.*, 1997).

2.4.1.2. Classificação quanto ao Método de Pesquisa Empregado

Dentre os métodos de pesquisa aplicados em estudos experimentais em Engenharia de Software temos a pesquisa qualitativa, pesquisa quantitativa (EASTERBROOK *et al.*, 2008) e semi-quantitativa (OGBORN e MILLER, 1994). Estudos primários podem ser classificados de acordo com método de pesquisa que empregam, que são:

- **Estudo qualitativo:** Empregam o método de pesquisa qualitativo, que buscam focar na compreensão do fenômeno de pesquisa em seu contexto de ocorrência natural. De um modo geral, as principais técnicas aplicadas na Engenharia de Software são a observação de participantes, entrevistas e análise de artefatos (EASTERBROOK *et al.*, 2008).
- **Estudo quantitativo:** Empregam o método de pesquisa quantitativo, que se baseia na representação numérica de observações com o propósito de descrever e explicar um fenômeno. A pesquisa quantitativa inicia-se com a coleta de dados, seguida da aplicação de vários métodos estatísticos descritivos e de inferência (EASTERBROOK *et al.*, 2008). Uma vantagem desta abordagem é permitir comparações e análises estatísticas (WÖHLIN *et al.*, 2000).
- **Estudo semi-quantitativo:** Empregam o método de pesquisa semi-quantitativo, que procura entender o comportamento de um sistema a partir de relações causais entre as variáveis que o descrevem. Modelos semi-quantitativos permitem expressar o que é conhecido sem fazer suposições inapropriadas, simulando faixas de comportamento e não valores pontuais (WIDMAN, 1989). Este tipo de estudo é aplicado em cenário no qual não se tem conhecimento dos valores numéricos nas relações matemáticas que regem as mudanças de um sistema. Neste contexto a direção da mudança é conhecida, mas não o tamanho de seu efeito (OGBORN e MILLER, 1994).

A pesquisa qualitativa pode ser utilizada junto com a pesquisa quantitativa para se alcançar uma melhor compreensão sobre as causas de um fenômeno social ou ajudar a criar novas questões de pesquisas. É possível investigar o mesmo tópico com pesquisa quantitativa e qualitativa, porém cada uma delas irá abordar um tipo diferente de questão. A pesquisa quantitativa é apropriada quando se deseja testar os efeitos de variações em uma variável, enquanto uma pesquisa qualitativa é apropriada para

descobrir porque os resultados de uma pesquisa quantitativa são como eles se apresentam (WÖHLIN *et al.*, 2000).

2.4.1.3. Classificação quanto ao Ambiente do Estudo

TRAVASSOS e BARROS (2003) propõem uma taxonomia para estudos primários que considera características relacionadas ao ambiente no qual o estudo é executado, o tipo de participantes envolvidos e a forma de interação dos mesmos durante a condução do estudo. Esta taxonomia classifica os estudos como *in vivo*, *in vitro*, *in virtuo* e *in silico*.

- **Estudos *in vivo*:** são aqueles que envolvem as pessoas em seus próprios ambientes. Tais estudos experimentais ocorrem durante o processo de desenvolvimento em circunstâncias reais. Apesar da necessidade de resultados experimentais, é difícil para muitas organizações de desenvolvimento de software adquirir tais resultados em ambientes de produção (*in vivo*) pelo fato de os estudos experimentais apresentarem riscos e custos elevados para sua realização;
- **Estudos *in vitro*:** são aqueles executados e controlados em ambientes tais como laboratórios ou comunidades controladas. Nestes estudos, pesquisadores criam situações próximas às reais em ambientes preparados (“laboratórios”) para observar como os participantes executam uma determinada tarefa. Neste contexto, um ambiente preparado é aquele em que algumas variáveis podem ser controladas ou, pelo menos, observadas. Em Engenharia de Software, a maioria dos estudos *in vitro* é realizada em universidades ou entre grupos selecionados de uma organização de desenvolvimento de software;
- **Estudos *in virtuo*:** a expressão *in virtuo* é um neologismo que representa um estado entre *in vivo* e *in vitro*. Um estudo experimental *in virtuo* é conduzido em um ambiente virtual, composto por modelos computacionais que são manipulados por pessoas. Um modelo numérico é uma representação computacional de elementos ou fenômenos do mundo real. O comportamento do modelo é analisado através de simulação. Em estudos experimentais *in virtuo*, um modelo computacional representa o objeto em estudo e, em alguns casos, o ambiente onde o estudo experimental é realizado. As pessoas que interpretam os resultados obtidos pela simulação do modelo podem manipulá-lo através de um

programa de computador. Na engenharia de software os estudos experimentais *in virtuo*, normalmente, são realizados em universidades e laboratórios de pesquisa e são caracterizados por pequenos grupos de pessoas manipulando simuladores, e;

- **Estudos *in silico***: são aqueles em que, além do objeto e do ambiente a serem estudados, o comportamento dos indivíduos envolvidos também é descrito por modelos computacionais. Neste caso, o ambiente é completamente composto por modelos numéricos onde nenhuma interação humana é, à princípio, esperada. Devido à necessidade de conhecimento, os estudos *in silico* ainda são pouco comuns na Engenharia de Software sendo limitados a áreas onde a participação de indivíduos não representa um aspecto importante para o estudo experimental ou quando agentes inteligentes podem substituir os indivíduos.

2.4.2. Estudos Secundários

Estudos secundários pretendem identificar, avaliar e interpretar todos os resultados relevantes a um determinado tópico de pesquisa, fenômeno de interesse ou questão de pesquisa (KITCHENHAM, 2004). Revisões Sistemáticas (KITCHENHAM, 2004; BIOLCHINI *et al.*, 2005) são tipos de estudos secundários.

Uma Revisão Sistemática é um meio de identificar, avaliar e interpretar toda pesquisa disponível e relevante sobre uma questão de pesquisa, um tópico ou um fenômeno de interesse (KITCHENHAM, 2004). Consiste de uma metodologia científica específica que vai além de uma revisão informal. Uma Revisão Sistemática busca integrar pesquisa experimental a fim de criar generalizações. Este empreendimento de integração envolve objetivos específicos, que permite ao pesquisador analisar criticamente os dados coletados, resolver conflitos detectados no material da literatura e identificar futuras questões para pesquisa.

O ponto de partida para uma Revisão Sistemática é a definição de um protocolo de revisão que especifique a questão central da pesquisa e os métodos que serão utilizados para executar a revisão. Os métodos usados na revisão devem ser baseados em uma estratégia de pesquisa definida que busque identificar o maior número possível de material relevante na literatura.

Os estudos individuais analisados em uma Revisão Sistemática são os estudos primários (KITCHENHAM, 2004). É importante que o protocolo de uma revisão sistemática contenha critérios de inclusão e exclusão explícitos para avaliar cada estudo

primário potencial e para especificar a informação a ser obtida de cada estudo deste tipo.

2.4.3. eSEE (*experimental Software Engineering Environment*)

O eSEE consiste em uma infra-estrutura computacional baseada em serviços *web* e capaz de instanciar ambientes para gerenciar o processo de experimentação em Engenharia de Software, incluindo as atividades de definição, planejamento, execução e empacotamento de estudos primários e secundários. O eSEE disponibiliza processos de experimentação, pacotes da experiência, padrões de apresentação de dados, ferramentas e serviços para os diferentes tipos de estudos identificados em Engenharia de Software TRAVASSOS *et al.* (2008).

Trabalhos como (DIAS NETO *et al.*, 2004), (CHAPETTA *et al.*, 2004) e (CHAPETTA *et al.*, 2006) contribuíram na elicitação de requisitos e estruturação do eSEE, para o qual AMARAL (2003) definiu uma base com objetivo de armazenar o conhecimento produzido ao longo da execução do Processo de Experimentação. Em Chapetta *et al.* (2004) foram definidas algumas pré-condições para a definição de um ambiente que pudesse apoiar a Experimentação em Engenharia de Software, e são elas:

- Integrar ferramentas de suporte a experimentação, atuando de forma similar a um Ambiente de Desenvolvimento de Software (ADS);
- Ser um sistema *web* – para permitir sua utilização em diferentes localidades, incluindo pesquisadores de diferentes instituições;
- Ter arquitetura baseada no paradigma de serviços *web*. Segundo (KIM *et al.*, 2002), serviços *web* são serviços disponíveis sobre uma rede (Internet ou Intranet), incluindo aplicações, recursos de computação, processos e informação, onde usuários e/ou aplicações devem ser capazes de buscá-los, localizá-los e invocá-los, em uma infra-estrutura distribuída;
- Possuir mecanismos que apoiem a Gestão de Conhecimento – devido a Experimentação em Engenharia de Software ser uma atividade intensiva em conhecimento (BASILI *et al.*, 1999; SHULL *et al.*, 2004).

Com base nessas pré-condições, o Grupo de Engenharia de Software Experimental da COPPE/UFRJ (<http://ese.cos.ufrj.br>) partiu para o detalhamento inicial de quais seriam as características desejáveis para o Meta-Ambiente eSEE e, assim, capturar os requisitos desejáveis (CHAPETTA *et al.*, 2004). Para identificar tais

características foi realizada uma análise das atividades de acompanhamento, execução e empacotamento de experimentos em Engenharia de Software conforme as propostas de WÖHLIN *et al.* (2000) e AMARAL (2003).

A partir da definição desses requisitos foi construída uma arquitetura inicial para o eSEE (DIAS NETO *et al.*, 2004). A arquitetura definida é composta por três macro-componentes, de acordo com um agrupamento das características desejáveis na infraestrutura. Esses macro-componentes são o Meta-Configurador, o Ambiente de Instanciação e o Ambiente de Execução (CHAPETTA *et al.*, 2005) (Figura 1-1).

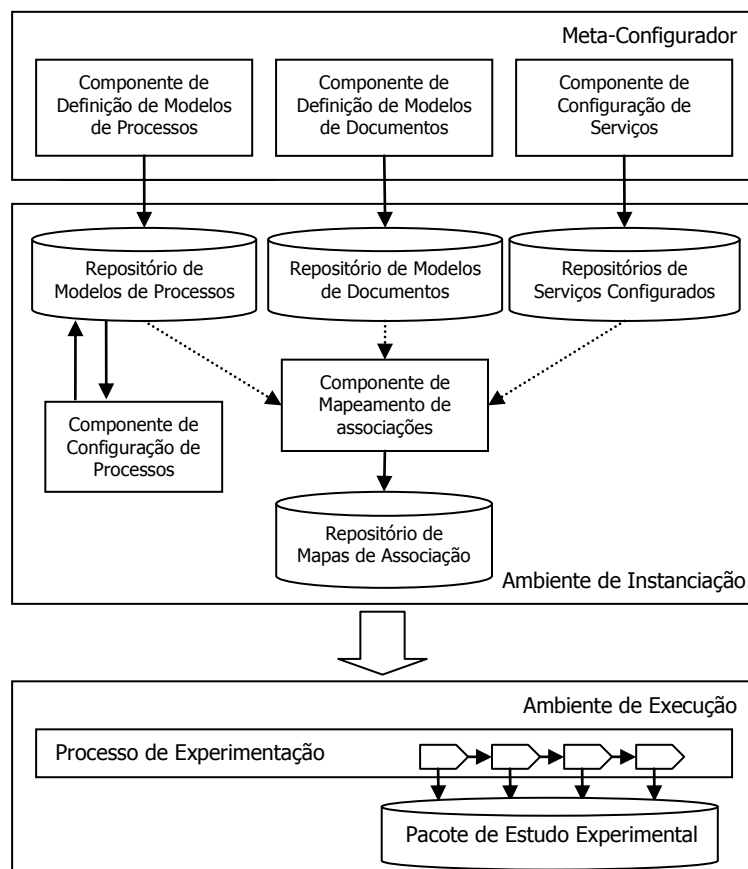


Figura 2-1. Arquitetura do eSEE

O Meta-Configurador possui como funcionalidade definir e configurar todos os elementos básicos que serão utilizados pelos dois outros componentes. Esses elementos básicos são:

- Modelos de documentos: são descrições que servem como base para instanciação de artefatos produzidos ao longo do Processo de Experimentação. O Meta-Configurador descreve modelos de documentos a partir dos modelos de documentos estabelecidos em (AMARAL, 2003) e que constituem o pacote do experimento;

- Modelos de configuráveis do processo: serve como base para a definição e instanciação de Processos de Experimentação. O Meta-Configurador descreve modelos de processos de software configuráveis a partir de um meta-modelo de processos pré-estabelecido. Entende-se como sendo um modelo configurável de processo aquele que possui:
 - Alguma atividade opcional, ou;
 - Alguma atividade que não faz parte do processo padrão, ou;
 - Alguma ferramenta opcional, ou;
 - Algum papel opcional, ou;
 - Alguma pergunta.
- Serviços Configurados: são informações relativas à utilização de serviços web que possam apoiar diferentes atividades do Processo de Experimentação. Esses serviços web, ou simplesmente serviços, disponibilizados têm de ser capazes de ter seus usos customizados para serem disponibilizados na infra-estrutura do eSEE.

Note que estes elementos são apenas definidos, mas nenhum relacionamento entre esses conceitos é descrito. Logo, nesse componente estamos somente definindo o tipo, organização e granularidade das informações de cada um desses conceitos. Para cada conceito existe um subcomponente que o define: o componente de Definição de Modelos de Processos, componente de Definição de Modelos de Documentos e o componente de Configuração de Serviços. Depois de descritos, os conceitos são armazenados em repositórios específicos do Ambiente de Instanciação.

O segundo componente, o Ambiente de Instanciação, possui como objetivo prover meios para customizar e instanciar um ambiente que permita a execução de um Processo de Experimentação. Neste ambiente, um subcomponente de configuração de processos é responsável por interpretar um Modelo Configurável de Processo, apresentar perguntas ao Pesquisador, aplicar as ações de acordo com as respostas fornecidas pelo Pesquisador, permitir que o Pesquisador aplique manualmente uma ou mais ações complementares no Modelo Configurável de Processo e publicar um novo Modelo de Processo. Essas perguntas estão definidas no modelo configurável de processo e envolvem basicamente tomadas de decisão a respeito de inclusão (ou não) de atividades, ferramentas e papéis opcionais ao processo. O Modelo de Processo gerado é

persistido em um repositório específico do Ambiente de Instanciação, de modo que possa reaproveitado em outro estudo experimental.

Baseado em um modelo configurado de processos, um modelo de documentos e os serviços configurados disponíveis definidos no componente anterior, o Ambiente de Instanciação relaciona os artefatos a serem consumidos/produzidos às atividades do Processo de Experimentação, bem como os possíveis serviços configurados a serem disponibilizados na realização de uma determinada tarefa. Para tal, as seguintes ações são realizadas:

- São definidos os relacionamentos entre o modelo do processo a ser seguido, o modelo de documentos que vai ser instanciado e os serviços configurados disponíveis para execução de um estudo experimental. Este conjunto de relacionamentos é denominado Mapa de Associação.
- Baseando-se nos relacionamentos descritos entre os elementos básicos da arquitetura eSEE, este ambiente instancia um Ambiente de Execução. Nesse momento, o modelo de processo e o modelo de documento são utilizados para instanciar um processo de Experimentação e um Pacote de Documentos, respectivamente. Os serviços configurados ficam disponíveis para serem invocados nas atividades do processo instanciado de acordo com os relacionamentos estabelecido em um determinado Mapa de Associação.

Cada mapa de Associação é então armazenado em um repositório específico, permitindo várias instanciações de uma determinada configuração de um Ambiente de Execução. O Ambiente de Execução é o componente responsável pelo acompanhamento e execução das atividades do Processo de Experimentação. O acompanhamento e controle do estudo serão realizados através do processo definido pelo modelo de processo, de acordo com as restrições de acesso às informações e entre atividades do processo. O acesso às informações de um documento é realizado através dos serviços configurados, onde os relacionamentos descritos no Mapa de Associação são responsáveis pelo link entre o artefato, uma determinada atividade do processo e o serviço configurado para realizar uma determinada tarefa. A utilização de serviços *web* é a principal maneira de anexar funcionalidades e ferramentas e, assim, customizar um Ambiente de Execução. Vale salientar que um ambiente instanciado eSEE existe sem nenhuma dependência do Ambiente de Instanciação, e que o Ambiente de Instanciação não tem nenhum controle sobre os ambientes eSEE instanciados. A mesma observação

pode ser feita entre o Ambiente Meta-Configurador e o Ambiente de Instanciação respectivamente. Deste modo o eSEE apresenta uma arquitetura de modo que seus macro-componentes podem estar geograficamente distribuídos.

2.5. Conclusão

Neste capítulo foram apresentados tópicos relacionados à Experimentação em Engenharia de Software, dentre eles:

- A definição dos principais conceitos e termos utilizados na execução de estudos experimentais em Engenharia de Software;
- Três taxonomias presentes freqüentemente na literatura e consideram características essenciais do estudo, tal como sua estratégia, o método de pesquisa adotado e o ambiente no qual é executado;
- Uma visão geral sobre o eSEE e a apresentação de sua arquitetura.

O próximo capítulo apresentará como o glossário será inserido no contexto do repositório de conhecimento e sua interdependência com as ontologias. É descrita a atividade de adequação do glossário a novos conceitos sobre estratégia de estudo, método de pesquisa e ambiente de estudo. Além disto, detalhamos o planejamento e a execução de uma inspeção conduzida para avaliar o conteúdo do glossário de termos.

Capítulo 3

Glossário de Termos de Engenharia de Software Experimental

Neste capítulo são apresentadas as principais motivações e o escopo que nortearam este trabalho. O foco deste capítulo é descrever a integração do glossário de termos ao repositório proposto, citando as adequações realizadas para ampliar sua abrangência e o processo de inspeção conduzido para avaliar seu conteúdo

3.1. Motivação

Executar estudos experimentais envolve grande volume de conhecimento consumido e produzido ao longo da execução do Processo de Experimentação (SHULL *et al.*, 2001). Neste cenário, SJØBERG *et al.* (2007) salientam três grandes desafios para ampliar a relevância de Experimentação no desenvolvimento de conhecimento científico em Engenharia de Software:

- (a) A qualidade e relevância dos estudos devem aumentar;
- (b) Deve haver maior e melhor síntese de evidências experimentais;
- (c) Mais teorias devem ser formuladas e testadas.

Segundo SJØBERG *et al.* (2007), uma das dificuldades atuais que justifica o desafio (a) está relacionada ao fato de muitos estudos ainda não analisarem amplamente o real escopo de validade da pesquisa. Este escopo compreende os atores, tecnologias, atividades e sistemas de software para os quais os resultados são aplicáveis. Muitos artigos científicos reportam estudos com escopo muito amplo em seções de ameaças à validade externa do estudo. A definição de escopos amplos pode ter como conseqüências resultados aparentemente conflitantes; aplicabilidade dos resultados comprometida por fatores de confusão; replicação e síntese de estudos comprometida (SJØBERG *et al.*, 2007).

Ainda no tocante ao desafio (a), SJØBERG *et al.* (2007) relata uma necessidade de melhorar a forma como os estudos são conduzidos, principalmente aqueles baseados em estratégias de *survey*, estudo de caso e pesquisa-ação. Os estudos reportados e que são conduzidos sob uma dessas estratégias demonstram carecer de detalhes importantes

de planejamento (SJØBERG *et al.*, 2007). YIN (2003) relata que a comunidade de pesquisa em Engenharia de Software frequentemente se refere a estudos de caso como sendo uma espécie de “*soft research*”, possivelmente por não seguirem todo o rigor exigido no planejamento de estudos baseados nessa estratégia.

SJØBERG *et al.* (2007) detalha os desafios (a) e (b) e os decompõe em quatorze desafios com propósitos específicos, capazes de afetar tanto a execução de estudos primários, quanto de secundários. Dentre esses desafios, destacam-se:

1. Adoção de estratégias de estudo considerando amplamente todos os detalhes de design das mesmas durante a etapa de planejamento;
2. Terminologia única em Engenharia de Software Experimental.

O desafio (1) ressalta que, no estado atual da prática de Experimentação em Engenharia de Software, pesquisadores aplicam estratégias de estudo ainda sem considerar amplamente todos os detalhes de design das mesmas. Características importantes das estratégias de estudo, quando não observadas, podem introduzir ameaças à validade do estudo, afetando assim sua qualidade. Além disto, a falta de uma compreensão mais consistente sobre as variedades de estratégias compromete também o entendimento sobre qual é a mais adequada para ser adotada em um estudo experimental (SJØBERG *et al.*, 2007).

O meio para enfrentar esses desafios, segundo SJØBERG *et al.* (2007), passa por aumentar a competência de pesquisadores na condução de estudos utilizando diferentes tipos de estratégias. Em linhas gerais, as formas de viabilizar esse aumento de competência incluem integrar disciplinas de Experimentação no ensino de Engenharia de Software e criar guias para condução de estudos. Em particular na criação de guias de condução, SJØBERG *et al.* (2007) ressalta a carência de um corpo de conhecimento que formalize os conceitos sobre estratégias de estudo em geral. Embora o conhecimento sobre a estratégia de estudo controlado venha sendo bem difundido nos últimos anos, ainda há necessidade de explicitar formalmente o conhecimento envolvido na condução de estudos baseados em *survey*, pesquisa-ação e estudo de caso. SJØBERG *et al.* (2007) especulam que esse desafio será vencido daqui a 10-15 anos.

O desafio (2) visa estabelecer uma terminologia comum em Engenharia de Software Experimental. Este desafio decorre das divergências de definição e aplicação entre os termos utilizados por diferentes grupos de pesquisa, o que pode dificultar a comparação de resultados heterogêneos. A dificuldade em comparar resultados impacta

diretamente a condução de estudos secundários planejados para estabelecer comparações. Ao existir diferentes definições e aplicações de termos da área, a própria comunicação entre grupos de pesquisa pode ser prejudicada, assim como a troca de conhecimento entre os mesmos. SJØBERG *et al.* (2007) especulam que a solução efetiva para este desafio virá daqui a aproximadamente 10 anos.

Para o desafio (1), o conhecimento necessário contempla o subdomínio de estratégia de estudo. Para o desafio (2), o conhecimento necessário abrange todo o domínio de Experimentação em Engenharia de Software. A previsão para vencer esse desafios, estabelecida por SJØBERG *et al.* (2007), já aponta para a elevada complexidade em tratá-los e difundir a solução por toda a comunidade de pesquisa em Engenharia de Software. Tendo em vista este cenário, este trabalho apresenta uma proposta de criação de um repositório de conhecimento de Experimentação em Engenharia de Software. Este repositório é constituído pelo glossário de termos do eSEE e por ontologias. O glossário será incorporado ao repositório de modo a enfrentar o desafio (2), que sempre foi o objetivo principal da construção do glossário. As ontologias foram construídas para explicitar formalmente o conhecimento sobre Engenharia de Software Experimental, em especial sobre estratégia de estudo. Assim, o desafio (1) é tratado.

LOPES e TRAVASSOS (2008) identificaram dois importantes subdomínios de conhecimento relacionados às diferentes estratégias de estudo: método de pesquisa e ambiente de estudo. O primeiro subdomínio contempla os métodos de análise de dados, que podem ser qualitativo, quantitativo e semi-quantitativo. O subdomínio de ambiente de estudo contém o conhecimento sobre estudos *in vivo*, *in vitro*, *in virtuo* e *in silico*. Tendo em vista a importância do conhecimento desses subdomínios no planejamento de estudos (LOPES e TRAVASSOS, 2008), a construção do repositório levou em consideração esses subdomínios.

3.2. Escopo

A estrutura proposta para o repositório de conhecimento do eSEE viabiliza sua utilização de duas formas:

- (1) Independentemente pelo pesquisador, como forma de consulta à terminologia de Experimentação em Engenharia de Software;
- (2) Integrada ao eSEE, para atuar como o núcleo de recuperação de conhecimento neste ambiente. Para ilustrar como o repositório pode ser

utilizado desta maneira, uma ferramenta foi construída como prova de conceito. A arquitetura desta ferramenta foi projetada de modo a permitir o processamento da ontologia de Pesquisa Científica e disponibilizar conhecimento ao pesquisador na etapa de planejamento. Este conhecimento inclui a definição dos principais conceitos desta etapa, além de um conjunto de decisões comuns ao planejamento, cada qual apresentada na forma de uma pergunta com suas respectivas respostas. Ao final, a ferramenta disponibiliza ao pesquisador o *template* do documento de plano de estudo gerado com base nas decisões tomadas. Este *template* é produzido através do processamento das ontologias de Pesquisa Científica e Pacote do Experimento, esta última tendo sido construída a partir do modelo de pacote de experimentos de AMARAL (2003).

O conteúdo inicial proposto para o repositório de conhecimento terá foco em:

- (1) Conhecimento geral sobre experimentação, visando oferecer uma terminologia única sobre Experimentação em Engenharia de Software;
- (2) Conhecimento sobre os domínios de abordagem, estratégia e ambiente de estudo, além de método de pesquisa, conforme identificados em (LOPES e TRAVASSOS, 2008) como sendo domínios de conhecimento importantes na atividade de planejamento de um estudo.
- (3) Conhecimento sobre o pacote do experimento, mapeado no trabalho de AMARAL (2003). Como este conhecimento é parte integrante da primeira proposta do repositório do eSEE, deve ser revisado e preservado no repositório proposto nesta dissertação.

3.3. Representação Formal do Conhecimento

Conhecimento é o recurso chave em tomadas de decisão inteligentes, previsão, projeto, planejamento, diagnóstico, análise, avaliação e julgamento intuitivo. O conhecimento pode estar embutido não somente em documentos e repositórios, mas também em rotinas, processos, práticas e normas organizacionais.

O conhecimento é classificado como de dois tipos mutuamente complementares (NONAKA e TAKEUCHI, 1995): tácito e explícito. Conhecimento tácito é pessoal, específico em um contexto e difícil de formalizar e comunicar. Conhecimento explícito é transmissível em linguagem formal e sistemática. Para o conhecimento explícito é dito

formal quando é representado utilizando uma linguagem que o torne passível de processamento computacional.

Segundo O'LEARY (1998), a infra-estrutura na qual o conhecimento deve ser armazenado é constituída de uma base de conhecimento utilizando ontologias. As bases de conhecimento armazenam fisicamente o conjunto coletivo de dados e informações através do qual conhecimento individual e organizacional é construído (PROBST *et al.*, 2000). Uma ontologia é uma especificação formal e explícita de uma conceitualização compartilhada (GRUBER, 1993), isto é, uma descrição de conceitos e relações que existem em um domínio de interesse. Basicamente, uma ontologia consiste desses conceitos e relações, e suas definições, propriedades e restrições, descritas na forma de axiomas. Por representarem formalmente o conhecimento, ontologias são capazes de apoiar recuperação de conhecimento de repositórios.

ABECKER *et al.* (1999) identificaram tecnologias empregadas para apoiar a execução de atividades que lidam com manipulação de conhecimento:

- **Groupware, Sistemas de Workflow e Trabalho Cooperativo Apoiado por Computador** apóiam a execução de tarefas que necessitam de forte interação entre os realizadores e envolvem grande volume de conhecimento;
- **Sistemas de Gerenciamento de Documentos, Filtragem e Recuperação de Informação** viabilizam a disseminação e identificação satisfatória do conhecimento entre profissionais e grupos de trabalho;
- **Inteligência Artificial** possibilita o desenvolvimento de ferramentas inteligentes de através de recursos como ontologias, *data mining* e entre outros.

ABECKER *et al.* (1999) destaca que ontologias são úteis para apoiar a especificação e implementação de qualquer sistema de computação complexo. Uma ontologia pode ser desenvolvida para vários fins, mas, de uma forma geral, os seguintes propósitos são atingidos:

- Ajuda as pessoas a compreenderem melhor certa área de conhecimento: no desenvolvimento de uma ontologia, as pessoas envolvidas no processo se vêem diante de um desafio: explicar seu entendimento sobre o domínio em questão, o que as faz refletir e melhorar sua compreensão sobre esse domínio;

- Ajuda as pessoas a atingir um consenso no seu entendimento sobre uma área de conhecimento: geralmente, em uma determinada área de conhecimento, diferentes especialistas têm entendimento diferenciado sobre os conceitos envolvidos, o que leva a problemas na comunicação. Ao se construir uma ontologia, essas diferenças são explicitadas e busca-se um consenso sobre seu significado e sua importância;
- Ajuda outras pessoas a compreender certa área de conhecimento: uma vez que haja uma ontologia sobre uma determinada área de conhecimento desenvolvida, uma pessoa que deseje aprender mais sobre essa área não precisa se reportar sempre a um especialista. Ela pode estudar a ontologia e aprender sobre o domínio em questão, absorvendo um conhecimento geral e de consenso.

3.4. Proposta de Estrutura do Repositório de Conhecimento

Segundo MONTONI (2003), uma infraestrutura para persistir conhecimento é constituída por dois elementos básicos: base de conhecimento e ontologia. A base armazena fisicamente o conhecimento, enquanto as ontologias atuam na representação e recuperação do conhecimento destas bases (O'LEARY, 1998 *appud* MONTONI, 2003). Dentre as linguagens de representação existentes, ontologias são classificadas como sendo uma das mais formais (BREITMAN, 2005).

Ao adotar ontologias como um dos mecanismos estruturantes do repositório, é possível representar formalmente o conhecimento sobre Experimentação. Assim, poderemos dar início à concretização da solução para o desafio de SJØBERG *et al.* (2007) envolvendo a externalização formal do conhecimento sobre o domínio da Engenharia de Software Experimental e seus subdomínios descritos em (LOPES e TRAVASSOS, 2008). Além disto, utilizando ontologias, o repositório poderá ser consultado a partir de ferramentas inteligentes (ABECKER *et al.*, 1999). Ferramentas deste tipo podem ser construídas para recuperar conhecimento e disponibilizá-lo, por exemplo, na etapa de planejamento, cuja execução detalhada é importante para não introduzir ameaças à validade do estudo (LOPES e TRAVASSOS, 2008).

Além de ontologias, adotamos o glossário de termos do eSEE também como um mecanismo estruturante do repositório. O estabelecimento de uma terminologia comum na área de Engenharia de Software de Software Experimental, um dos desafios descritos

em (SJØBERG *et al.* 2007), foi o objetivo primário na construção de sua primeira versão pela comunidade da ISERN (ISERN, 1995) e posteriormente em sua reorganização e evolução pelo Grupo ESE. Por esta razão, a inclusão do glossário no repositório e a disponibilização a partir da plataforma *wiki* representa mais um passo no sentido de difundi-lo na comunidade Engenharia de Software.

Entretanto, glossários são, quanto ao espectro semântico, mecanismos de representação pouco formais e de baixa expressividade (BREITMAN, 2005). Para minimizar essa deficiência, a estrutura do repositório deve tornar o glossário e as ontologias interdependentes entre si. Assim, a proposta de estrutura do repositório consiste na integração entre dois mecanismos de representação do conhecimento, que são um glossário de termos em Experimentação em Engenharia de Software e ontologias (Figura 3-1). O glossário de termos lista os conceitos e seu respectivo significado, enquanto que as ontologias expressam os relacionamentos entre esses conceitos, assim como suas propriedades. Portanto, ambos são dependentes entre si, uma vez que o glossário não define as relações entre os conceitos da lista e o significado dos conceitos nos modelos ontológicos devem estar em sintonia com o glossário.

Para integrar o glossário ao repositório, entretanto, foi necessário ampliar sua abrangência para os subdomínios selecionados. A atividade de integração, assim como as características gerais do glossário e seu histórico de construção são apresentados a seguir.

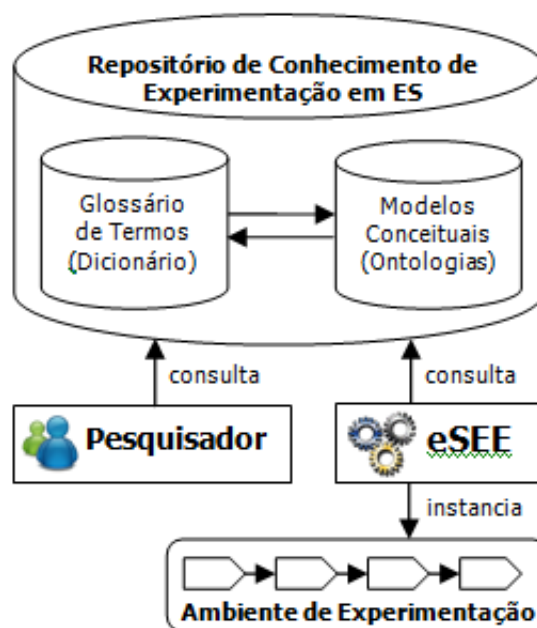


Figura 3-1. Estrutura do Repositório de Conhecimento

3.5. Glossário de Termos do eSEE

No contexto de experimentação em larga-escala, a troca de conhecimento e experiência entre os pesquisadores fica dificultada quando estes não se utilizam de um vocabulário em comum. Frequentemente, cada grupo de pesquisa adapta conceitos e definições sob suas próprias perspectivas. Isto conduz a divergências entre os pesquisadores, por vezes tornando difícil a comparação entre os resultados dos estudos e assim atrasando a evolução da área (SJØBERG *et al.*, 2007).

Visando tentar reduzir as divergências terminológicas e construir uma perspectiva em comum para a área, foi iniciado o desenvolvimento de um glossário de termos de Experimentação em Engenharia de Software. As subseções a seguir detalham a construção do glossário em parceria com diferentes pesquisadores e a integração do glossário ao repositório de conhecimento do eSEE.

3.5.1. Histórico da Construção do Glossário de Termos

A comunidade ISERN disponibilizou uma primeira versão de um glossário de termos (ISERN, 1995), elaborado a partir dos estudos usualmente executados na época e relacionados principalmente a estudos *in vitro* no domínio de inspeções de software. Desde então foi possível perceber uma demanda crescente pela utilização de experimentação em Engenharia de Software, com foco diversificado e direcionado a diferentes categorias de estudo. Desta forma, o glossário proposto inicialmente, embora adequado para uma categoria de estudos em particular, não permitia a utilização consistente dos conceitos envolvidos nos novos tipos de estudo. Assim, visando contribuir para uma maior compreensão da experimentação em Engenharia de Software, uma nova versão do glossário de termos passou a ser organizada.

Um conjunto evoluído de termos foi organizado pelo grupo ESE a partir da consolidação da terminologia básica estabelecida pelo ISERN e dos trabalhos de AMARAL (2003), COSTA *et al.* (2003) e CHAPETTA (2006). Estes trabalhos abordam conceitos básicos sobre experimentação e também específicos sobre diferentes categorias de estudo, contribuindo assim para consolidar um glossário mais contemporâneo e com abrangência satisfatória. Foi organizada uma sessão no ESELAW'06 (*Experimental Software Engineering Latin American Workshop 2006*) para apresentá-lo à comunidade. Assim, consolidou-se a primeira versão do glossário de termos de Experimentação em Engenharia de Software.

A discussão conduzida no ESELAW'06 apontou a importância de facilidades para navegação, inclusão, alteração e exclusão de conceitos e definições no uso do glossário. Os participantes ressaltaram também a facilidade de acesso como sendo uma característica determinante no incentivo à normatização terminológica na área. Tendo em vista estas perspectivas, foram identificadas e estudadas tecnologias candidatas para apoiar o desenvolvimento de uma ferramenta que tornasse o glossário mais acessível e fornecesse operações básicas de manipulação de seu conteúdo.

Após apresentação no ESELAW'06 e consolidação da primeira versão do glossário, seu conteúdo passou por sessões de revisão por diferentes grupos relacionados à Engenharia de Software Experimental. Foram conduzidas sessões de revisão em eventos científicos que reunissem pesquisadores de diferentes comunidades, representando assim uma excelente oportunidade de divulgação do glossário no meio de pesquisa e de discussões visando melhorias na completude e correteza da lista de termos.

Uma primeira revisão ocorreu na reunião do ISERN em 2007. Como uma das primeiras iniciativas de concepção do glossário veio do ISERN através da definição de uma terminologia básica, a reunião entre os membros deste grupo foi considerada para divulgar a nova versão do glossário e executar uma revisão. Em seguida, outra revisão ocorreu no ESELAW'07. Os objetivos destas revisões foram avaliar a pertinência e a correção do conteúdo do glossário.

Nestas revisões, apresentou-se o glossário e uma discussão foi promovida a fim de identificar quais termos deveriam ser incluídos e quais seriam passíveis de exclusão por não estarem ligados diretamente à experimentação em Engenharia de Software. Os participantes foram divididos em grupos que receberam um documento (Figura 2-1) listando os termos inclusos no glossário, organizados em ordem alfabética. Os participantes foram orientados a discutir em grupo quais termos poderiam ser incluídos e, dentre aqueles listados, quais seriam passíveis de exclusão.

	Current entries	Suggested entries
A	Alternate hypothesis (H1) Assertions	
B		
C	Case study confidence interval confidence limits conclusion validity construct validity control object controlled experiment critical region	
D	dependent variable direct measure dynamic analysis	
E	Experiment experimental object or unit Experimental error external validity	
F	Fact factorial design factors feasibility study Field study	

Figura 3-2. Fragmento do documento distribuído para registrar sugestões de inclusão e exclusão de termos

Os termos sugeridos por cada grupo foram comparados para eliminar eventuais duplicatas e serem efetivamente inseridos. Como as avaliações não solicitaram definições para os termos sugeridos, realizou-se uma pesquisa na literatura para fornecer definições com suas devidas referências. Como resultado, um modelo bem mais estável e cobrindo um conjunto mais abrangente de tipos de estudos foi obtido. Este resultado serviu como base para o trabalho desenvolvido nesta dissertação.

3.5.2. Características do Glossário de Termos

A primeira versão do glossário foi construída e disponibilizada em um documento textual. O aumento gradativo no número de termos, embora crucial do ponto de vista de completude, trouxe dificuldades para os pesquisadores navegarem pela lista, procurar por um termo específico e, sobretudo, controlar o acesso aos direitos de alteração no glossário. Assim, deu-se início, após consolidação da primeira versão do glossário, de um estudo sobre tecnologias disponíveis que pudessem ser empregadas no desenvolvimento de uma ferramenta que disponibilizasse o glossário. Os requisitos da ferramenta que a tecnologia deve apoiar são:

- (1) Ser composto como um sistema *web*, permitindo seu uso em diferentes localidades e entre comunidades de pesquisa;
- (2) Prover controle de acesso;
- (3) Disponibilizar operações de inclusão, alteração e exclusão de termos, definições e referências.

A tecnologia *wiki* (MEDIAWIKI, 2010) foi escolhida como sendo uma tecnologia adequada por congrega, de forma nativa, diversas facilidades para

construção de glossários. A escolha da tecnologia *wiki* se deu por conveniência. A infraestrutura é disponibilizada em um pacote de instalação contendo nativamente recursos de controle de acesso, navegação, inclusão, alteração e exclusão de termos e definições de glossários. Perfis de moderadores podem cadastrar novos usuários com diferentes permissões de alteração no glossário. Por padrão, todo usuário da ferramenta pode consultar os termos do glossário sem, entretanto, estar efetivamente cadastrado. A ferramenta está acessível através do endereço <http://lens-ese.cos.ufrj.br/wikiese>.

A ferramenta apresenta a lista de termos em ordem alfabética, na qual o usuário pode navegar facilmente entre os termos utilizando hiperlinks (Figura 2-2). A fim de adicionar uma forma de navegação diferenciada, o grupo ESE, através dos pesquisadores Felipe Pinto e Vinícios Bravo, desenvolveu um componente baseado em hipergrafo (Figura 2-3) que oferece uma nova perspectiva sobre a lista. Os termos são apresentados em um hipergrafo e ligados pelos nós rotulados pela sua inicial.

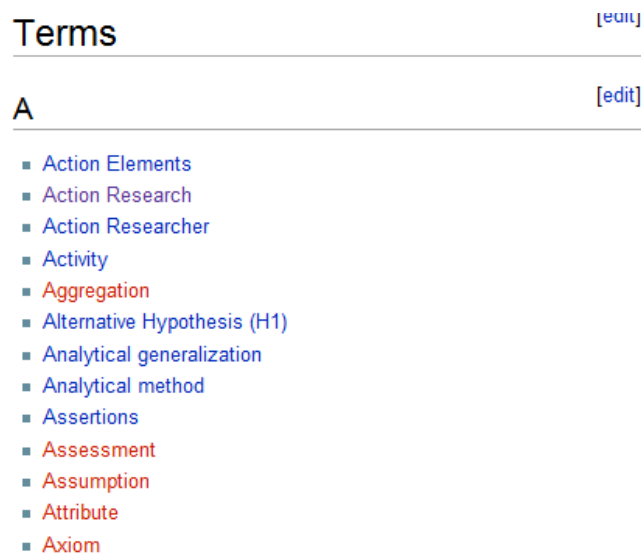


Figura 3-3. Fragmento da Lista de Termos do Glossário

Experimental Software Engineering - Glossary of Terms

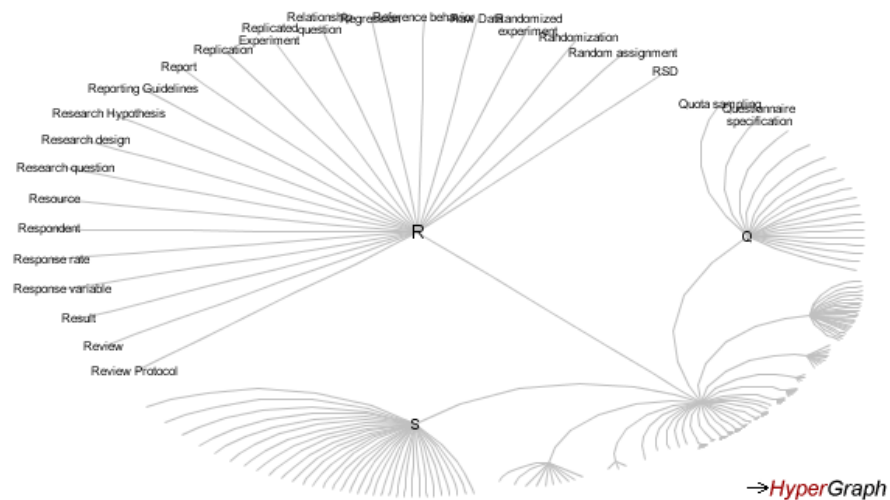


Figura 3-4. Hipergrafo do Glossário de Termos

Ao acionar o link de um dos termos da lista ou do hipergrafo, o pesquisador é conduzido à página (Figura 2-4) contendo a definição em inglês, português e espanhol do termo em questão. O recurso multilíngue se faz necessário para viabilizar a consulta do glossário por pesquisadores de diferentes nacionalidades. Além disso, são citadas as referências que fundamentam a definição e outras fontes importantes na contextualização do termo.

Definition	[edit]
English	[edit]
Case Study A case study is an empirical inquiry that investigates a contemporary phenomenon within its real-life context, especially when the boundaries between the phenomenon and context are not clearly evident (Yin, 2003). Case studies are used for monitoring projects, activities or assignments. Data is collected for a specific purpose throughout the study. It is normally aimed at tracking a specific attribute or establishing relationships between different attributes.	
Portuguese	[edit]
Estudo de Caso Um projeto é monitorado e dados são coletados. Esta coleta de dados é realizada de acordo com um objetivo específico do projeto. Um determinado atributo é monitorado (por exemplo, confiabilidade, custo) e os dados são coletados com o propósito de medir este atributo.	
Spanish	[edit]
Estudio de caso Los casos de estudio son usados para supervisar proyectos, actividades o tareas. Los datos son recogidos con un propósito específico a lo largo del estudio. Normalmente apuntan a realizar el seguimiento de un atributo específico o establecer relaciones entre diferentes atributos.	
Reference	[edit]
<ul style="list-style-type: none">▪ (Yin, 2003) 📄▪ (WÖHLIN et al, 2000).▪ (ZELKOWITZ e WALLACE, 1998a).	

Figura 3-5. Definições em diferentes línguas do termo selecionado

3.6. Integração do Glossário ao Repositório de Conhecimento

O repositório de conhecimento proposto e o glossário compartilham do objetivo de estabelecer uma terminologia única na área de Experimentação em Engenharia de Software. Ao adotar o glossário como um dos mecanismos estruturantes do repositório, a ferramenta *wiki* passa a atuar também como um mecanismo de consulta dos conceitos expressos no repositório, tornando-o acessível diretamente pelos pesquisadores. Assim, o glossário construído até então se revelou um importante mecanismo de representação de conhecimento para estruturar o repositório.

O glossário construído até então objetivou capturar conceitos gerais sobre Experimentação em Engenharia de Software e específicos no subdomínio de inspeção de software. Ao avaliar o conteúdo do glossário, observou-se uma carência de termos ligados aos subdomínios de conhecimento considerados relevantes por SJØBERG *et al.* (2007) e LOPES e TRAVASSOS (2008) no planejamento de estudos. Então, para integrar o glossário ao repositório, fez-se necessário proceder com melhorias na cobertura de conceitos acerca desses subdomínios. A tabela 3-1 reflete o estado inicial do glossário, antes de ser integrado ao repositório do eSEE.

Para consolidar a integração do glossário ao repositório, uma referência reconhecida em Engenharia de Software Experimental foi selecionada para ser pesquisada. O objetivo dessa pesquisa é agregar novos termos sobre as taxonomias previstas em (LOPES e TRAVASSOS, 2008), tornando o conteúdo do glossário mais aderente ao propósito estabelecido. A referência selecionada foi (WÖHLIN *et al.*, 2000), tendo em vista seu reconhecimento na comunidade de Engenharia de Software Experimental. Foram identificados 28 novos termos, incluídos com sua respectiva definição no glossário através da ferramenta *wiki*. Com a inclusão dos conceitos sobre os diferentes tipos de estudos previstos em (LOPES e TRAVASSOS, 2008), o glossário foi definitivamente integrado ao repositório de conhecimento.

Tabela 3-1. Lista de termos do glossário antes da integração ao repositório do eSEE

análise dinâmica	análise estática	análise qualitativa
análise quantitativa	análise teórica	assertivas
dado legado	Entrevista	estudo de campo
estudo de caso	estudo de observação	estudo experimental

estudo in-silico	estudo in-virtuo	estudo in-vitro
estudo in-vivo	estudo primário	estudo secundário
estudo viabilidade	experimento controlado	experimento replicado
fator	hipótese	hipótese nula
lição aprendida	medida direta	meta-análise
monitoramento de projeto	objeto	objeto de controle
participante	pesquisa na literatura	pré-estudo experimental
projeto fatorial	p-value	quasi-experimento
revisão sistemática	rodada	simulação
survey	teoria	toy study
tratamento	validade de conclusão	validade de construção
validade externa	validade interna	variável dependente
variável independente		

3.7. Revisão do Glossário de Termos

A construção de modelos de conhecimento é uma atividade essencialmente colaborativa (BREITMAN, 2005). Assim, submetemos o glossário e as ontologias à avaliação em grupos, visando melhorá-los em termos de completude e corretude. Conduzimos avaliações com planejamento específico para cada mecanismo de representação.

A oportunidade para executar estas avaliações surgiu durante a disciplina de Experimentação em Engenharia de Software, ministrada pelo Programa de Engenharia de Sistemas e Computação da COPPE/UFRJ. A disciplina foi direcionada para 22 alunos, dentre os quais 18 mestrandos e 4 doutorandos, tendo transcorrido nos meses de outubro, novembro e dezembro de 2008. A disciplina transcorreu com uma carga horária de 45 horas. A revisão do glossário de termos ocorreu ao final como um dos últimos trabalhos, de forma que o conhecimento acumulado ao longo da disciplina favorecesse a revisão. A avaliação do glossário e dos modelos foi planejada como uma inspeção, na qual algumas diretrizes foram fornecidas para identificação de discrepâncias em cada artefato. Estas discrepâncias foram então analisadas por pesquisadores do grupo ESE a fim de determinar quais, de fato, representariam defeitos.

A turma foi dividida aleatoriamente em duas equipes de três integrantes e oito equipes de dois integrantes. Para subsidiar a inspeção do glossário e dos modelos, cada

equipe foi orientada a pesquisar sobre um determinado tipo de estratégia de estudo primário, que são: pesquisa-ação, *survey*, estudo de caso, estudo controlado e estudo de observação. Lopes e Travassos (2008) identificaram o domínio de conhecimento de estratégia de estudo como sendo um dos domínios mais relevantes a ser considerado no contexto da etapa de planejamento do estudo, razão pela qual foi considerado na construção do glossário e das ontologias conforme descrito no capítulo anterior. Em função disto, o domínio de estratégia de estudo foi escolhido para ser alvo de pesquisa pelas equipes, de forma a avaliar o glossário segundo o conhecimento adquirido sobre uma estratégia específica. A quinta equipe foi orientada a pesquisar sobre estudos que aplicam simulação, dentre os quais se destacam estudos *in virtuo* e *in silico* (TRAVASSOS e BARROS, 2003).

Para viabilizar a pesquisa, foi disponibilizada aos alunos uma bibliografia básica de leitura obrigatória para todas as equipes. (WÖHLIN, 2000) foi a bibliografia básica selecionada para nivelar todos os alunos sobre os conceitos básicos de Experimentação em Engenharia de Software. Adicionalmente, uma bibliografia específica foi fornecida para cada equipe segundo o tema pesquisado. Além disso, cada equipe foi instruída a identificar e ler, obrigatoriamente, duas referências adicionais sobre o tema pesquisado. A identificação por parte da equipe de duas referências distintas é importante para não limitar a leitura dos alunos a apenas referências indicadas pelos pesquisadores envolvidos no planejamento da inspeção reduzindo, assim, possíveis fatores de confusão ao resultado. A tabela 3-2 relaciona as equipes e seus inspetores à bibliografia específica para pesquisa.

Tabela 3-2. Bibliografia selecionada para pesquisa pelas equipes

Equipe	Inspetor	Bibliografia específica para pesquisa
Pesquisa-ação 1	Inspetor 1	<ul style="list-style-type: none"> • (TRIPP, 2005) • (BASKERVILLE, 1999)
	Inspetor 2	
Pesquisa-ação 2	Inspetor 3	
	Inspetor 4	
<i>Survey</i>	Inspetor 5	<ul style="list-style-type: none"> • (MENDONÇA, 2004) • (KITCHENHAM, 2008)
	Inspetor 6	
	Inspetor 7	
Estudo de Caso 1	Inspetor 8	<ul style="list-style-type: none"> • (PERRY <i>et al.</i>, 2005)
	Inspetor 9	

Estudo de Caso 2	Inspetor 10	<ul style="list-style-type: none"> • (RUNESON, 2007)
	Inspetor 11	
Estudo de Observação 1	Inspetor 12	<ul style="list-style-type: none"> • (SEAMAN, 2008)
	Inspetor 13	
Estudo de Observação 2	Inspetor 14	
	Inspetor 15	
Estudo Controlado 1	Inspetor 16	<ul style="list-style-type: none"> • (JURISTO e MORENO, 2003) • (BASILI <i>et al.</i>, 1999)
	Inspetor 17	
Estudo Controlado 2	Inspetor 18	
	Inspetor 19	
Estudos com Simulação	Inspetor 20	<ul style="list-style-type: none"> • (VIGDER <i>et al.</i>, 2008) • (WOOLLARD <i>et al.</i>, 2008) • (MÜLLER <i>et al.</i>, 2008) • (BASILI <i>et al.</i>, 2008)
	Inspetor 21	
	Inspetor 22	

Diretrizes foram estabelecidas para inspecionar o glossário de termos. Cada equipe foi instruída a avaliá-lo identificando diferentes oportunidades de melhoria. Nesta avaliação, foi sugerida a utilização de uma taxonomia (Tabela 3-3) na qual cada classificação está relacionada a uma ação particular no glossário. A ação constitui uma oportunidade de melhoria no glossário.

Tabela 3-3. Taxonomia de defeitos e ação adotada para cada defeito identificado

Classificação do Defeito	Ação
Omissão de termo	Introduzir o termo identificado juntamente com sua descrição em inglês, português e espanhol. Citar também a referência que fundamenta a descrição.
Omissão de definição	Fornecer definição para os termos onde a mesma ainda não está especificada. Citar também a referência que fundamenta a descrição.
Correção de definição	Complementar ou corrigir a definição dos termos quando a equipe considerá-la incompleta ou imprecisa. Citar também a referência que fundamenta a descrição.
Alteração de nome	Alterar o nome dos termos para algum sinônimo que

	a equipe julga ser mais apropriado. Citar também a referência que fundamenta a descrição.
--	---

Cada equipe foi orientada a elaborar um documento detalhando quais modificações propôs, ressaltando uma justificativa apropriada para tal alteração e fornecendo a referência bibliográfica que subsidia a modificação. Para finalizar a atividade de inspeção e disseminar para toda a turma o conhecimento adquirido, cada equipe elaborou uma apresentação sobre o assunto pesquisado. Para as estratégias com mais de uma equipe, os integrantes se juntaram para elaborar a apresentação. Propusemos um conjunto de tópicos obrigatórios para serem abordados na apresentação sobre as estratégias. Os tópicos escolhidos seguem abaixo:

- Objetivo
- Características gerais
- Aplicabilidade (onde e para quem se aplica)
- Variações do estudo (quando pertinente)
- Procedimentos de execução
- Instrumentação envolvida
- Principal *deliverable*
- Grau de relevância experimental
- Ameaças mais comuns à validade do estudo
- Exemplos
- Propostas de alteração no glossário, ressaltando uma justificativa para cada modificação proposta

Após o término da inspeção, procedeu-se com a discriminação das discrepâncias apontadas. Foram indicadas 326 discrepâncias, sendo 74 falsos positivos. Duplicatas foram excluídas e o número de defeitos encontrados totalizou 190. A maior parte dos defeitos encontrados diz respeito a novos termos (105 no total), o que denota que o glossário não contemplava conceitos mais específicos relativos aos tipos de estudos experimentais. Outra categoria com quantidade expressiva de defeitos é a de omissão de definição. Um aspecto de grande importância da inspeção reside na identificação de defeitos sobre descrições incorretas ou incompletas. Estes defeitos são muito prejudiciais por interferirem no entendimento do pesquisador acerca dos defeitos, o que pode impactar negativamente na realização de um estudo. A inspeção totalizou 22 destes defeitos. A tabela 3-4 sintetiza os resultados alcançados por cada equipe para

cada tipo de defeito. Os defeitos identificados foram devidamente corrigidos no glossário. A lista de termos novos ou alterados encontra-se na tabela 3-5.

Tabela 3-4. Síntese dos resultados da inspeção do glossário de termos

Equipe	Número de defeitos identificados após discriminação				Total
	Omissão de termos	Omissão de definição	Correção de definição	Alteração de nome	
Pesquisa-ação 1	11	0	0	0	11
Pesquisa-ação 2	18	1	2	0	21
Survey	13	5	1	0	19
Estudo de Caso 1	15	8	5	0	28
Estudo de Caso 2	3	0	1	0	4
Estudo de Observação 1	3	0	0	0	3
Estudo de Observação 2	2	1	2	0	5
Estudo Controlado 1	13	15	5	4	37
Estudo Controlado 2	17	13	4	0	34
Estudos com Simulação	10	14	2	2	28
Total	105	57	22	6	190

Tabela 3-5. Lista de termos incluídos ou alterados na revisão do glossário

Termos	
Accuracy	Concurrent triangulation strategy
Action Elements	Confounding factor
Action Research	Constructivism
Action Researcher	Context of study
Activity	Continous simulation
Admissible transformation	Continuous variables
Alpha-value	Controlled experiment
Alternative hypothesis (H1)	Convenience sampling
Analytical generalization	Correlation
Analytical method	Critical theory
Balanced scale	Cross-sectional survey design
Balancing	Cumulative distribution

Base-rate question	Data collection
Beta-value	Deterministic simulation
Blocking variables	Discrete variable
Blocking	Distribution
Box plot	Dynamic simulation
Case study	Embedded case study
Causality question	Empirical method
Central limit theory	Empirical study
Checklist	Empirical validity
Closed question	Engineering method
CMD (Cooperative Method Development)	Entity
Experimental error	Measurement control
Experimental process	Measurement
Exploratory question	Method
External attribute	Metric
External replication	Model calibration
Factors	Model concept
Focus group	Model
Formal experiment	Multiple-case design
Framework of Ideas	Noise
GQM (Goal-Question-Metric)	Nonequivalent group design
Holistic case study	Non-parametric test
Hybrid simulation	Null hypothesis (H0)
Hypothesis testing	Object of study
Hypothesis	Objective data
In the field	Observer
Independent variables	Ontology
Input data analysis	Open question
Instrument	Outlier
Instrumentation	Output data analysis
Internal attribute	Parameters
Internal replication	Parametric test

Interrupted time-series designs	Participant
Interviewer specification	Perspective
Justified true belief	Philosophical stance
Life cycle models	Pilot study
Literal replication	Population
Local contingency	Positivism
Longitudinal survey design	Pragmatism
Mean squared error	Precision
Mean	Predictor variables
Meaningful measure	Primary study
Measure	Problem statement
Proposition	Self-administered questionnaire
Purpose	Sequential explanatory strategy
Qualitative research	Simple random sampling
Qualitative simulation	Simulation results
Qualitative study	Simulation technique
Quality focus	Single-case study
Quality improvement paradigm	Software metrics
Quantitative research	Static simulation
Quantitative simulation	Statistical generalization
Quantitative study	Stochastic simulation
Questionnaire specification	Stratified random sampling
Questionnaire	Subject
Quota sampling	Subjective data
Random assignment	Summated rating scale
Randomization	Survey design
Randomized experiment	Survey evaluation
Reading Techniques	Survey instrument
Reference behavior	Survey objective
Regression	Survey reliability
Relationship question	Survey validity
Replication	Survey

Research design	Systematic sampling
Research question	Target population
Resource	Techniques
Respondent	Test power
Response rate	Test
Response variable	Theoretical replication
RSD (Reflective Systems Development)	Theory building
Sample	Theory
Sampling bias	Triangulation
Scientific method	Type-I error
Scientific theory	Type-II error
Scope of survey	Undesired variations
Unit	
Unit of analysis	
Variance	
Viable research strategy	

3.8. Conclusões

Este capítulo apresentou um glossário de termos integrado ao repositório o eSEE. O glossário visa atender a necessidade de se propor uma terminologia única na área. Para isso, é disponibilizado em uma infra-estrutura *wiki*, na qual o pesquisador pode consultar os diferentes conceitos, visualizando definições e referências. Foi apresentada também a avaliação do conteúdo do glossário após sua integração ao glossário. A avaliação contemplou uma atividade de inspeção na disciplina de Experimentação em Engenharia de Software, ministrada pelo Programa de Engenharia de Sistema e Computação da COPPE/UFRJ. Importantes oportunidades de melhoria foram identificadas no glossário, o que contribuiu para melhorar sua completude e corretude.

O capítulo a seguir trata do processo de construção das ontologias do repositório, detalhando alguns de seus modelos principais. Descreve-se ainda o planejamento e a execução da avaliação dos modelos ontológicos, realizada de modo similar à inspeção do glossário.

Capítulo 4

Ontologias do Repositório de Conhecimento do eSEE

Neste capítulo apresentamos o processo de construção da ontologia e alguns dos principais modelos ontológicos. São detalhados o planejamento e a execução da inspeção que revisa os modelos ontológicos

4.1. Introdução

Para representar formalmente o conhecimento sobre o domínio de Engenharia de Software Experimental e seus subdomínios, necessitamos de mecanismos de representação mais expressivos que um glossário de termos. O trabalho de SJØBERG *et al.* (2007) já aponta para a necessidade de explicitarmos formalmente este conhecimento. Um cenário que exemplifica esta necessidade é quando o pesquisador deseja planejar um estudo qualitativo no eSEE. O glossário de termos deste ambiente contém a definição de estudo qualitativo e de outros termos pertinentes a este tipo de estudo, tais como *grounded theory* (SHULL *et al.* 2008). Contudo, como no glossário não está expressa a relação entre os termos “estudo qualitativo” e “*grounded theory*”, o ambiente não pode inferir e informar ao pesquisador que *grounded theory* pode ser aplicada no contexto de estudos qualitativos. Por esta razão, foram elaborados e integrados ao repositório modelos ontológicos que capturam os relacionamentos entre os conceitos do glossário. Esses modelos podem ser processados por ferramentas inteligentes (ABECKER *et al.*, 1999), capazes de realizar inferência computacional e disponibilizar conhecimento ao pesquisador.

A seção a seguir apresenta o processo adotado para construir as ontologias e detalha os principais modelos ontológicos. A construção das ontologias considerou os seguintes domínios de conhecimento:

- (1) Conhecimento geral sobre Experimentação em Engenharia de Software, de forma a explicitar os conceitos expressos no glossário e com o qual as ontologias serão interdependentes dentro do repositório;

- (2) Conhecimento sobre os subdomínios de abordagem, estratégia e ambiente de estudo, além de método de pesquisa, conforme identificados em (LOPES e TRAVASSOS, 2008) como sendo domínios de conhecimento importantes na atividade de planejamento de estudo. A Ontologia de Pesquisa Científica de BIOLCHINI *et al.* (2006) será evoluída para melhorar a abrangência de conceitos de (1) e será evoluída para contemplar o corpo de conhecimento dos subdomínios citados acima;
- (3) Conhecimento sobre o pacote do experimento, mapeado no trabalho de AMARAL (2003). Como este repositório será integrado ao eSEE, este conhecimento deve ser preservado e devidamente registrado nas ontologias. O conjunto de modelos que explicitam este conhecimento é denominado Ontologia do Pacote do Experimento;
- (4) Conhecimento sobre a integração entre as ontologias de Pesquisa Científica e Pacote do Experimento. Esta integração dar-se-á por modelos específicos que relacionam os conceitos de uma ontologia com o de outra. Ou seja, esses modelos expressam o conhecimento sobre em quais documentos da Ontologia de Pacote de Experimento são registrados os conceitos da Ontologia de Pesquisa Científica. Esse conhecimento é particularmente importante para a construção da ferramenta que atua como prova de conceito para utilização do repositório. A ferramenta, descrita no próximo capítulo, utilizará esses modelos para derivar quais *templates* de documentos serão gerados e com que informações.

4.2. Processo de Construção das Ontologias

Ontologias sobre o domínio de experimentação em Engenharia de Software já vêm sendo alvo de estudos na comunidade. Um dos trabalhos para mapear conceitos deste domínio é a Ontologia de Pesquisa Científica (BIOLCHINI *et al.*, 2006), criada com o intuito de garantir homogeneidade terminológica dos conceitos envolvidos na pesquisa em Engenharia de Software. O foco da ontologia está em representar o conhecimento sobre Revisão Sistemática. Adicionalmente, foi externalizado também conhecimento dos domínios de Estudos Primário, Método Experimental e Engenharia de Software. A figura abaixo ilustra a relação taxonômica entre as entidades que representam esses domínios.

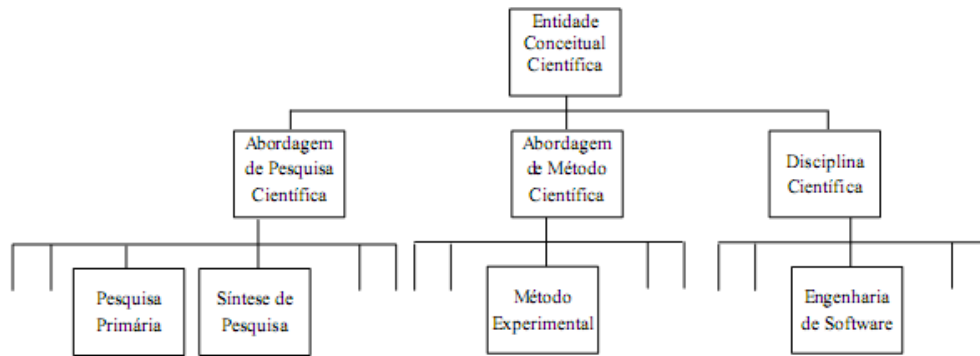


Figura 4-1. Relação taxonômica entre os domínios (BIOLCHINI *et al.*, 2006)

A ontologia (Figura 4-2) utiliza dois tipos de associações: “é um” e “tem”. (BIOLCHINI *et al.*, 2006). Do ponto de vista estrutural, os conceitos estão organizados em níveis. Neste tipo de organização, uma ontologia básica e central para o domínio em estudo é construída e, a partir dela, ontologias de nível mais alto, adicionando novos elementos, são construídas para estender as ontologias dos níveis inferiores (FALBO, 1998). A figura 4-2 a seguir ilustra os quatro primeiros níveis da hierarquia taxonômica da entidade de Pesquisa Primária, presente na figura 4-1. Esta ontologia mostra a organização hierárquica para o conceito Medição na Ontologia de Pesquisa Primária, incluindo relações “é um” e “tem” entre os elementos na ontologia.

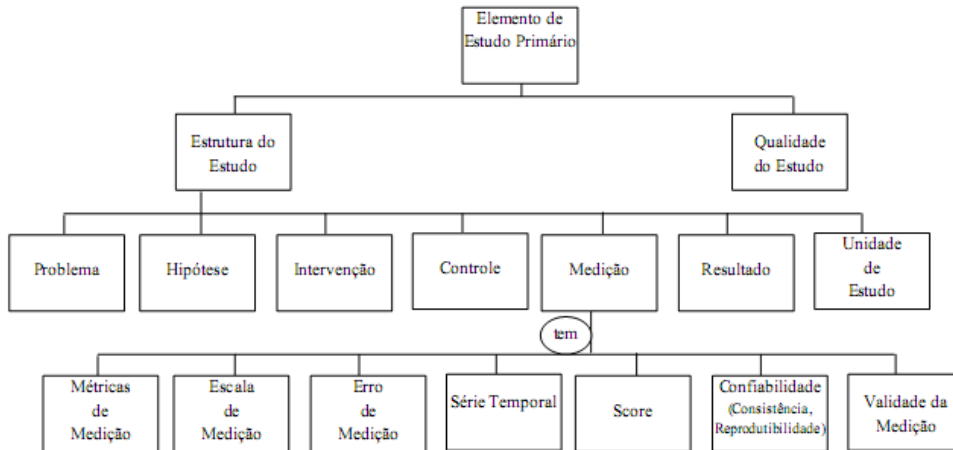


Figura 4-2. Relações entre subcategorias do conceito Medição (BIOLCHINI *et al.*, 2006)

O ponto de partida para a definição do conhecimento que está envolvido na condução de Revisões Sistemáticas em Engenharia de Software foi estabelecer os diferentes domínios relacionados a esta área, tais como Método Experimental, Pesquisa Primária, Síntese de Pesquisa e Engenharia de Software. Entretanto, como o foco da construção da ontologia está em Revisão Sistemática, os modelos carecem de uma maior cobertura de conceitos sobre estas áreas, principalmente para os conceitos do domínio de Estudos Primários (Figura 4-2). Esta deficiência dificulta a integração direta

da Ontologia de Biolchini *et al.* (2006) ao repositório de conhecimento proposto, uma vez que o conhecimento sobre estudos primários exerce grande influência na etapa de planejamento do Processo de Experimentação e na instanciação de ambientes no eSEE (LOPES e TRAVASSOS, 2008). Assim, para fazer parte da estrutura do repositório proposto, a ontologia construída, baseada no conjunto de conceitos e relacionamentos expressos nos modelos de Biolchini *et al.* (2006), foi agregada de conhecimento sobre estudos primários identificados por LOPES e TRAVASSOS (2008). Este conhecimento contempla conceitos de estratégia de estudo, método de pesquisa e ambiente de estudo.

O processo adotado para construção da ontologia foi o *Methontology* (FERNANDÉZ *et al.*, 1997), desenvolvido no laboratório de Inteligência Artificial do Politécnico de Madri (Figura 4-3). Sua adoção foi influenciada pela ampla utilização em diferentes comunidades e por basear-se no processo-padrão IEEE para o desenvolvimento de software. O ANEXO II detalha o processo *Methontology* em termos de suas atividades. A figura 4-3 ilustra o processo adotado. Em seguida, cada atividade do processo é detalhada.

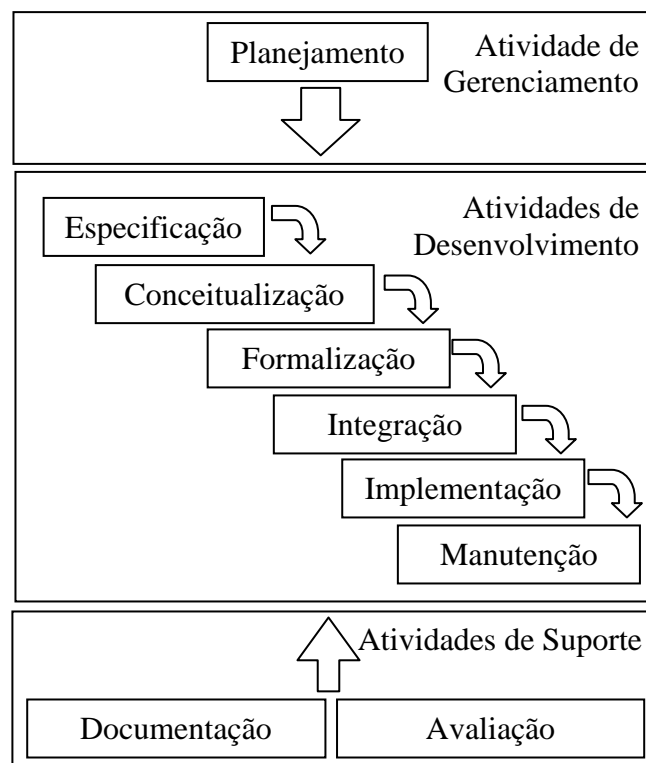


Figura 4-3. Processo de construção de ontologias adotado (FERNANDÉZ *et al.*, 1997)

4.2.1. Planejamento

No processo *Methontology*, a etapa de avaliação é uma atividade de apoio às atividades de desenvolvimento da ontologia e sua realização é flexível ao longo do processo (FERNANDÉZ *et al.*, 1997). Na etapa de planejamento, definimos como sendo adequado executar a avaliação da ontologia após a etapa de formalização, que já apresenta como produto um modelo formal com nível de completude satisfatório. Além disso, quanto mais cedo no processo for possível detectar defeitos nos modelos, menor o custo da correção dos mesmos. Caso a avaliação fosse conduzida somente após a atividade de implementação, as correções seriam conduzidas nos modelos gerados nesta atividade e também na etapa de formalização, aumentando assim o esforço de correção.

A etapa de planejamento, tal como é proposta no *Methontology*, não foi executada em todos os detalhes. Características como tempo e custo de construção da ontologia não foram alvo de avaliação por não representarem impacto relevante no escopo desta dissertação.

4.2.2. Especificação

A etapa de especificação envolveu definir a razão para a construção e a forma de utilização da ontologia. A razão está diretamente relacionada com a necessidade de construção do repositório de conhecimento para o eSEE. A forma de uso se dará por ferramentas inteligentes que recuperam conhecimento dos modelos. No capítulo 5 descreve-se um protótipo de ferramenta que ilustra a utilização do repositório neste sentido.

4.2.3. Conceitualização

O objetivo da etapa de conceitualização consistiu em melhorar a cobertura dos termos presentes na ontologia original de Ontologia de Pesquisa Científica (BIOLCHINI *et al.*, 2006). Como o repositório prevê a interdependência entre o glossário e as ontologias, o conjunto de termos do primeiro atuou também como fonte de termos para o segundo. Os termos presentes no glossário e ausentes na ontologia foram incorporados nesta. Os termos adicionados com sua respectiva definição totalizaram 25.

Além de incluir novos termos na ontologia provenientes do glossário, também foi realizada uma comparação para determinar quais termos do glossário são homônimos ou sinônimos àqueles presentes na versão original da Ontologia de Pesquisa Científica. Os termos de mesmo nome foram avaliados quanto à definição: caso a referência da definição do termo no glossário seja mais atual do que àquela da ontologia, a definição foi atualizada com a do glossário. Caso contrário, a definição foi mantida. Como não havia definição para diversos termos na ontologia, a maior parte das definições foi atribuída a partir do glossário, totalizando 21 atribuições. As definições que permaneceram totalizam seis, sendo que a definição do respectivo termo no glossário foi atualizada.

Quanto aos termos sinônimos, a decisão a respeito de qual nome será adotado tanto na ontologia quanto no glossário, envolveu também a avaliação da referência da definição: caso a referência da definição do termo no glossário seja mais atual do que àquela da ontologia, o nome do termo no glossário prevalece e é alterado na ontologia. Caso contrário, o nome na ontologia permanece e o do respectivo termo no glossário é alterado. Um total de quatro nomes foi atualizado na ontologia e nenhum foi alterado no glossário.

Realizou-se também o trabalho inverso: os termos presentes na ontologia e que não estavam presentes no glossário foram incluídos. Como o foco da ontologia de BIOLCHINI *et al.* (2006) é mapear o conhecimento sobre Revisão Sistemática, todos os termos adicionados ao glossário (8 no total) são deste domínio. A respectiva definição na ontologia também foi incluída no glossário.

É importante ressaltar que o foco desta etapa é melhorar a cobertura dos termos na ontologia, incluindo novos termos a partir do glossário e atualizando os nome e a definição de conceitos homônimos e sinônimos. A identificação de novos relacionamentos é tratada na etapa seguinte, de formalização.

4.2.4. Formalização

Na etapa de formalização, o modelo semi-formal da etapa de conceitualização é gerado utilizando uma linguagem de representação de ontologias. A linguagem adotada é a LINGO (FALBO, 1998), que se caracteriza por estabelecer um perfil UML, isto é, um subconjunto pré-definido de elementos padrão da UML, associados a restrições, valores etiquetados e estereótipos. Estes elementos reunidos especializam e configuram a UML para um determinado domínio de aplicação ou propósito particular. Como a

notação UML é considerada uma linguagem intermediária para construção de ontologias (FERNANDEZ *et al.*, 1997), detalhes de implementação das mesmas não precisam ser abordados nesta etapa, reduzindo a complexidade de construção dos modelos. O perfil UML define elementos para representar conceitos, relações e propriedades, diminuindo substancialmente o esforço na escrita de axiomas formais. Não foi objetivo deste trabalho a definição desses axiomas. A figura 4-4 ilustra um exemplo de como alguns elementos do perfil podem ser utilizados para representar conhecimento.

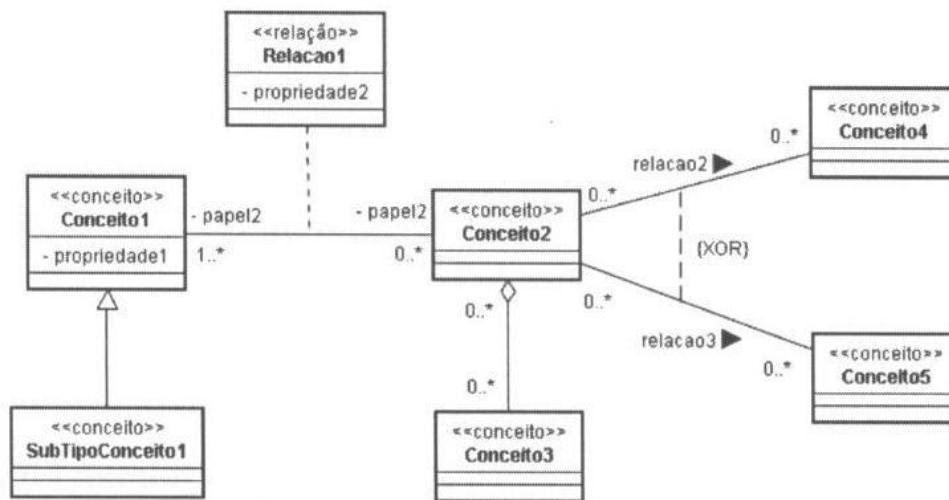


Figura 4-4. Exemplo de representação de conhecimento utilizando a linguagem LINGO (MIAN, 2003)

Para não prevalecer a semântica de “classe” em UML, a linguagem LINGO adota o mecanismo de extensão de estereótipo. Classes com estereótipo <<conceito>> representam conceitos da ontologia, da mesma forma que relações que contém propriedades ou que possuem cardinalidade maior que dois, são representadas como classes associativas com estereótipo <<relação>>.

As relações binárias sem propriedades são definidas como associações nomeadas. As propriedades de conceitos e relações são representadas como atributos de classes estereotipadas. Relações de subtipo e todo-parte são representadas como relações de generalização/especialização e de agregação/composição, respectivamente. Condicionantes entre relações são representados por outro mecanismo de extensão da UML, as restrições entre associações (MIAN, 2003).

4.2.4.1. Formalização da Ontologia Pesquisa Científica

A etapa de formalização contemplou a identificação dos relacionamentos entre os novos termos e aqueles já existentes na ontologia original. Para identificá-los, adotou-se como bibliografia a mesma referência em Engenharia de Software Experimental utilizada na melhoria da cobertura dos termos do glossário: (WÖHLIN, 2000). Foram mapeados na ontologia 32 novos relacionamentos.

A adoção da linguagem LINGO (FALBO, 1998) consistiu em traduzir cada tipo de relacionamento da linguagem original adotada por BIOLCHINI *et al.* (2006) no relacionamento UML correspondente. Os relacionamentos hierárquicos “é um” (*is a*) da ontologia de BIOLCHINI *et al.* (2006) foram substituídos por relacionamento UML de generalização. Já os relacionamentos “parte de” (*part of*) foram substituídos por agregações ou composições. Utilizou-se agregação quando há uma evidência na literatura técnica de que um dos conceitos “é parte” também de outro conceito da ontologia. A composição foi empregada quando o conceito denotado como “parte” é específico e não está presente em qualquer outro modelo ontológico. A figura 4-5 ilustra a Ontologia de Pesquisa Científica evoluída utilizando a linguagem LINGO. A ontologia contempla os novos conceitos identificados na etapa de conceitualização e os novos relacionamentos mapeados no início da fase de formalização.

À medida que a Ontologia de Pesquisa Científica é formalizada utilizando a linguagem LINGO, observou-se a necessidade de reorganizar o modelo. Os termos inseridos na ontologia a partir do glossário contribuíram para aumentar consideravelmente o tamanho do modelo, tornando mais difícil sua leitura. Grande parte desses termos é pertinente aos domínios de abordagem de pesquisa, estratégia de estudo, método de pesquisa e ambiente de estudo. Em vista disto, para os domínios de **estratégia de estudo**, **método de pesquisa** e **tipo de estudo**, cada qual com seu respectivo conceito na ontologia, novos modelos ontológicos foram construídos, de modo que modelos mais específicos expressem termos e relacionamentos pertinentes a um domínio em particular. Assim, torna-se mais fácil a leitura da Ontologia de Pesquisa Científica. Os novos modelos são denominados “subontologias” por designarem ontologias mais específicas e que estendem um modelo ontológico com conceitos mais gerais. Os modelos ontológicos construídos são:

- Subontologia de Estratégia de Estudo
 - Subontologia de Pesquisa-ação

- Subontologia de Survey
- Subontologia de Estudo Caso
- Subontologia de Estudo de Observação
- Subontologia de Estudo Controlado
- Subontologia de Método de Pesquisa
 - Subontologia de Método Qualitativo
 - Subontologia de Método Quantitativo
- Subontologia de Ambiente de Estudo
 - Subontologia de Estudo *In Vivo*
 - Subontologia de Estudo *In Vitro*
 - Subontologia de Estudo *In Virtuo*
 - Subontologia de Estudo *In Silico*

Por exemplo, a Subontologia de Pesquisa-ação expressa conceitos mais específicos em relação à Subontologia de Estratégia de Estudo e pertinentes apenas ao domínio de Pesquisa-ação. Já a Subontologia de Estratégia de Estudo estende a Ontologia de Pesquisa Científica, da qual foram removidas as referências aos termos do domínio de estratégia de estudo. Com a construção dessas subontologias, a Ontologia de Pesquisa Científica passa a explicitar apenas conhecimentos gerais sobre pesquisa científica. Os modelos citados acima serão descritos em detalhes nas próximas seções.

No texto a seguir, os termos em **negrito** ressaltam o nome de um conceito dentro da ontologia. Nomes de relacionamentos estão destacados em *itálico*. A figura 4-5 ilustra a Ontologia de Pesquisa Científica construída após a segregação dos termos dos domínios de **estratégia de estudo**, **método de pesquisa** e **tipo de estudo**. Os conceitos da ontologia são explicados abaixo

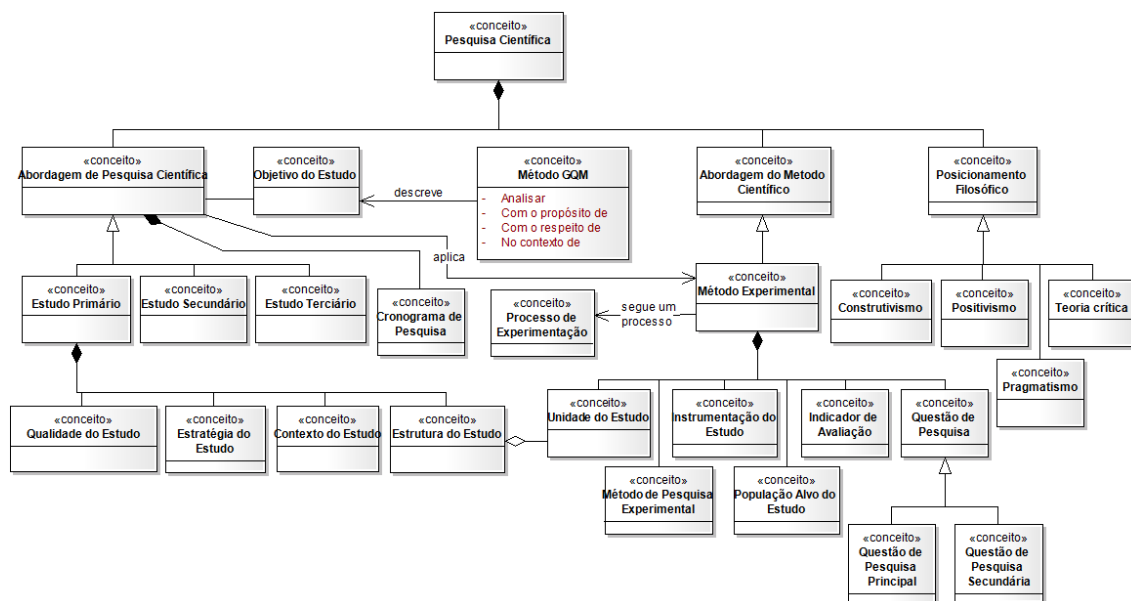


Figura 4-5. Ontologia de Pesquisa Científica do repositório de conhecimento do eSEE

Incorporado por agregação ao conceito de **pesquisa científica** está o conceito **abordagem de pesquisa científica**. Ligados a este conceito por relação de especialização, estão os conceitos de **estudo primário**, **estudo secundário** e **estudo terciário**. Estudos terciários são recentes em Engenharia de Software e consistem na agregação de resultados de estudos secundários. O conhecimento sobre estratégia de estudo e tipo de estudo, organizado no nível meta da infra-estrutura, foi inserido no contexto do conceito **estudo primário**. Os conceitos **estratégia de estudo** e **tipo do estudo** foram incorporados por composição a **estudo primário**.

Herdados da ontologia de BIOLCHINI *et al.* (2006) estão os conceitos de **abordagem de método científico**, relacionado por agregação à **pesquisa científica**, e **método experimental**, caracterizado como um tipo de **abordagem de método científico**. JURISTO e MORENO (2001) apontam que uma abordagem de pesquisa científica *aplica*, independente do tipo, o método experimental como forma de gerar evidências, razão pela qual foi estabelecido que uma **abordagem de pesquisa científica aplica** um **método experimental**.

Os conceitos relacionados à **abordagem de método científico** foram reorganizados. Permanece relacionado a **método experimental** o conceito de **unidade de estudo**, que representa, em diferentes níveis de abstração, o alvo da pesquisa. Em BIOLCHINI *et al* (2006), o conceito **objeto do estudo** estava relacionado como um elemento de **método experimental**. Entretanto, em referências como (JURISTO e MORENO, 2001), o **objeto do estudo** é um conceito integrante de outro maior, a **instrumentação do estudo**. Sendo assim, foi substituído o relacionamento de **método**

experimental com **formulação do objeto** pela composição com o conceito **instrumentação do estudo**.

Outro conceito introduzido no contexto de **método experimental** foi **população-alvo do estudo** (WÖHLIN *et al.* 2000), que refere-se à população da qual será obtida a amostra de indivíduos que efetivamente participarão do estudo experimental. O conceito **método de pesquisa experimental**, referenciado no nível meta da infra-estrutura conceitual do eSEE, foi considerado como parte integrante de **método experimental** (WÖHLIN *et al.* 2000).

4.2.4.2. Construção da Subontologia de Estratégia de Estudo

Na subontologia de estratégia de estudo (Figura 4-6), uma visão mais específica da ontologia de Pesquisa Científica é apresentada, contemplando mais um nível de conhecimento. Este novo nível representa o conhecimento no contexto do conceito estratégia do estudo.

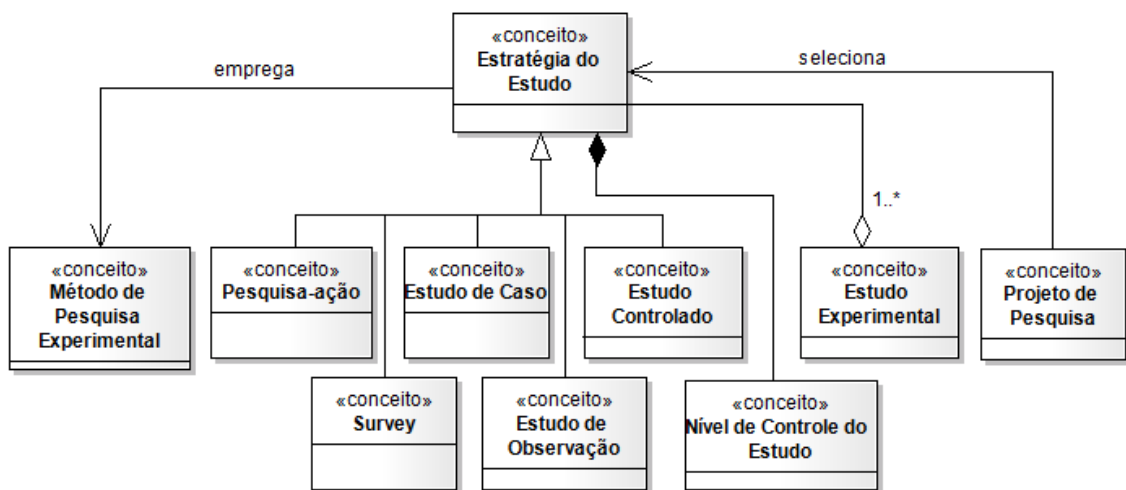


Figura 4-6. Subontologia de Estratégia de Estudo

Um **estudo experimental** consiste em um estudo no qual uma ou mais estratégias podem ser combinadas na pesquisa. Por exemplo, o estudo pode contemplar inicialmente um *survey* para caracterização dos participantes e, em seguida, executar um estudo de caso com esses participantes. Um **projeto de pesquisa** *seleciona* uma estratégia de estudo para um problema de pesquisa específico, aproveitando-se de seus pontos fortes e atenuando seus pontos fracos; é a lógica que liga os dados coletados às questões iniciais do estudo. A validade dos resultados depende de quão bem o projeto de pesquisa compensa os pontos fracos das estratégias (EASTERBROOK *et al.*,2008).

Uma estratégia de pesquisa *emprega* um **método de pesquisa** específico para analisar os dados obtidos. O conceito de **nível de controle** define quanto controle o pesquisador possui em uma determinada estratégia. Os tipos de estratégia são **pesquisa-ação, survey, estudo de caso, de observação e controlado**.

COSTA *et al.* (2004) definiram uma classificação taxonômica para estudos primários, que considera, dentre outras perspectivas, a classificação segundo a estratégia de estudo. COSTA *et al.* (2004) indica as seguintes estratégias que foram incluídas como conceitos na subontologia: **survey, estudo de caso e estudo controlado**.

O conceito **survey** refere-se a uma investigação realizada, quando, por exemplo, uma determinada tecnologia tem sido utilizada durante certo período de tempo (SHULL *et al.* 2008). Um survey permite capturar um "retrato instantâneo" da situação atual. Através de questionários ou entrevistas, dados são coletados de uma amostra de indivíduos que representam um grupo específico.

Estudo de caso é o conceito de uma estratégia que investiga um fenômeno contemporâneo dentro de um contexto real, especialmente quando os limites do fenômeno e do contexto não são claros (YIN, 2003).

O conceito **estudo controlado** consiste em uma estratégia na qual o objetivo é manipular uma ou mais variáveis e controlar as demais a níveis fixos ou determinados. Em um estudo experimental, os participantes são atribuídos a diferentes tratamentos de forma aleatória. O objetivo é manipular uma ou mais variáveis e controlar todas as outras variáveis num valor fixo. O efeito da manipulação das variáveis é, então, medido e, com base nessa medição, análises estatísticas são conduzidas (WÖHLIN *et al.* 2000).

Demais conceitos introduzidos como estratégias de estudo são **pesquisa-ação, e estudo de observação**. Em **pesquisa-ação** objetiva-se resolver uma situação social de um domínio específico do mundo real, através da participação intervencionista e colaborativa de pesquisadores junto aos demais envolvidos no problema. Busca-se reunir a prática e a pesquisa, ação e reflexão para promover melhorias na situação e contribuir para o conhecimento científico (SANTOS e TRAVASSOS, 2008).

Estudo de observação é uma estratégia que pode se assemelhar às demais, a depender do planejamento do estudo. O investigador não expõe objetos ou participantes a nenhum tratamento. O propósito de um estudo de observação pode ser de caracterizar, definir um *baseline* e/ou identificar relacionamentos. Os dados podem ser coletados

através de observação visual, entrevistas e formulários de coleta de dados, ou pode ainda utilizar dados históricos de um repositório (DALY *et al.*, 1997).

Os conceitos de cada tipo de estratégia de estudo foram separados em modelos específicos, cada qual constituindo uma subontologia de uma dada estratégia de estudo. Foram construídas as subontologias de pesquisa-ação, *survey*, estudo de caso, estudo controlado e estudo de observação. A seguir é apresentada a subontologia de *survey* (Figura 3-14), explicitando os principais conceitos deste domínio. Demais subontologias podem ser encontradas no ANEXO I desta dissertação.

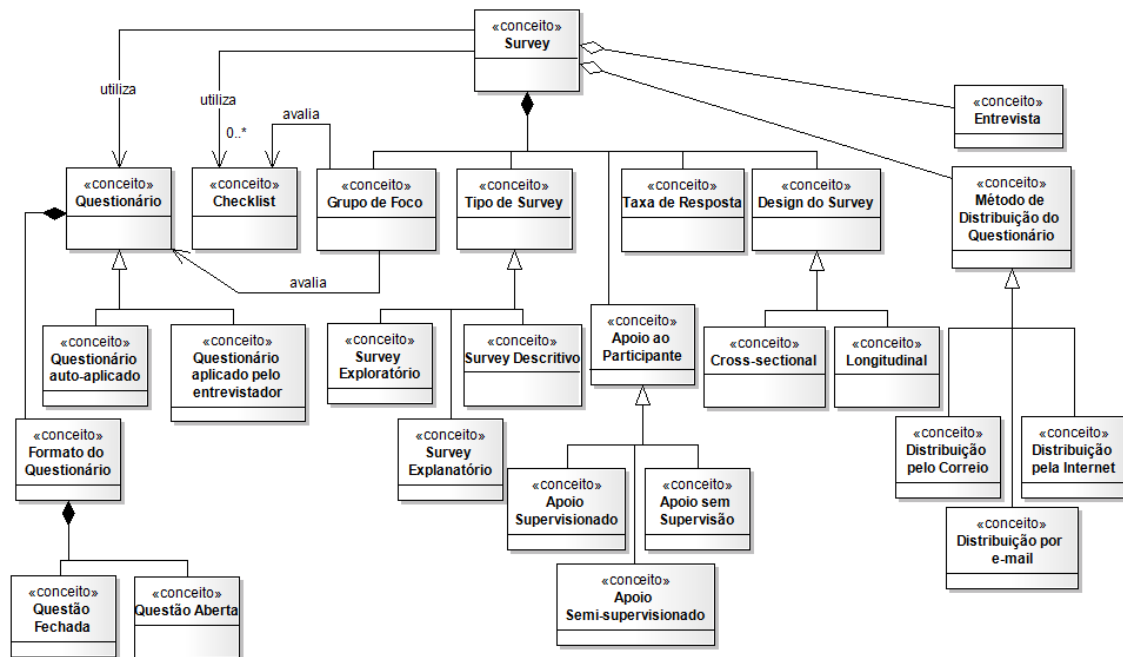


Figura 4-7 - Subontologia de Survey

Agregado ao conceito de **survey** está o conceito de **entrevista**, comumente aplicada neste tipo de estratégia de estudo. Os conceitos **grupo de foco**, **tipo**, **taxa de resposta** e **design do survey** são pertinentes ao domínio de *survey*, por isso a ligação via composição com este conceito. O **grupo de foco** consiste em um grupo de discussão mediano que ajuda a avaliar o instrumento do *survey* (KITCHENHAM *et al.*, 2008). O tipo de *survey* pode ser **exploratório**, **descritivo** ou **explanatório**. O primeiro visa familiarizar-se com o tópico ou identificar os conceitos iniciais sobre um tópico, dando ênfase na determinação de quais conceitos devem ser medidos e como devem ser medidos (KITCHENHAM *et al.*, 2008). O tipo descritivo objetiva identificar situações, eventos, atitudes ou opiniões de uma população. O tipo explanatório tem como objetivo testar uma hipótese e as relações causais, além de questionar por que a relação existe.

A **taxa de resposta** de um survey indica quanto dos indivíduos da população alvo do estudo efetivamente responderam ao survey no caso do uso de um questionário ou quantos aceitaram serem entrevistados. O conceito de **design** está ligado à estrutura adotada para conduzir a estratégia. Os tipos de design, relacionados por herança ao conceito de design, são **cross-sectional** e **longitudinal**. No primeiro, os participantes são questionados em momentos específicos. Por exemplo, todos os membros de uma organização de desenvolvimento de software podem ser requisitados às dez da manhã de uma segunda-feira para determinar em quais atividades estão trabalhando naquela manhã. Esta informação fornece um panorama de como está o andamento das atividades na organização. Já no design longitudinal, as informações são obtidas para avaliar mudanças em uma população específica durante certo tempo. Neste tipo de design, o mesmo conjunto de participantes pode ser questionado em períodos de tempo específicos ou diferentes pessoas podem ser escolhidas (MENDONÇA, 2004).

O **apoio ao participante** de um survey pode ser **supervisionado**, **semi-supervisionado** ou **sem supervisão**. No apoio supervisionado, o pesquisador auxilia o respondente para garantir que todas as questões foram entendidas e respondidas. No apoio semi-supervisionado, o pesquisador explica os objetivos e o formato da pesquisa, deixando que o respondente obtenha a informação para resposta por conta própria. Na ausência de supervisão as instruções da pesquisa são descritas no documento (KITCHENHAM *et al.*, 2008).

Um survey pode utilizar **questionários** e **checklists**, a depender do planejamento do estudo. Um questionário possui um **formato próprio**, que está ligado por associação aos conceitos de **questão fechada** ou **aberta**. Uma questão é fechada quando os participantes devem escolher uma resposta a partir de uma lista de opções predefinida, enquanto que em uma questão aberta os entrevistados são questionados para enquadrar a sua própria resposta (KITCHENHAM *et al.*, 2008). Um questionário pode ser de dois tipos: **auto-aplicado** ou **aplicado pelo entrevistador**. Um questionário auto-aplicado é aplicado pelo próprio participante da pesquisa (e não por um entrevistador), seguindo apenas as instruções definidas no questionário. Já o segundo refere-se à especificação do questionário quando este é feito por entrevista (KITCHENHAM *et al.*, 2008).

Agregado ao conceito de survey está o **método de distribuição de um questionário**, que pode ser por **correio**, **internet** ou **email**.

4.2.4.3. Construção da Subontologia de Método de Pesquisa

A subontologia de método de pesquisa (Figura 4-8) ilustra o conhecimento sobre os tipos de método de pesquisa, a saber: **métodos quantitativo, qualitativo, semi-quantitativo e triangulação concorrente**.

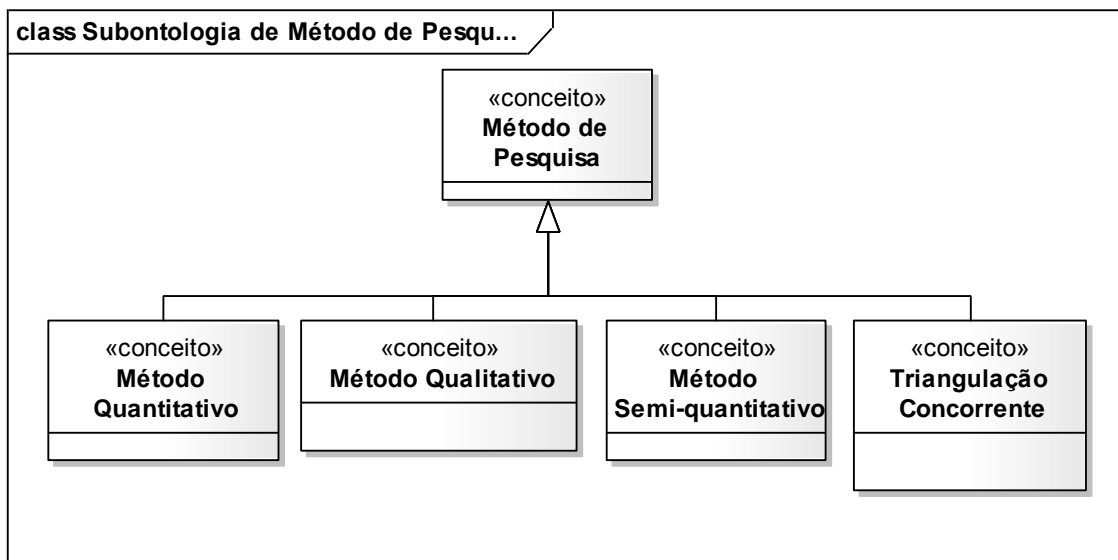


Figura 4-8. Subontologia de Método de Pesquisa

O conceito **método quantitativo** consiste na representação numérica de observações com o propósito de descrever e explicar um fenômeno. A pesquisa quantitativa inicia-se com a coleta de dados, seguida da aplicação de vários métodos estatísticos descritivos e de inferência (TAYLOR e BOGDAN, 1998). Uma vantagem desta abordagem é permitir comparações e análises estatísticas (WÖHLIN *et al.*, 2000). O **método qualitativo** busca focar na compreensão do fenômeno de pesquisa em seu contexto de ocorrência natural. De um modo geral, as principais técnicas aplicadas são a observação de participantes, entrevistas e análise de artefatos (TAYLOR e BOGDAN, 1998). O **método semi-quantitativo** procura entender o comportamento de um sistema a partir de relações causais entre as variáveis que o descrevem. É aplicado quando não há conhecimento dos valores numéricos em relações matemáticas que regem as mudanças de um sistema, o que geralmente está ligado ao fato da mudança ser conhecida, mas não o tamanho de seu efeito (OGBORN e MILLER, 1994). Frequentemente a aplicação do **método semi-quantitativo** é acompanhada da simulação do comportamento a ser observado (WIDMAN, 1989).

A estratégia de **triangulação concorrente** usa os métodos qualitativo e quantitativo concorrentemente, na tentativa de confirmar, validar e embasar descobertas (EASTERBROOK *et al.*, 2008).

Os conceitos **método quantitativo** e **método qualitativo** estão relacionados a conceitos específicos que os descrevem em detalhes. Tais conceitos foram especificados em subontologias distintas, que são: subontologia de método quantitativo e subontologia de método qualitativo, que se encontram descritas no ANEXO I. O modelo dos métodos semi-quantitativo e de triangulação concorrente ainda não foram construídos.

4.2.4.4. Construção da Subontologia de Ambiente de Estudo

A subontologia de ambiente de estudo (Figura 4-9) ilustra os conceitos que representam os tipos de contexto de estudo, que são: *in vivo*, *in vitro*, *in virtuo* e *in silico* (TRAVASSOS e BARROS, 2003).

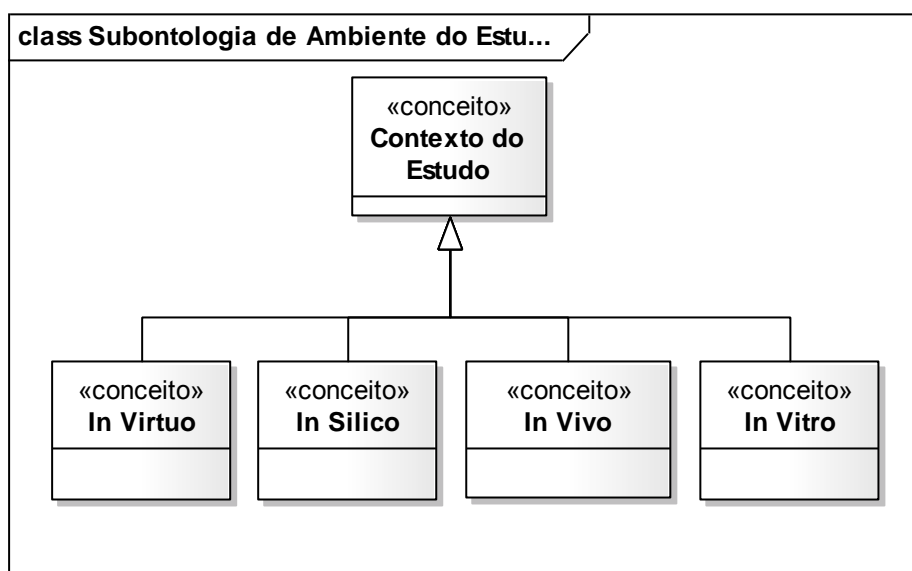


Figura 4-9. Subontologia de Tipo de Estudo

Estudos *in vivo* são aqueles que envolvem os indivíduos em seus próprios ambientes de trabalho. Tais estudos experimentais ocorrem durante o processo de desenvolvimento em circunstâncias reais. Estudos *in vitro* são aqueles executados e controlados em ambientes tais como laboratórios ou comunidades controladas. Nestes estudos, pesquisadores criam situações próximas às reais em ambientes preparados (“laboratórios”) para observar como os participantes executam uma determinada tarefa (TRAVASSOS e BARROS, 2003).

Estudos *in virtuo* são conduzidos em um ambiente virtual, composto por modelos numéricos que são manipulados por indivíduos. O comportamento do modelo é analisado através de simulação computacional. Em estudos experimentais *in virtuo*, um modelo computacional representa o objeto em estudo e, em alguns casos, o ambiente onde o estudo experimental é realizado. Os indivíduos que interpretam os resultados obtidos pela simulação do modelo podem manipulá-lo através de um programa de computador. Estudos *in silico* são aqueles em que, além do objeto e do ambiente a serem estudados, o comportamento dos indivíduos envolvidos também é descrito por modelos computacionais (TRAVASSOS e BARROS, 2003).

Os conceitos *estudo in vivo*, *in vitro*, *in virtuo* e *in silico* estão associados a outros conceitos pertinentes ao seu domínio particular. A fim de representar esse conhecimento, foram construídas as subontologias de estudo *in vivo*, *in vitro*, *in virtuo* e *in silico*. Estas subontologias encontram-se descritas no ANEXO I.

4.2.4.5. Formalização da Ontologia do Pacote do Experimento

O trabalho de AMARAL (2003) organizou o pacote de estudos experimentais a partir da definição de seus modelos de documento, que especificam o conjunto de informações que cada documento físico deve registrar. Para demonstrar a viabilidade do uso deste pacote, CHAPETTA (2006) construiu uma infra-estrutura computacional para apoiar a execução e empacotamento de estudos experimentais utilizando o pacote de AMARAL (2003). Esta infra-estrutura foi incorporada ao eSEE como sendo um de seus alicerces para apoiar o processo de Experimentação. O modelo de pacote de AMARAL (2003) encontra-se descrito no ANEXO III.

Tendo vista que o repositório sendo construído deve ser disponibilizado no contexto do eSEE, o conhecimento sobre o pacote de estudos de AMARAL (2003) não pode ser perdido e, portanto, deve ser explicitado. Consequentemente, a etapa de formalização procedeu com a construção de modelos ontológicos que explicitam o conhecimento sobre os documentos envolvidos no processo de experimentação, tal como estão definidos em (AMARAL, 2003). Os modelos de documentos foram definidos por AMARAL (2003) em modelos UML, o que simplificou a conversão para os modelos ontológicos. Adotou-se a organização original do modelo de pacote de AMARAL (2003), sendo construídos os seguintes modelos ontológicos:

- Ontologia do Pacote do Experimento
 - Subontologia de Documento de Organização do Experimento

- Subontologia de Formulário de Resultados
- Subontologia de Instrumento
 - Subontologia de Artefato de Software
 - Subontologia de Artefato do Experimento
- Ontologia do Processo de Experimentação
- Ontologia de Participantes, Papéis e Responsabilidades
- Ontologia de Permissões de Acesso aos Documentos

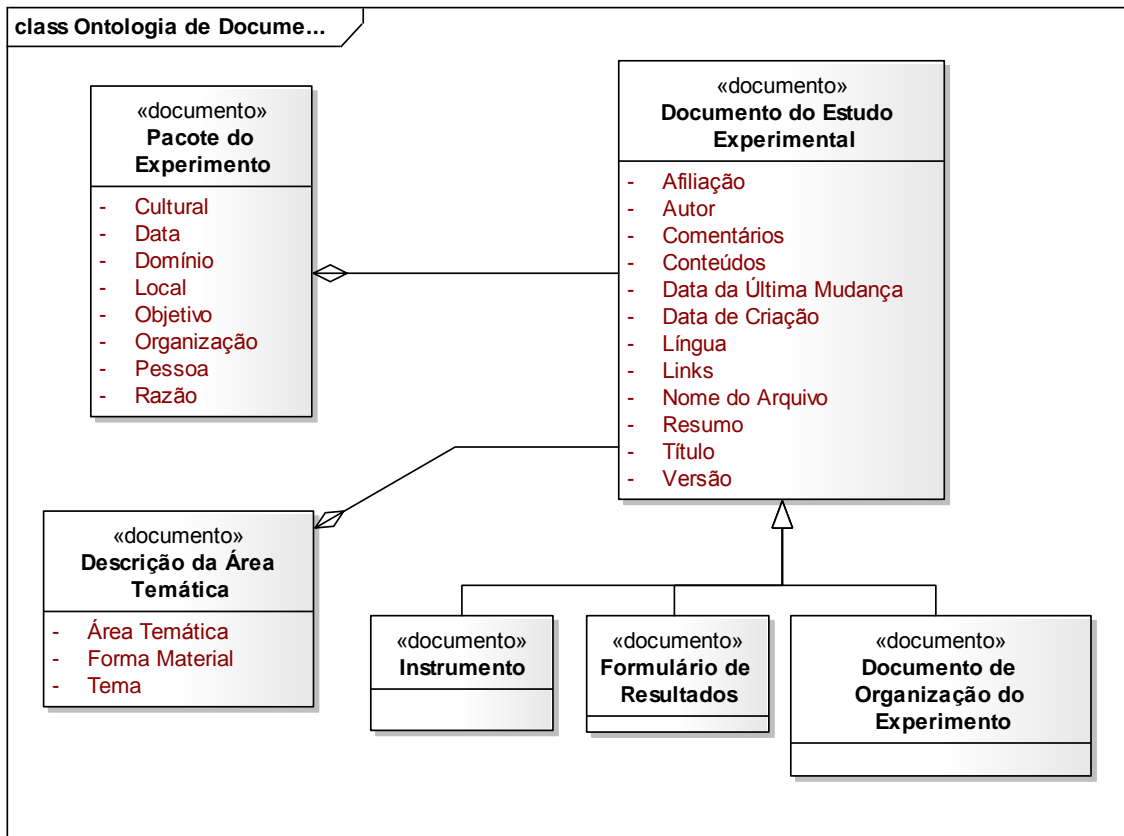


Figura 4-10 - Ontologia do Pacote do Experimento

A figura 4-10 ilustra a Ontologia do Pacote do Experimento, na qual estão descritos o documento do **Pacote do Experimento**, que descreve em linhas gerais o pacote e ao qual estão agregados todos os documentos, representados pela classe **Documento do Estudo Experimental**. Os documentos podem ser de três tipos: **Instrumento**, **Formulário de Resultados** ou de **Organização do Experimento**. O primeiro define e formaliza os instrumentos para criar meios de executar o experimento e monitorá-lo, sem afetar o controle do experimento. O segundo descreve os resultados na forma de dados e relatórios, obtidos a partir da execução de um experimento. O terceiro é uma generalização dos seguintes documentos: plano do experimento, descrição dos exercícios, notas do curso de treinamento, procedimento do experimento,

glossário e descrição dos parceiros. Cada um destes documentos é um tipo de documento de organização do experimento.

Durante a conversão do modelo de pacote para modelos ontológicos, observou-se a necessidade de evoluir a linguagem LINGO. Esta linguagem considera as classes como sendo apenas conceitos, mas na Ontologia do Pacote do Experimento esses conceitos são também documentos constituintes do pacote. Portanto, visando adicionar maior rigor à linguagem, optamos por incluir o estereótipo <<documento>> para denotar um conceito ontológico que constitui também um documento propriamente dito. Com esta diferenciação, criamos também uma semântica própria para o relacionamento das classes com este estereótipo com classes marcadas com a notação <<conceito>>. Esta nova semântica agrega maior flexibilidade para representar o conhecimento sobre os documentos, tais como definir quais conceitos cada documento registra. A semântica criada é apresentada abaixo.

1. Toda classe marcada com estereótipo <<documento>> só pode possuir subclasses marcadas com o mesmo estereótipo;
2. Toda classe marcada com estereótipo <<documento>> só pode se relacionar via composição com classes marcadas com o mesmo estereótipo;
3. Toda superclasse marcada com o estereótipo <<documento>> não corresponde a um documento físico no pacote de experimento. Neste caso, todas as subclasses mais específicas (nós-folha da hierarquia de classes) representam documentos reais presentes no pacote. Pela figura 4-10, a classe **documento do estudo experimental** define apenas um tipo de documento, mas não representa um documento real do pacote. Suas subclasses **instrumento**, **formulário de resultados** e **documento de organização do experimento** também não definem um documento propriamente dito, pois ainda não são as classes mais especializadas na hierarquia de classes. No caso de um **instrumento**, a sua subclasse mais específica é **artefato do experimento**, que de fato representa um documento real do pacote do experimento.
4. Toda classe marcada com o estereótipo <<documento>> e que é componente de outra classe (relacionamento de composição), representa uma seção dentro do documento referenciado pela classe composta.

5. No caso de um relacionamento de agregação entre classes com estereótipo <<documento>>, o documento representado pela classe-alvo contém como anexo o documento representado pela classe-origem.

O modelo de pacote de Experimento de AMARAL (2003) contém um documento chamado **plano do experimento**, que descreve informações sobre o estudo experimental, como participantes, objetos, diretrizes, hipóteses, variáveis dependentes e independentes, dentre outros. Entretanto, não foi objetivo de AMARAL (2003) mapear todas as possíveis informações que este documento pode registrar. Os domínios de conhecimento relevantes na atividade de planejamento e descritos em (LOPES e TRAVASSOS, 2008) não fizeram parte do escopo de AMARAL (2003) e, portanto, o documento de plano carece de informações deste domínio. Neste sentido, criamos a Subontologia de Planejamento do Experimento (Figura 4-11), que explicita o conhecimento sobre quais informações devem estar presentes no documento de plano para os domínios de (LOPES e TRAVASSOS, 2008). Por questões de escopo, optamos inicialmente por explicitar o conhecimento relativo apenas ao domínio de estratégia de estudo. Nesta subontologia, a classe **plano do estudo experimental**, marcada com o estereótipo <<documento>>, é composta pelas classes **planos de pesquisa-ação**, **survey**, **estudo de caso**, **estudo de observação** e **estudo controlado**, cada qual definido como sendo um documento que está inserido como uma seção do documento de plano de estudo experimental. Os atributos de cada subclasse correspondem a subseções deste documento.

4.2.4.6. Formalização da Associação entre as Ontologias de Pesquisa Científica e Pacote do Experimento

O conhecimento expresso na Ontologia do Pacote do Experimento não está associado ao conhecimento da Ontologia de Pesquisa Científica. Ou seja, não é possível determinar quais conceitos de pesquisa científica um determinado documento do pacote registra. Assim, foram construídos modelos para mapear esse conhecimento que deriva da integração entre as duas ontologias no repositório. Este mapeamento explicita quais seções e subseções um documento possui, no qual cada seção e subseção estão relacionadas a um conceito específico na Ontologia de Pesquisa Científica.

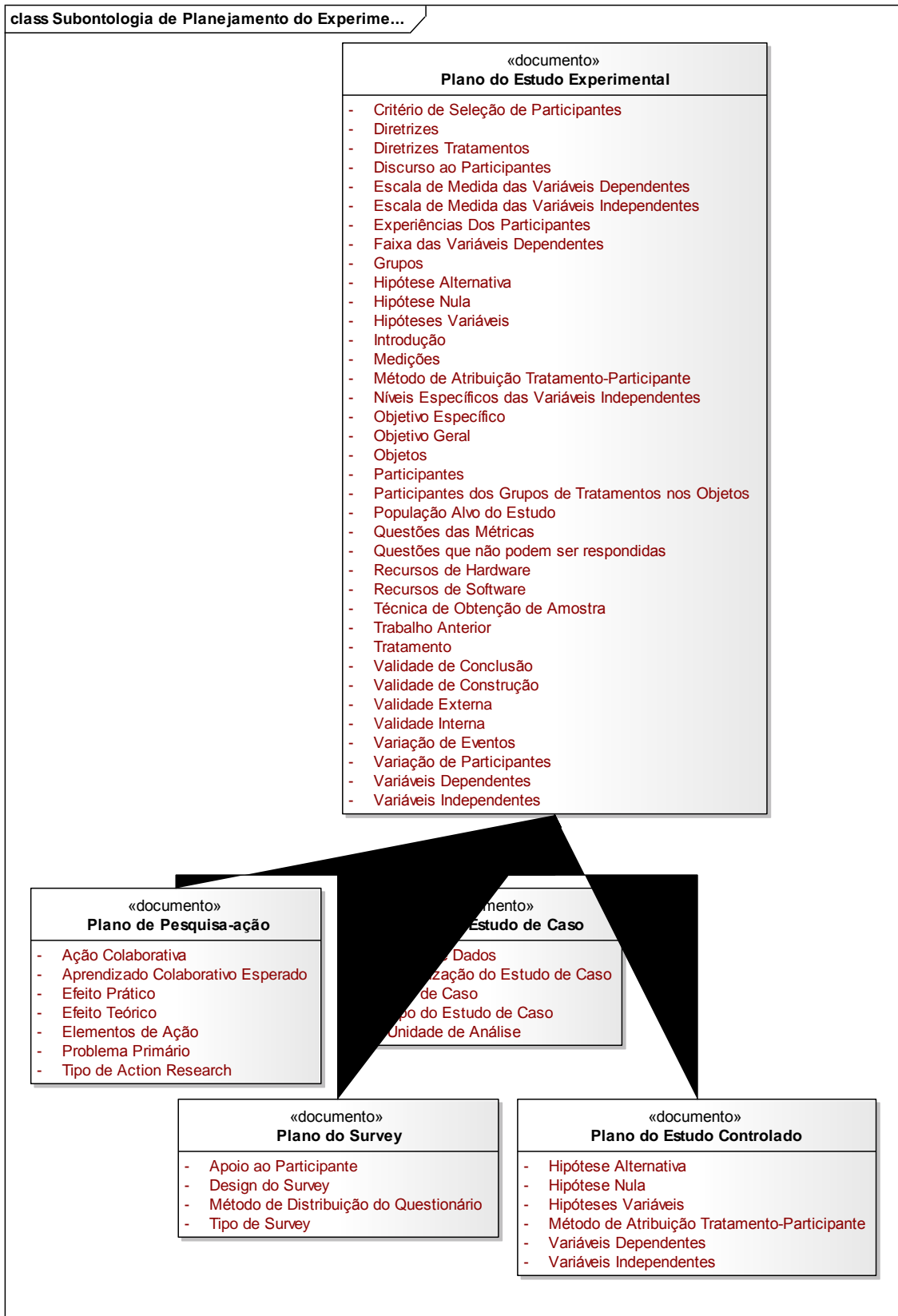


Figura 4-11. Subontologia de Planejamento do Experimento

Em vista disto, não está especificado na linguagem LINGO este tipo de relacionamento entre um documento e os conceitos registrados em seções e subseções.

Neste caso, precisou-se evoluir uma segunda vez a linguagem, adicionando novas semânticas:

1. Para todo atributo de uma classe <<documento>> mapeado na associação desta classe com alguma classe <<conceito>>:
 - a. Caso atributo do documento esteja mapeado em uma associação entre as classes com estereótipo <<documento>> e <<conceito>>, o atributo denota uma seção que documenta o conhecimento explicitado pela classe <<conceito>>. O nome da associação deve ser “documenta” seguido do nome da classe <<conceito>>. Caso o atributo não esteja atrelado a um relacionamento com uma classe de estereótipo <<conceito>>, o atributo ainda assim representa uma seção no documento.
2. Toda a associação direcionada entre classes <<conceito>> e <<documento>>, na qual a primeira é o alvo e não existe atributo na classe de documento mapeado para esta associação, indica que a classe documenta o conhecimento representado pela classe <<conceito>>. Neste caso, o nome da associação deve ser apenas “documenta”.

Para ilustrar como realizar esta associação, optou-se pelo documento de planejamento do experimento para representar quais conceitos da Ontologia de Pesquisa Científica são registrados por este tipo de documento. Além disso, optou-se também pelo subdomínio de estratégia de estudo para realizar esta associação. Foram construídas as seguintes subontologias:

- Subontologia de Documentos de Pesquisa-ação
- Subontologia de Documentos de *Survey*
- Subontologia de Documentos de Estudo de Caso
- Subontologia de Documentos de Estudo de Observação
- Subontologia de Documentos de Estudo Controlado

Essas subontologias estão organizadas dentro da Ontologia de Pesquisa Científica e referenciam os documentos da Ontologia de Pacote do Experimento. A figura abaixo ilustra a subontologia de documentos de *survey* (Figura 3-19), que associa os conceitos de *survey* ao documento **plano do survey**.

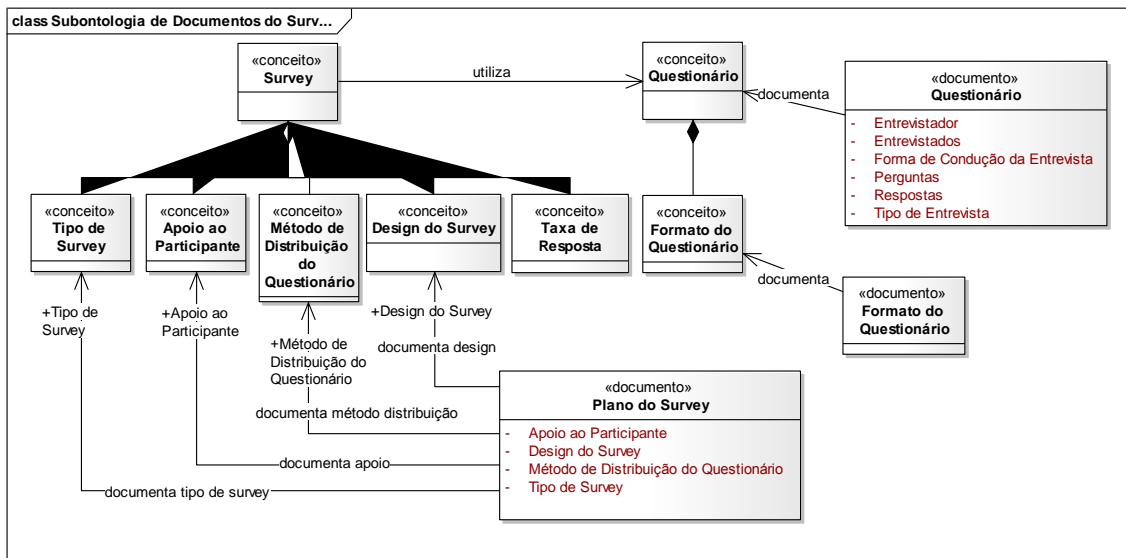


Figura 4-12 - Subontologia de Documentos de Survey

O documento **plano de survey** está associado aos conceitos **tipo de survey**, **apoio ao participante**, **método de distribuição do questionário**, **design do survey** e **taxa de resposta**. Cada associação referencia um atributo específico na classe plano do survey, na qual o nome do atributo define uma seção do documento. Além disto, há o relacionamento *documenta* entre os documentos questionário/formato e seus respectivos conceitos questionário/formato. Neste caso não há seção envolvida: o relacionamento *documenta* apenas explicita que um determinado documento registra o conhecimento relativo a um conceito.

4.2.5. Avaliação

Na etapa de avaliação foram conduzidas sessões de inspeção do glossário e das ontologias, visando identificar eventuais defeitos, conforme descrito no capítulo 3 desta dissertação. O planejamento e a execução da avaliação das ontologias são descritos na seção 4.3.

4.2.6. Implementação

Após a avaliação dos modelos e correção dos defeitos encontrados, a ontologia é codificada através da representação dos conceitos e relacionamentos utilizando uma linguagem formal para ontologias, tornando-a assim passível de processamento. Esta etapa foi realizada para que fosse desenvolvido o protótipo da ferramenta que consulta o repositório. Um estudo sobre linguagens de representação de ontologias foi conduzido para identificar aquela mais apropriada. Um sistema de gerência de ontologias foi

escolhido e utilizado para apoiar o processo de implementação dos modelos. Maiores detalhes sobre a linguagem, o sistema de gerência de ontologias e a ferramenta que atua como prova de conceito de utilização do repositório encontram-se no próximo capítulo desta dissertação.

4.2.7. Integração

Na etapa de integração questionou-se a possibilidade de reutilização de ontologias existentes. Não foi objetivo neste processo reutilizar outras eventuais ontologias de Experimentação em Engenharia de Software. O foco esteve na construção de modelos cujo conhecimento seja aderente ao domínio de Experimentação em Engenharia de Software, em especial ao subdomínio das diferentes categorias de estudos experimentais.

4.2.8. Documentação

Por fim, na etapa de documentação o planejamento e a execução do processo foram descritos, detalhando a construção dos artefatos em cada etapa. Foge ao escopo deste trabalho fornecer detalhes de cronograma e outras informações que não estão relacionadas diretamente ao conteúdo dos modelos.

4.3. Revisão das Ontologias do Repositório

A revisão dos modelos ontológicos foi conduzida no mesmo contexto da revisão do glossário, utilizando-se para isso do mesmo conjunto de participantes da disciplina de Experimentação em Engenharia de Software, ministrada no Programa de Engenharia de Sistemas e Computação da COPPE/UFRJ. Um dos objetivos da inspeção do glossário (Capítulo 3) por cada equipe foi de prepará-las para a inspeção dos modelos conceituais, já que todos estariam familiarizados com a terminologia do domínio de experimentação em Engenharia de Software. Além disso, cada equipe obteve conhecimento acerca de uma estratégia de estudo específica, capacitando-as para a inspeção de modelos conceituais específicos.

Na avaliação das ontologias, os inspetores foram orientados a adotar uma perspectiva crítica, de modo que oportunidades de melhoria pudessem ser identificadas e propostas de alteração no modelo fossem elaboradas. Para cada proposta, instruímos o inspetor a fornecer uma justificativa e a(s) referência(s) que subsidia(m) a proposta.

A taxonomia de defeitos sugerida foi:

- Omissão de conceitos e propriedades;
- Omissão de descrição de conceitos;
- Descrição incorreta de conceitos;
- Relacionamentos inexistentes entre conceitos;
- Relacionamentos cujo nome (descrição sobre a linha) e cardinalidade estejam ausentes e isto represente, de fato, uma inconsistência no modelo;
- Relacionamentos incorretos entre conceitos. Exemplos de defeitos: tipo de relacionamento incorreto (composição ao invés de agregação), descrição incorreta, cardinalidade errada e entre outros;
- Omissão de modelos. Caso algum conceito seja complexo, o inspetor pode julgar necessário um modelo em separado para este conceito, a fim de organizar melhor o conhecimento. Adicionalmente, pode ser identificada também a oportunidade de incluir um modelo novo e que não esteja relacionado a um conceito específico.

A inspeção foi organizada como uma tarefa individual. Cada inspetor avaliou as subontologias que expressam o conhecimento sobre o qual foi pesquisado durante a inspeção do glossário de termos. Além disso, cada inspetor avaliou uma das subontologias de método de pesquisa, que explicitam conhecimento importante sobre métodos de análise de dados. As equipes de pesquisa-ação, *survey* e estudo de caso inspecionaram a subontologia de método qualitativo em função deste ser empregado com frequência no contexto desses tipos de estratégia experimental, cuja natureza é qualitativa (EASTERBROOK *et al.*, 2008). A equipe de estudo controlado inspecionou, além da subontologia de estudo controlado, a subontologia de método de quantitativo, também frequentemente aplicada no contexto dessa estratégia (EASTERBROOK *et al.*, 2008). Como no momento da inspeção ainda não haviam sido desenvolvidas as subontologias de estudo *in vivo*, *in vitro*, *in virtuo* e *in silico*, a equipe de estudos baseados em simulação foi designada para inspecionar os mesmos modelos alocados para a equipe de estudo controlado. É importante ressaltar que os inspetores 13 e 14 abandonaram a disciplina e não concluíram a atividade de inspeção da subontologia de estudo de observação. A tabela 4-1 sumariza os modelos inspecionados por cada indivíduo, incluindo também os inspetores com a mesma identificação atribuída na inspeção do glossário.

Tabela 4-1. Tabela das subontologias inspecionadas por equipe

Equipe	Inspetor	Subontologias para inspeção
Pesquisa-ação e método de pesquisa qualitativo	Inspetor 1	<ul style="list-style-type: none"> • Subontologia de pesquisa-ação • Subontologia de documentos de pesquisa-ação • Subontologia de método qualitativo
	Inspetor 2	
	Inspetor 3	
	Inspetor 4	
<i>Survey</i> e método de pesquisa qualitativo	Inspetor 5	<ul style="list-style-type: none"> • Subontologia de <i>survey</i> • Subontologia de documentos de <i>survey</i> • Subontologia de método qualitativo
	Inspetor 6	
	Inspetor 7	
Estudo de Caso e método de pesquisa qualitativo	Inspetor 8	<ul style="list-style-type: none"> • Subontologia de estudo de caso • Subontologia de documentos de estudo de caso • Subontologia de método qualitativo
	Inspetor 9	
	Inspetor 10	
	Inspetor 11	
Estudo de Observação e método de pesquisa qualitativo	Inspetor 12	<ul style="list-style-type: none"> • Subontologia de estudo de observação • Subontologia de método qualitativo
	Inspetor 13	
Estudo Controlado e método de pesquisa quantitativo	Inspetor 16	<ul style="list-style-type: none"> • Subontologia de estudo controlado • Subontologia de documentos de estudo controlado • Subontologia de método quantitativo
	Inspetor 17	
	Inspetor 18	
	Inspetor 19	
	Inspetor 21	
	Inspetor 22	

4.3.1. Inspeção dos Modelos de Pesquisa-ação

A inspeção das subontologias de Pesquisa-ação e de documento de Pesquisa-ação revelou em maior número a omissão de conceitos e de relacionamentos. Na Subontologia de Pesquisa-ação, os principais conceitos omitidos dizem respeito aos tipos de **natureza de projeto de pesquisa-ação** e ao **modo de participação** dos envolvidos no estudo. A ausência desses conceitos também foi apontada na Subontologia de Documentos de Pesquisa-ação, sendo que nesta ainda foi apontada a

omissão de quatro propriedades no documento de Plano de Pesquisa-ação, indicando a necessidade das seções de **modo de participação**, **natureza do projeto**, **problema primário** e **elementos de ação**. A maior parte dos relacionamentos propostos ocorreu entre os conceitos omissos. A exceção foi para os conceitos já existentes de **ação colaborativa** e **elementos de ação**. Neste caso, foi apontado que ação colaborativa é *descrita por elementos de ação*. As tabelas 4-2 e 4-3 apresentam o número de discrepâncias por inspetor que foram identificadas como sendo defeitos para cada tipo estabelecido na taxonomia.

Tabela 4-2. Número de defeitos identificados após discriminação para os modelos de pesquisa-ação (Parte 1)

Número de defeitos identificados após discriminação			
Inspetor	Omissão de conceitos	Omissão de propriedades	Omissão de descrição de conceitos
Inspetor 1	5	0	3
Inspetor 2	6	0	4
Inspetor 3	4	4	0
Inspetor 4	3	0	0
Total	18	4	7

Tabela 4-3. Número de defeitos identificados após discriminação para os modelos de pesquisa-ação (Parte 2)

Número de defeitos identificados após discriminação				
Inspetor	Omissão de relacionamentos	Omissão de modelos	Descrição incorreta de conceitos	Relacionamentos incorretos
Inspetor 1	9	0	0	0
Inspetor 2	4	0	1	0
Inspetor 3	4	0	0	2
Inspetor 4	0	0	1	0
Total	17	0	2	2

4.3.2. Inspeção dos Modelos de Survey

Dentre os defeitos detectados nas subontologia de *survey* e documentos de *survey*, destacamos a omissão dos conceitos **questão aberta** e **fechada**, que são tipos de formatos de questionário. A propriedade apontada como omissa é pertinente ao documento de plano de survey e está relacionada ao conceito de **formato do questionário**. Demais defeitos sobre omissão de descrição de conceitos e descrições

incorretas foram devidamente corrigidos nas subontologias. As tabelas 4-4 e 4-5 apresentam o número de discrepâncias por inspetor que foram identificadas como sendo defeitos para cada tipo estabelecido na taxonomia.

Tabela 4-4. Número de defeitos identificados após discriminação para os modelos de survey (Parte 1)

Número de defeitos identificados após discriminação			
Inspetor	Omissão de conceitos	Omissão de propriedades	Omissão de descrição de conceitos
Inspetor 5	0	0	0
Inspetor 6	2	0	0
Inspetor 7	0	1	10
Total	2	1	10

Tabela 4-5. Número de defeitos identificados após discriminação para os modelos de survey (Parte 2)

Número de defeitos identificados após discriminação				
Inspetor	Omissão de relacionamentos	Omissão de modelos	Descrição incorreta de conceitos	Relacionamentos incorretos
Inspetor 5	0	0	8	0
Inspetor 6	0	0	11	0
Inspetor 7	0	0	0	0
Total	0	0	19	0

4.3.3. Inspeção dos Modelos de Estudo de Caso

A inspeção da Subontologia de Estudo de Caso revelou um grande número omissões de descrição de conceitos. O glossário foi consultado para verificar as descrições correspondentes. Além disso, várias omissões foram apontadas, dentre as quais destacam-se os conceitos **estudo de caso causal** e **explanatório, organização, estrutura de relatório** e seus respectivos tipos. Os relacionamentos omissos ocorrem entre os conceitos apontados como ausentes. Quanto aos relacionamentos incorretos, os inspetores observaram a necessidade de alterar o nome dos relacionamentos dos conceitos **estudo de caso confirmatório** e **causal** com o conceito **positivismo**. Este relacionamento passou a ser rotulado como “conduzido sobre”. Na Subontologia de Documentos de Estudo de Caso, os defeitos detectados compartilham de todos os conceitos omissos na Subontologia de Estudo de Caso. As tabelas 4-6 e 4-7 apresentam o número de discrepâncias por inspetor que foram identificadas como sendo defeitos para cada tipo estabelecido na taxonomia.

Tabela 4-6. Número de defeitos identificados após discriminação para os modelos de estudo de caso (Parte 1)

Número de defeitos identificados após discriminação			
Inspetor	Omissão de conceitos	Omissão de propriedades	Omissão de descrição de conceitos
Inspetor 8	2	0	0
Inspetor 9	0	0	18
Inspetor 10	3	0	0
Inspetor 11	13	0	0
Total	18	0	18

Tabela 4-7. Número de defeitos identificados após discriminação para os modelos de estudo de caso (Parte 2)

Número de defeitos identificados após discriminação				
Inspetor	Omissão de relacionamentos	Omissão de modelos	Descrição incorreta de conceitos	Relacionamentos incorretos
Inspetor 8	6	0	6	1
Inspetor 9	0	0	1	2
Inspetor 10	3	0	0	0
Inspetor 11	13	0	1	1
Total	22	0	8	4

4.3.4. Inspeção dos Modelos de Estudo de Observação

Os inspetores das subontologias de estudo de observação e de documentos de estudo de observação não identificaram grande número de defeitos, ainda que esses modelos contenham poucos conceitos e relacionamentos. Uma possível explicação para isto pode estar no tipo de discrepâncias apontadas pelos inspetores. Grande parte delas foi descartada após discriminação em função de estarem relacionadas a conceitos de estudo de caso. Os inspetores podem ter sido influenciados por suas próprias experiências na execução de estudos de caso, cuja definição foi confundida com a de estudo de observação. Isso acarretou num descarte considerável de discrepâncias. Os defeitos mais relevantes que foram detectados envolvem a ausência de relacionamento entre os conceitos **etnografia** e **comunidade**, além dos relacionamentos de **estudo de viabilidade** com os conceitos de **tecnologia** e **técnica**. As tabelas 4-8 e 4-9 apresentam o número de discrepâncias por inspetor que foram identificadas como sendo defeitos para cada tipo estabelecido na taxonomia.

Tabela 4-8. Número de defeitos identificados após discriminação para os modelos de estudo de observação (Parte 1)

Número de defeitos identificados após discriminação

Inspetor	Omissão de conceitos	Omissão de propriedades	Omissão de descrição de conceitos
Inspetor 12	0	0	2
Inspetor 13	0	0	2
Total	0	0	4

Tabela 4-9. Número de defeitos identificados após discriminação para os modelos de estudo de observação (Parte 2)

Número de defeitos identificados após discriminação				
Inspetor	Omissão de relacionamentos	Omissão de modelos	Descrição incorreta de conceitos	Relacionamentos incorretos
Inspetor 12	0	0	8	0
Inspetor 13	3	0	1	3
Total	3	0	9	3

4.3.5. Inspeção dos Modelos de Estudo Controlado

Para as subontologias de Estudo Controlado e de Documentos de Estudo Controlado, a inspeção revelou numerosas oportunidades de melhoria. Após discriminação, os defeitos mais impactantes são de omissão de conceitos e de relacionamentos, que totalizaram 16 e 25 respectivamente. Dentre os conceitos omissos, destacam-se os conceitos de **projeto de estudo controlado**, **princípios de projeto** e **método de atribuição dos tratamentos aos participantes**, cada qual com seus respectivos subtipos. As tabelas 4-10 e 4-11 apresentam o número de discrepâncias por inspetor que foram identificadas como sendo defeitos para cada tipo estabelecido na taxonomia.

Tabela 4-10. Número de defeitos identificados após discriminação para os modelos de estudo controlado (Parte 1)

Número de defeitos identificados após discriminação			
Inspetor	Omissão de conceitos	Omissão de propriedades	Omissão de descrição de conceitos
Inspetor 16	0	7	0
Inspetor 17	5	0	0
Inspetor 18	2	10	2
Inspetor 19	2	2	2
Inspetor 20	1	0	6
Inspetor 21	5	0	0
Inspetor 22	1	0	12
Total	16	19	22

Tabela 4-11. Número de defeitos identificados após discriminação para os modelos de estudo controlado (Parte 2)

Inspetor	Número de defeitos identificados após discriminação			
	Omissão de relacionamentos	Omissão de modelos	Descrição incorreta de conceitos	Relacionamentos incorretos
Inspetor 8	2	0	0	0
Inspetor 9	9	0	0	2
Inspetor 10	4	0	4	0
Inspetor 11	3	0	1	1
Inspetor 18	4	0	0	0
Inspetor 19	3	0	0	0
Inspetor 20	0	0	2	0
Total	25	0	7	3

4.3.6. Inspeção do Modelo de Método Qualitativo

O tipo de defeito com maior incidência na inspeção da subontologia de método qualitativo corresponde à omissão de descrição de defeitos que, de fato, comprovam a carência de definição para diversos termos do modelo. A omissão das propriedades **método de observação**, **observador** e **observados** ocorreu no documento **nota de observação de participantes**. Os conceitos omissos indicados pelos inspetores foram os tipos de formas de condução de entrevista, que são **pessoal**, **por telefone** e **por internet**. A baixa incidência de omissão de conceitos talvez se explique pelo grande número de conceitos no modelo, que totaliza 61. As tabelas 4-12 e 4-13 apresentam o número de discrepâncias por inspetor que foram identificadas como sendo defeitos para cada tipo estabelecido na taxonomia.

Tabela 4-12. Número de defeitos identificados após discriminação para o modelo de método qualitativo (Parte 1)

Inspetor	Número de defeitos identificados após discriminação		
	Omissão de conceitos	Omissão de propriedades	Omissão de descrição de conceitos
Inspetor 1	0	0	31
Inspetor 2	0	0	0
Inspetor 3	1	0	1
Inspetor 4	0	0	0
Inspetor 5	0	0	0
Inspetor 6	0	0	0
Inspetor 7	0	0	25
Inspetor 8	0	0	0
Inspetor 9	0	0	29
Inspetor 10	1	0	0
Inspetor 11	0	0	14

Inspetor 12	0	1	15
Inspetor 13	1	1	2
Total	3	3	117

Tabela 4-13. Número de defeitos identificados após discriminação para o modelo de método qualitativo (Parte 2)

Inspetor	Número de defeitos identificados após discriminação			
	Omissão de relacionamentos	Omissão de modelos	Descrição incorreta de conceitos	Relacionamentos incorretos
Inspetor 1	0	0	0	0
Inspetor 2	0	0	0	0
Inspetor 3	0	0	0	0
Inspetor 4	0	0	0	0
Inspetor 5	0	0	0	0
Inspetor 6	0	0	3	0
Inspetor 7	0	0	0	0
Inspetor 8	0	0	0	0
Inspetor 9	0	0	0	0
Inspetor 10	0	0	0	0
Inspetor 11	0	0	0	0
Inspetor 12	0	0	6	0
Inspetor 13	0	0	0	0
Total	0	0	9	0

4.3.7. Inspeção do Modelo de Método Quantitativo

A inspeção conduzida na subontologia de método quantitativo revelou importantes oportunidades de melhoria no que tange à cobertura de conceitos e relacionamentos. Foram identificados 11 e 12 novos conceitos e relacionamentos, respectivamente. Dentre os conceitos omissos, podemos citar como mais relevantes os tipos de teste paramétrico (**binomial**, **chi-2**, **ANOVA** e todos os seus subtipos), não-paramétricos (**Wilcoxon**, **Kruskal-Wallis**, **Mann-Whitney**, **Kolmogorov-Smirnov**) e os tipos de medida de tendência central (**média**, **moda**, **mediana**, **valor máximo**, **valor mínimo**, **percentil** e **quartil**). A maior parte dos relacionamentos omissos deriva destes novos conceitos. As tabelas 4-14 e 4-15 apresentam o número de discrepâncias por

inspetor que foram identificadas como sendo defeitos para cada tipo estabelecido na taxonomia.

Tabela 4-14. Número de defeitos identificados após discriminação para o modelo de método quantitativo (Parte 1)

Número de defeitos identificados após discriminação			
Inspetor	Omissão de conceitos	Omissão de propriedades	Omissão de descrição de conceitos
Inspetor 16	2	0	0
Inspetor 17	0	0	0
Inspetor 18	5	0	4
Inspetor 19	0	0	5
Inspetor 20	4	0	5
Inspetor 21	0	0	0
Inspetor 22	0	0	15
Total	11	0	29

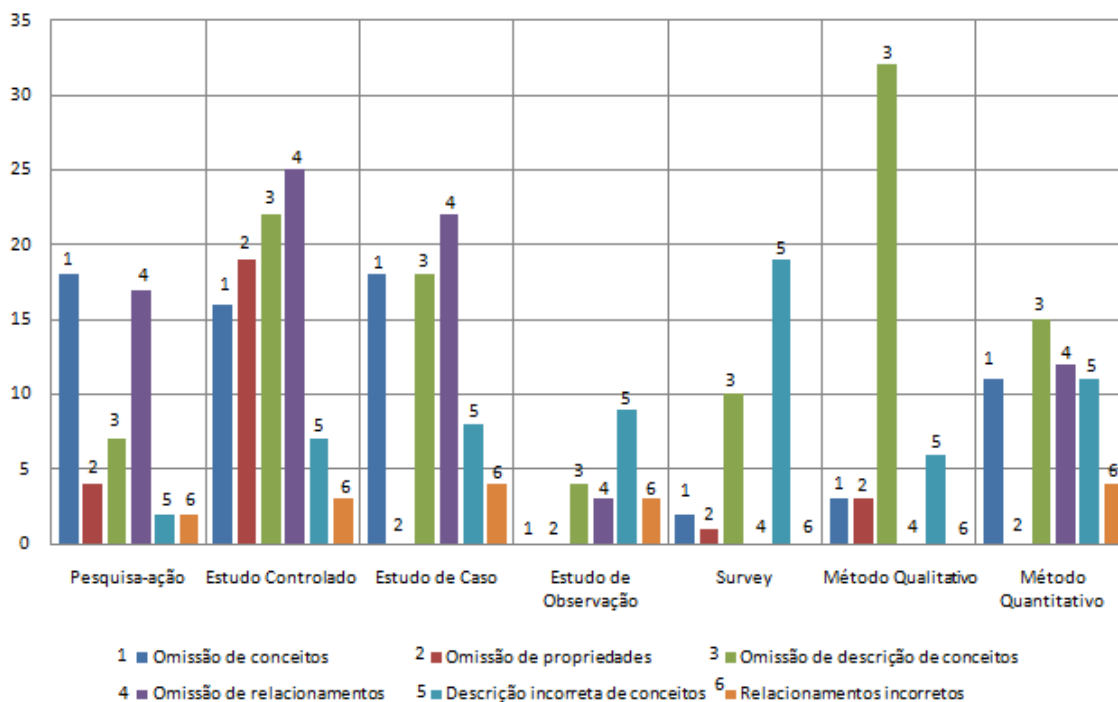
Tabela 4-15. Número de defeitos identificados após discriminação para o modelo de método quantitativo (Parte 2)

Número de defeitos identificados após discriminação				
Inspetor	Omissão de relacionamentos	Omissão de modelos	Descrição incorreta de conceitos	Relacionamentos incorretos
Inspetor 8	3	0	0	0
Inspetor 9	1	0	1	1
Inspetor 10	2	0	3	0
Inspetor 11	3	0	0	0
Inspetor 18	3	0	7	3
Inspetor 19	0	0	0	0
Inspetor 20	0	0	0	0
Total	12	0	11	4

4.3.8. Síntese dos resultados

Após o término da inspeção, foi dado início à atividade de discriminação das discrepâncias apontadas. O procedimento para descarte de discrepâncias se deu avaliando dois aspectos: propostas sem referências que subsidiassem a afirmação foram descartadas e, para aquelas que a indicavam, foi analisada a referência em busca da explicação da justificativa informada. Os dados para cada subontologia foram sintetizados na tabela 4-16, que quantifica os defeitos identificados após discriminação.

Tabela 4-16. Síntese do resultado (número de defeitos) da inspeção dos modelos conceituais



Os defeitos identificados nas inspeções foram devidamente corrigidos nos modelos, incorporando novos conceitos, propriedades e relacionamentos. Além disso, os conceitos expressos nos modelos passaram a ter significado mais preciso, fruto da inspeção do glossário, com o qual as ontologias estão diretamente relacionada e, portanto, semanticamente dependentes.

Os defeitos mais importantes do ponto de vista de completude (omissão de conceitos e relacionamentos) totalizaram 147, o que revela uma contribuição importante no sentido de melhorar a cobertura dos modelos quanto a conceitos e relacionamentos. Do ponto de vista de corretude, o defeito de descrição incorreta de conceitos totalizou 65, o que também demonstra como os modelos careciam de uma revisão mais detalhada quanto à definição dos conceitos.

4.4. Conclusões

Este capítulo o processo adotado para construção dos modelos ontológicos, baseado no processo *Methodology* (FERNANDÉZ *et al.*, 1997). A construção dos modelos considerou como ponto de partida a Ontologia de Pesquisa Científica de Biolchini *et al.* (2006), cujo foco está em explicitar o conhecimento sobre Revisão Sistemática. A etapa de conceitualização do processo procedeu com a inclusão de

termos ontologia de Biolchini *et al.* (2006) a partir do glossário, uma vez que este é um dos mecanismos de representação de conhecimento do repositório. Na etapa de formalização, pesquisou-se um conjunto de referências conhecidas em Engenharia de Software Experimental para identificar novos relacionamentos e atributos para os conceitos da ontologia evoluída na etapa de conceitualização. Visando agregar maior expressividade e rigor à representação da ontologia e melhorar sua compreensibilidade, a ontologia foi reescrita utilizando a linguagem LINGO (MIAN, 2003), baseada em UML.

Além disto, o repositório deve preservar o conhecimento sobre o modelo de pacote de estudos de AMARAL (2003). A Ontologia de Pacote de Experimento foi construída a partir do modelo de AMARAL (2003), utilizando para isto a linguagem LINGO (MIAN, 2003). Nesta conversão, a linguagem LINGO foi evoluída para representar conhecimento envolvendo conceitos e documentos, sendo agregada a linguagem novos estereótipos e semântica específicos. Na semântica adicionada à linguagem, um determinado documento da Ontologia de Pacote do Experimento está associado a um conceito da Ontologia de Pesquisa Científica, significando que o documento contém uma seção que registra este conceito. Novos modelos foram construídos para representar essa associação, sendo agregados à Ontologia de Pesquisa Científica.

No capítulo seguinte é apresentado o protótipo da ferramenta construída para ressaltar como o repositório pode ser consultado a fim de disponibilizar conhecimento ao pesquisador durante a etapa de planejamento. Este conhecimento envolve as diversas tomadas de decisão desta etapa sobre os tipos de estratégia de estudo existentes. Antes da apresentação do protótipo, o capítulo seguinte aborda também como foi realizada a análise de linguagens de representação e gerência de ontologias, necessária para concluir a formalização do conhecimento. Esta análise foi conduzida para fundamentar a escolha de uma linguagem formal para codificar as ontologias, conforme previsto na etapa de implementação do processo adotado para construção dos modelos. Uma ferramenta de gerência de ontologias também foi escolhida para apoiar a implementação.

Capítulo 5

Ferramenta para Configuração de Modelos de Pacote de Experimentos em Engenharia de Software

Este capítulo apresenta o protótipo da ferramenta que processa as ontologias do repositório para consultá-lo, apoiando o pesquisador em tomadas de decisão e configuração de modelos de pacote de experimento na etapa de planejamento

5.1. Introdução

O glossário de termos e as ontologias apresentados nos capítulos anteriores constituem um modelo para o repositório de conhecimento de Experimentação em Engenharia de Software. Um dos objetivos, conforme previamente descrito, com a construção deste repositório é explorá-lo utilizando ferramentas inteligentes (ABECKER *et al.*, 1999). Em vista disto, apresentamos um protótipo de ferramenta que ilustra como o repositório pode ser utilizado para auxiliar o pesquisador na etapa de planejamento, na qual o acesso ao conhecimento sobre os tipos de estudos experimentais pode contribuir para reduzir a inserção de ameaças a validade do estudo. A ferramenta implementa um mecanismo que consulta o repositório para extrair o conhecimento sobre experimentação e, conseqüentemente, uma série de tomadas de decisão pertinentes à etapa de planejamento. Exemplos de tomadas de decisão incluem qual estratégia de estudo adotar, qual método de pesquisa deve ser utilizado para análise de dados, entre outras. Para apoiar tomadas de decisão da etapa de planejamento, o mecanismo da ferramenta extrai do repositório o conhecimento sobre as decisões na forma de pergunta e respostas. As respostas dadas pelos usuários são registradas em um *template* de documento de plano do estudo experimental, que pode continuar sendo editado ao longo da execução do Processo de Experimentação.

Para consultar o repositório, este mecanismo precisa acessar os modelos ontológicos escritos em linguagem formal, isto é, compatíveis com processamento automatizado. Assim, as ontologias descritas no capítulo anterior foram formalizadas utilizando uma linguagem de representação de ontologias específica. Esta formalização foi realizada utilizando uma ferramenta de gerência de ontologias, que codifica os

modelos em uma linguagem de representação de ontologias. As linguagens e ferramentas disponíveis foram avaliadas individualmente segundo critérios previamente estabelecidos.

Este capítulo descreve na próxima seção o protótipo de ferramenta construído e, em seguida, discorre sobre detalhes de implementação da ontologia. Em seguida, um exemplo de utilização da ferramenta é fornecido.

5.2. Protótipo da Ferramenta para Configuração de Modelos de Pacote de Experimento

Um dos objetivos desta dissertação é apresentar uma proposta de utilização do repositório como núcleo de recuperação de conhecimento na etapa de planejamento, na qual o pesquisador deve definir diferentes características do estudo e que descrevem como o mesmo será executado. Foi escolhida por ser uma etapa na qual a ausência de conhecimento disponível ao pesquisador apresenta maiores riscos de introdução de ameaças à validade do estudo. A definição das características do estudo requer que o pesquisador tome diferentes decisões. Por exemplo, ao definir qual será a abordagem do estudo, o pesquisador precisa optar por realizar um estudo primário ou secundário (JURISTO e MORENO, 2001).

Em vista disto, há diferentes conceitos importantes do domínio de experimentação que devem ser considerados pelo pesquisador durante a etapa de planejamento. Ao não considerar um determinado conceito, a execução do estudo pode vir a ser comprometida ou o resultado pode ser ameaçado em termos de sua validade. Por exemplo, caso o pesquisador não defina previamente quais tipos de método de pesquisa serão empregados, os dados do estudo podem ser incorretamente avaliados. No caso de um estudo que empregue simulação, um método de pesquisa semi-quantitativo pode ser mais adequado do que puramente quantitativo. Neste caso, o pesquisador está diante de uma tomada de decisão sobre quais tipos de métodos de pesquisa devem ser empregados. As decisões foram explicitadas nos conceitos da ontologia que possuem conceitos-filhos. Neste caso, a natureza da decisão envolve escolher qual dos conceitos-filhos deve ser considerado no estudo em questão.

Todas as decisões envolvidas no processo, além de serem importantes no sentido de garantir um planejamento adequado, também são úteis na geração documento de plano do estudo. As características de cada estudo exigem que o plano contemple seções

específicas sobre essas características para que o pesquisador as preencha. Por exemplo, ao definir que os dados do estudo serão avaliados quantitativamente, o documento de plano do experimento deve contemplar seções específicas ao método de pesquisa quantitativo, para que sejam fornecidas pelo pesquisador informações como o *p-value* adotado e testes estatísticos que serão utilizados.

Acessar o repositório no contexto da etapa de planejamento pode contribuir no sentido de tornar disponível ao pesquisador o conhecimento sobre os diferentes conceitos de experimentação associados a importantes tomadas de decisão nesta etapa. Para viabilizar a recuperação de conhecimento, propomos um protótipo de ferramenta que consulta o repositório em busca de conceitos associados a importantes decisões na atividade de planejamento. O protótipo recupera na ontologia do repositório quais decisões devem ser consideradas e expõe ao pesquisador na forma de perguntas, cada qual com um conjunto de opções de resposta. O pesquisador responde à pergunta escolhendo uma determinada opção e a ferramenta registra a escolha. Neste caso, o repositório atua também como um guia para o planejamento, uma vez que a o protótipo navega através dos relacionamentos entre os diferentes conceitos procurando por pontos de decisão. Este guia de condução é citado por SJØBERG *et al.* (2007) como sendo uma das ações a serem tomadas no sentido de termos estudos com melhor qualidade. Quando a navegação pelo modelo ontológico for concluída, o protótipo fornece um *template* de documento de planejamento do estudo com um conjunto de seções aderentes às opções escolhidas pelo pesquisador para que informações específicas sejam fornecidas.

Para que o protótipo seja desenvolvido, o repositório precisa estar preparado para que o conhecimento possa ser extraído diretamente. Entretanto, os modelos conceituais, concebidos na etapa formalização do processo adotado para construção da ontologia, não são facilmente processados automaticamente, uma vez que a linguagem LINGO (MIAN, 2003) utilizada não foi concebida com o objetivo de produzir modelos processáveis. Neste caso, o processo de construção da ontologia prevê a execução da etapa de implementação. Nesta etapa, os modelos conceituais da fase de formalização são codificados utilizando uma linguagem que torna os modelos processáveis, permitindo assim a implementação de mecanismos que percorram os conceitos a partir da rede de relacionamentos entre os mesmos.


As seções subsequentes discorrem sobre detalhes a respeito da implementação da ontologia, a estratégia utilizada pelo protótipo para recuperar o conhecimento da ontologia e um exemplo de aplicação da ferramenta.

5.5. Implementação da Ontologia

Adotamos por conveniência a linguagem OWL (W3C, 2010) e o sistema de gerência de ontologias *Protégé* (PROTÉGÉ, 2010) para implementação dos modelos. O grupo ESE já possuía experiência na construção de ontologias utilizando esta linguagem e ferramenta. Outro fator que levou a esta escolha é que OWL é bem adotada no meio acadêmico, e a ferramenta *Protégé* gera modelos nesta linguagem e possui ampla aceitação por grupos de pesquisa em Gestão do Conhecimento. A versão do *Protégé* utilizada foi a 4.0 beta, a mais atual no momento em que a etapa de implementação foi executada. A versão do mecanismo de inferência OWL utilizada é a Fact++, nativa da versão do *Protégé*. A codificação dos modelos envolve transformar determinados relacionamentos e propriedades da notação LINGO (baseada integralmente em elementos UML) para a notação OWL.

Para transformar os modelos LINGO em modelos OWL, desenvolvemos uma heurística de conversão de alguns elementos de representação UML para elementos OWL. Nesta heurística, os principais elementos da notação UML são representados em um elemento OWL específico, conforme sintetiza a tabela abaixo:

Tabela 5-1 - Transformação de elementos LINGO (UML) em elementos OWL

Elemento LINGO (UML)		Elemento OWL
Classe com estereótipo “conceito” e documento		<i>Class</i>
Atributo cujo tipo é primitivo e string		<i>Data Property</i>
Generalização/Especialização entre classes com estereótipo “conceito”		Hierarquização
Agregação/ Composição entre classes com estereótipo “conceito”		<i>Object Property</i>
Agregação/Composição entre classes com estereótipo “documento”		<i>Object Property</i>
Associação		<i>Object Property</i>
Atributo de classes com estereótipo “documento”		<i>Object Property</i>

5.5.1. Classes com estereótipos “conceito” e “documento”

A heurística adotada transforma classes com estereótipos “conceito” e “documento” em elementos do tipo *class*. O nome da classe em OWL não apresenta espaços e a inicial de cada palavra está em maiúsculo. A figura 5-1 ilustra a classe UML “Abordagem de Pesquisa Científica” (à esquerda) e sua versão em classe OWL na hierarquia de classes do *Protégé* (à direita).

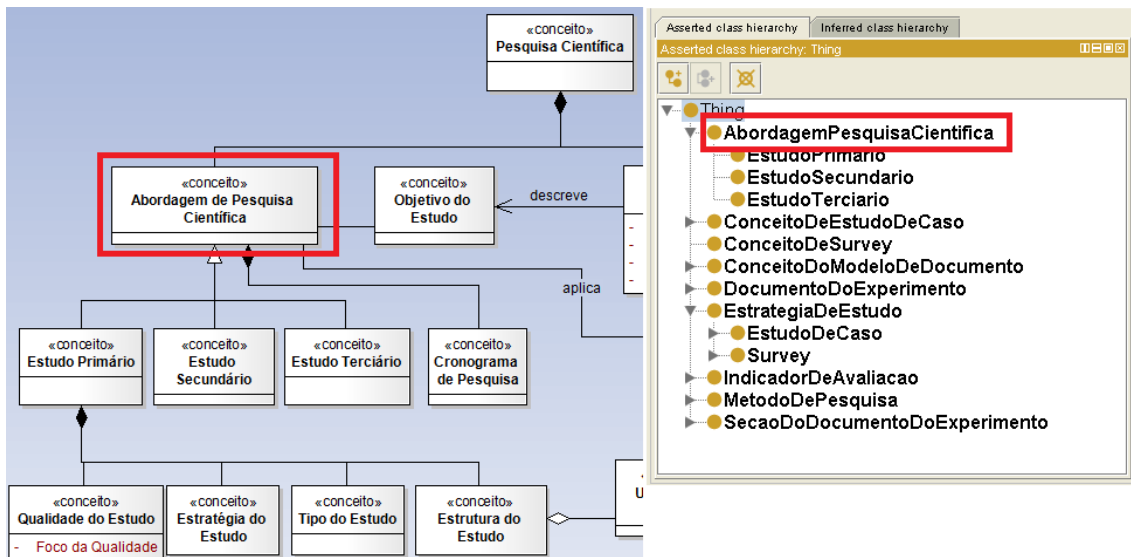


Figura 5-1. Classe UML com estereótipo "conceito" transformada em classe OWL

5.5.2. Atributo cujo tipo é primitivo e string

Os atributos cujo tipo é um primitivo e string foram transformados em elementos *data property*, que estabelecem o relacionamento de um elemento *class* com um dado primitivo. O nome do *data property* apresenta o prefixo “has”, sendo sucedido do nome do atributo sem espaços e com as iniciais de cada palavra em maiúsculo. O *data property* é criado com domínio (*domain*) correspondendo à classe OWL da classe UML que possui o atributo em questão. O alcance (*range*) é configurado para ser do tipo primitivo especificado no modelo UML (inteiro, string e entre outros). A figura 5-2 ilustra o exemplo para a classe UML “Qualidade do Estudo”, que possui como atributo “Foco da Qualidade”. A *data property* “hasFocoDaQualidade” possui como domínio a classe OWL “QualidadeDoEstudo” (relacionada à classe UML “Qualidade do Estudo”) e alcance *string*.

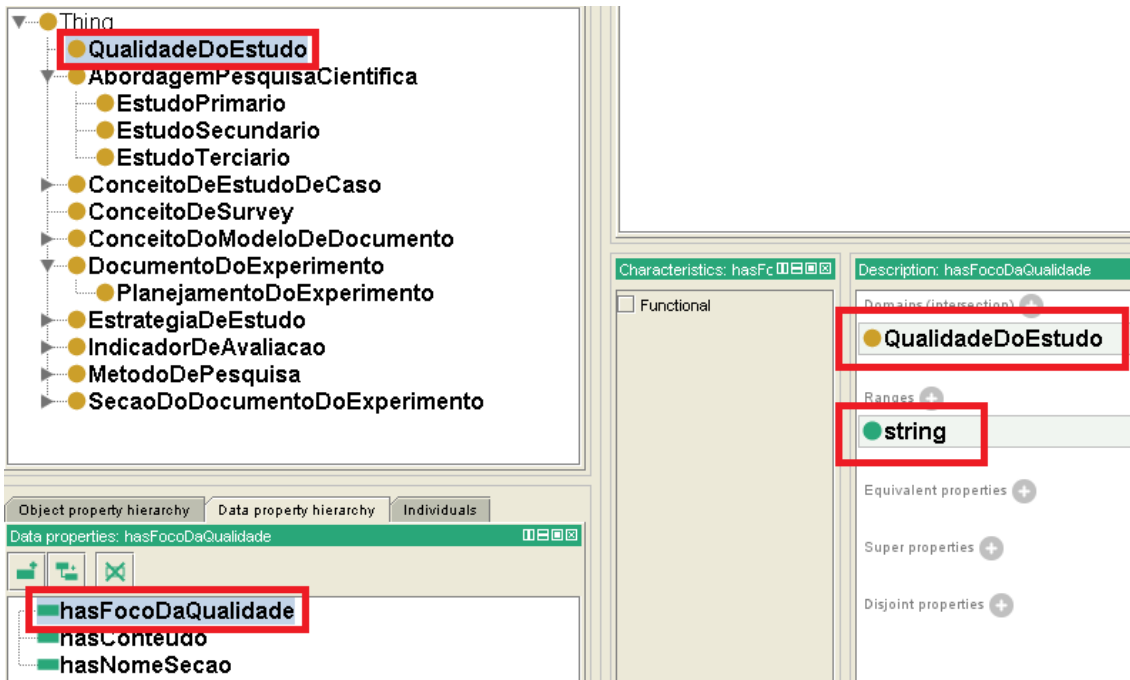


Figura 5-2. Data property “hasFocoDaQualidade” e seu domínio e alcance

Além disso, a classe OWL relacionada ao atributo passa a conter uma superclasse construída utilizando a expressão

$$\langle data\ property \rangle\ SOME\ \langle tipo\ primitivo \rangle$$

Desta forma, tornamos a classe OWL uma subclasse de todas as classes que possuem pelo menos um relacionamento através da *data property* criada. Essa restrição foi criada para ser aderente ao paradigma orientado a objetos, na qual uma classe é descrita obrigatoriamente em termos de seus atributos e métodos. Neste caso, a classe OWL também deve ser descrita obrigatoriamente segunda a *data property* criada em função do atributo. A figura 5-3 ilustra as superclasses da classe OWL “QualidadeDoEstudo”. Sua classe UML correspondente (“Qualidade do Estudo”) possui o atributo “Foco Da Qualidade”, e cuja necessidade de existência se dá em OWL em função da especificação da superclasse “hasFocoDaQualidade some string”, conforme destaca a figura. Isso implica que “toda conceito de qualidade do estudo possui ao menos um foco de qualidade descrito como uma string”.

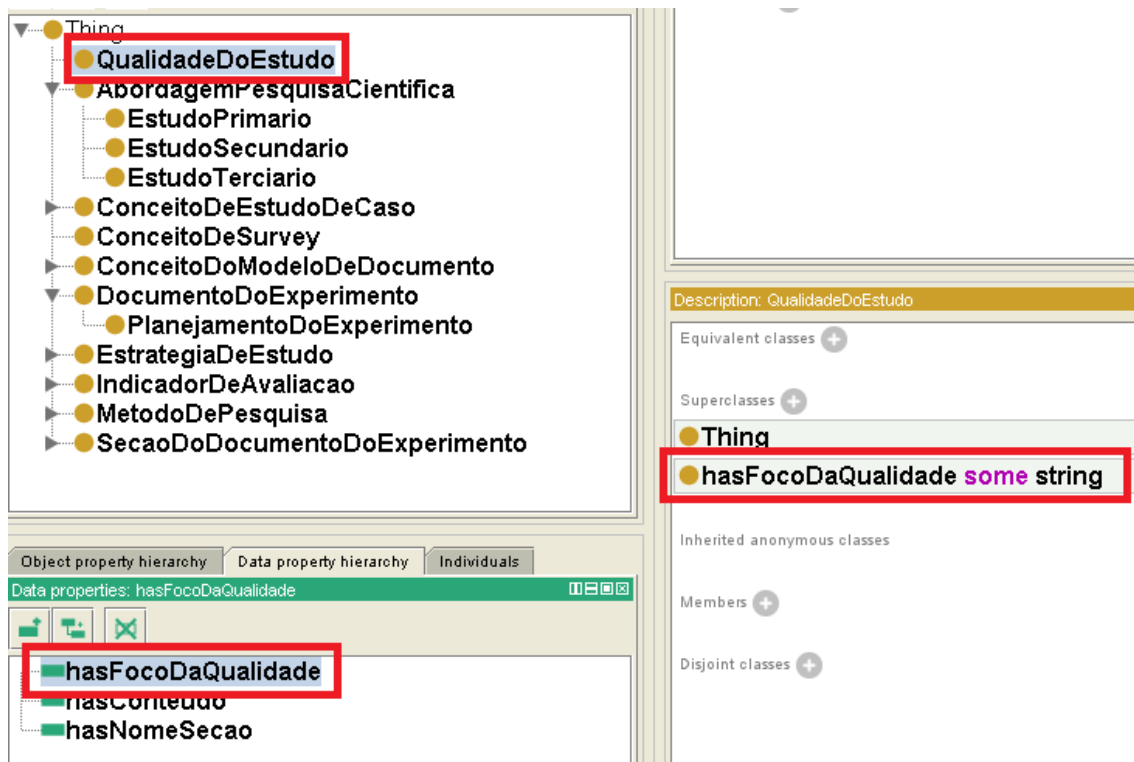


Figura 5-3. Superclasse da classe “QualidadeDoEstudo”

5.5.3. Generalização/Especialização entre classes com estereótipo “conceito”

Os relacionamentos de generalização/especialização entre classes com estereótipos “conceito” foram convertidos em relacionamentos hierárquicos OWL. A figura 5-4 ilustra, à direita, o relacionamento hierárquico entre os conceitos de tipo de abordagem de estudo (estudos primário, secundário e terciário) com a classe OWL “AbordagemDePesquisaCientifica”. Os tipos de abordagem são classes especializadas à esquerda na figura.

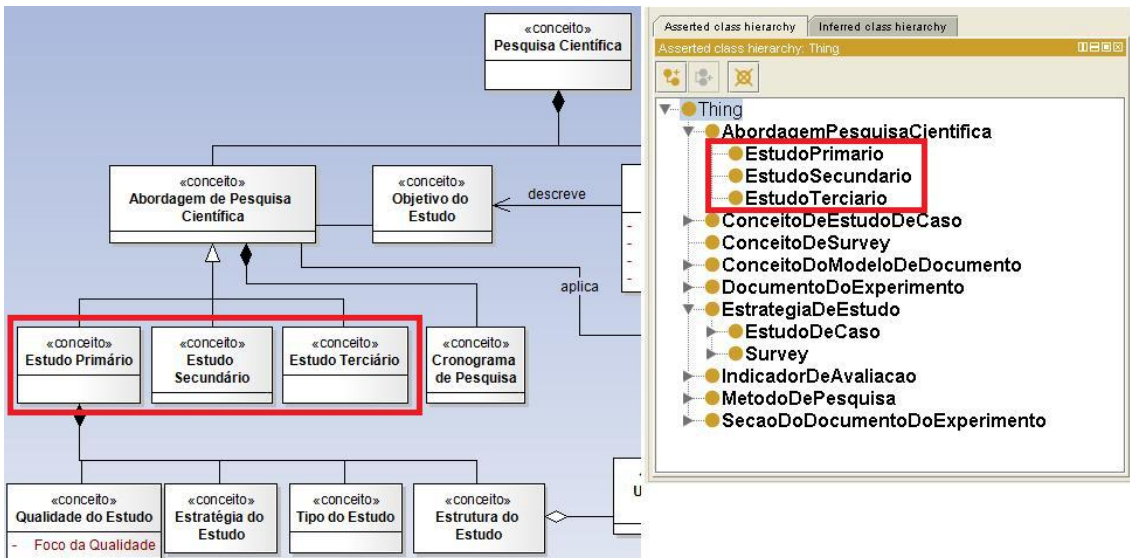


Figura 5-4. Relacionamento hierárquico entre os tipos de abordagem de pesquisa com a classe AbordagemDePesquisaCientifica

5.5.4. Agregação/ Composição entre classes com estereótipo “conceito”

Os relacionamentos de agregação/composição entre classes com estereótipo “conceito” são representados como um object property, cujo nome possui o prefixo “has” e segue com o nome da classe agregada/componente com as iniciais em maiúsculo. A figura 5-5 ilustra o exemplo da classe UML “Estudo Primário” que é composta pela classe “Estratégia do Estudo”. A hierarquia de classes do Protégé, ilustrada à direita na figura destaca a classe OWL de cada classe UML.

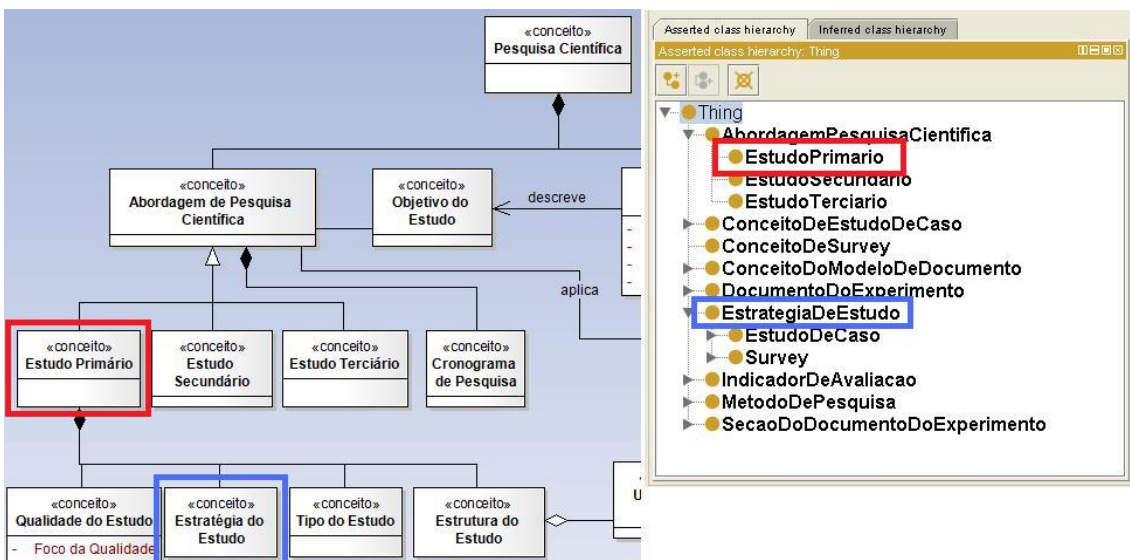


Figura 5-5. Classe “Estudo Primário” que contém a classe “Estratégia do Estudo”. A direita na figura está a hierarquia de classes do Protégé ilustrando as respectivas classes OWL

A classe OWL que contém ou é agregada possui na modelagem em OWL uma superclasse definida pela regra abaixo

$\langle object\ property \rangle\ SOME\ \langle classe\ agregada/componente \rangle$

Essa restrição foi criada para especificar a necessidade de existir ao menos um relacionamento com a classe agregada ou o componente com a classe OWL via a *object property*. A figura 5-6 evidencia no Protégé a regra de superclasse para a classe OWL “EstudoPrimario”, que contém a classe “EstrategiaDoEstudo” através da *object property* “hasEstrategiaDoEstudo”.

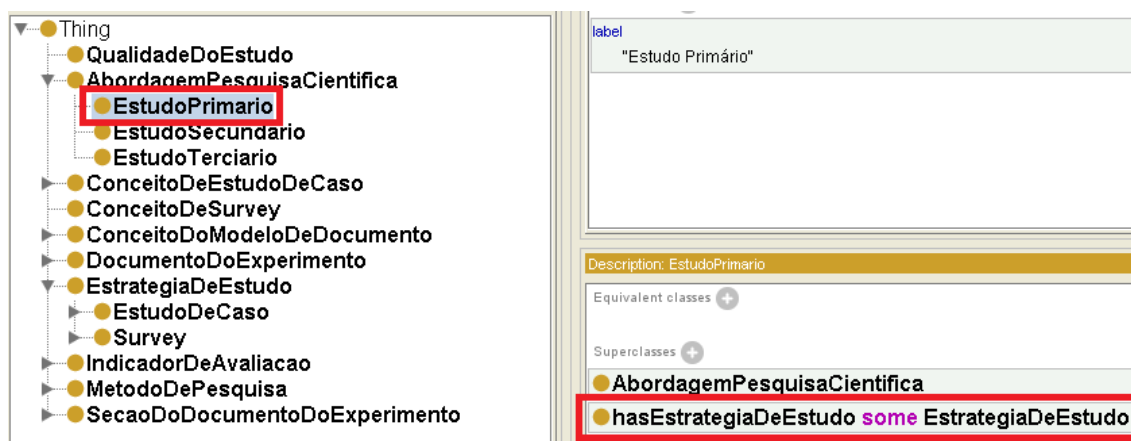


Figura 5-6. Superclasse da classe “EstudoPrimario”

5.5.5. Agregação/Composição entre classes com estereótipo “documento”

Os modelos conceituais (subontologias de documento) que descrevem os documentos do processo de experimentação possuem classes com estereótipo “documento”. Nestes modelos, as classes com estereótipos “documento” podem estar relacionadas por agregação/composição a outras classes de mesmo estereótipo. Neste caso, a semântica atribuída à agregação/composição entre documentos é a de que um documento corresponde a uma seção do outro. Todas as regras de construção dessas subontologias podem ser analisadas com maiores detalhes no ANEXO I. A figura 5-7 exhibe um fragmento da Subontologia de Documentos de Estudo de Caso.

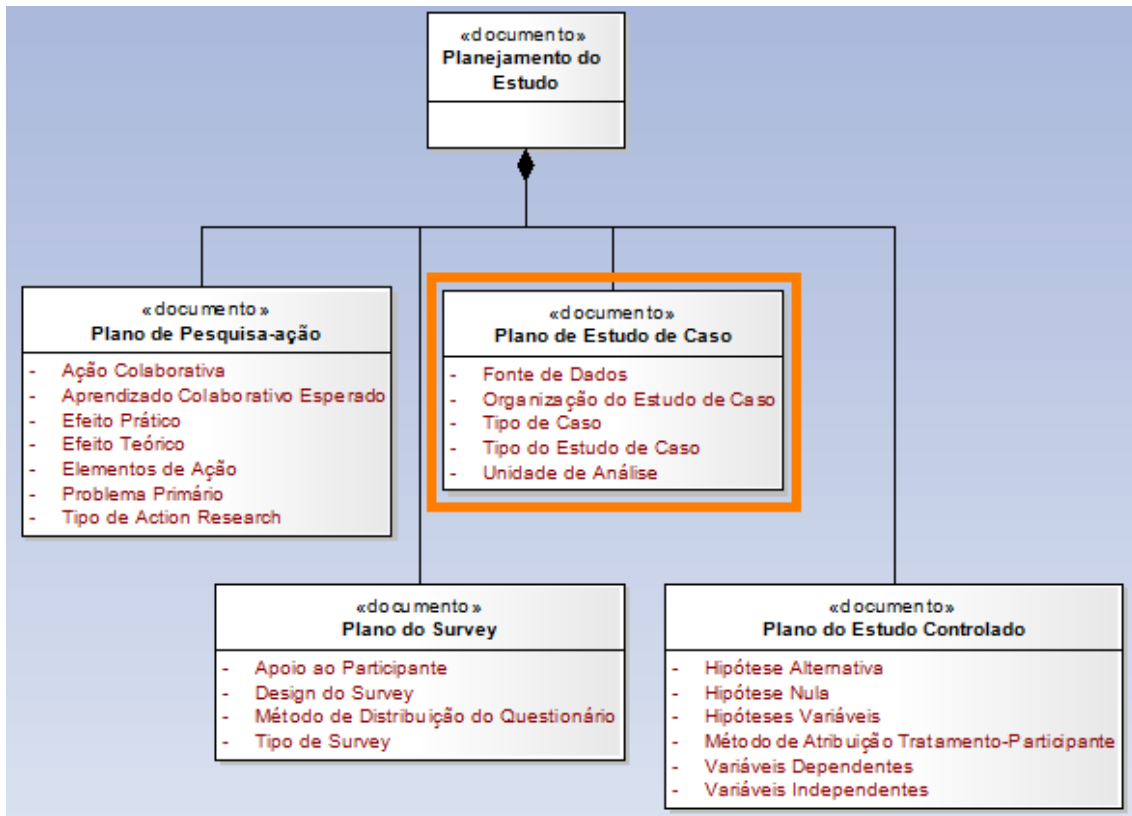


Figura 5-7. Subontologia de Planejamento do Estudo

A transformação desses relacionamentos envolve a criação de um object property com o nome “hasSecao”, no qual o alcance é uma classe OWL chamada “Secao”. Todas as classes OWL agregadas ou componentes possuem como restrição de superclasse a classe “Secao”, implicando que são obrigatoriamente também uma seção de outro documento. Na classe OWL referente à classe-alvo da agregação/composição no modelo UML, é adicionada uma regra na superclasse com a seguinte formação:

hasSecao SOME <classe agregada/componente>

No exemplo da figura 5-7, a classe “Plano de Estudo de Caso” é uma seção do documento “Planejamento do Estudo”. Neste caso, a classe “PlanejamentoDoExperimento” apresenta também como restrição a superclasse “DocumentoDoExperimento”, para especificar que trata-se de um documento conforme explicita o estereótipo “documento” no modelo UML. A classe “PlanoDoEstudoDeCaso” apresenta na definição de sua superclasse a classe “SecaoDoDocumentoDoExperimento” para especificar que trata-se de uma seção. A figura 5-8 ilustra a regra de superclasse explicitada para a classe OWL “PlanejamentoDoEstudo”.

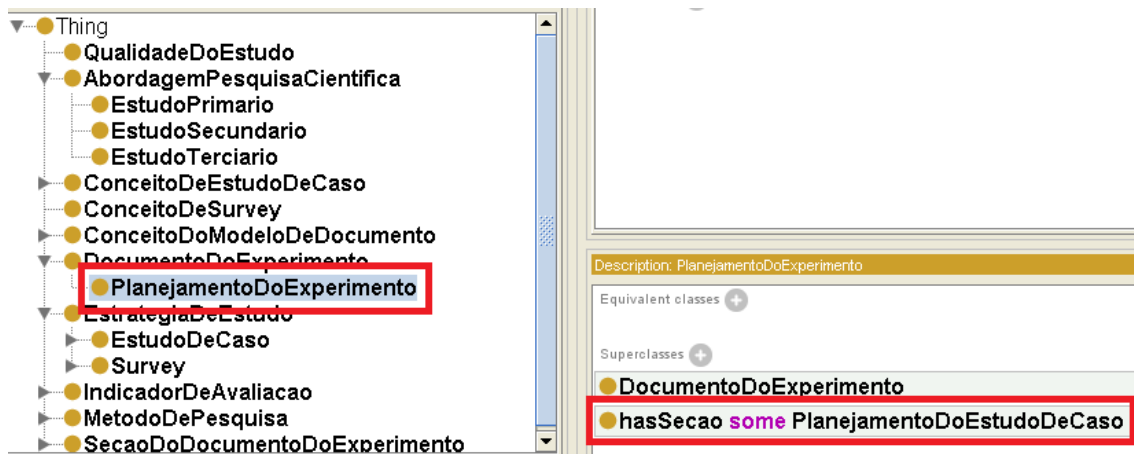


Figura 5-8. Regra de superclasse para a classe OWL "PlanejamentoDoExperimento"

5.5.6. Atributo de classes com estereótipo “documento”

Os atributos das classes das subontologias de documento representam seções de um dado documento. Caso este documento já represente uma seção de outro documento, o atributo da classe representa uma subseção. A figura 5-9 destaca o contorno da classe documento “Plano de Estudo de Caso” que, por já representar uma seção do documento “Planejamento do Estudo”, apresenta com subseção os conceitos “Tipo de Estudo de caso”, “Tipo de Casos para Estudo”, “Organização do Estudo de Caso”, “Unidade de Análise do Estudo de Caso” e “Fonte de Dados do Estudo de Caso”.

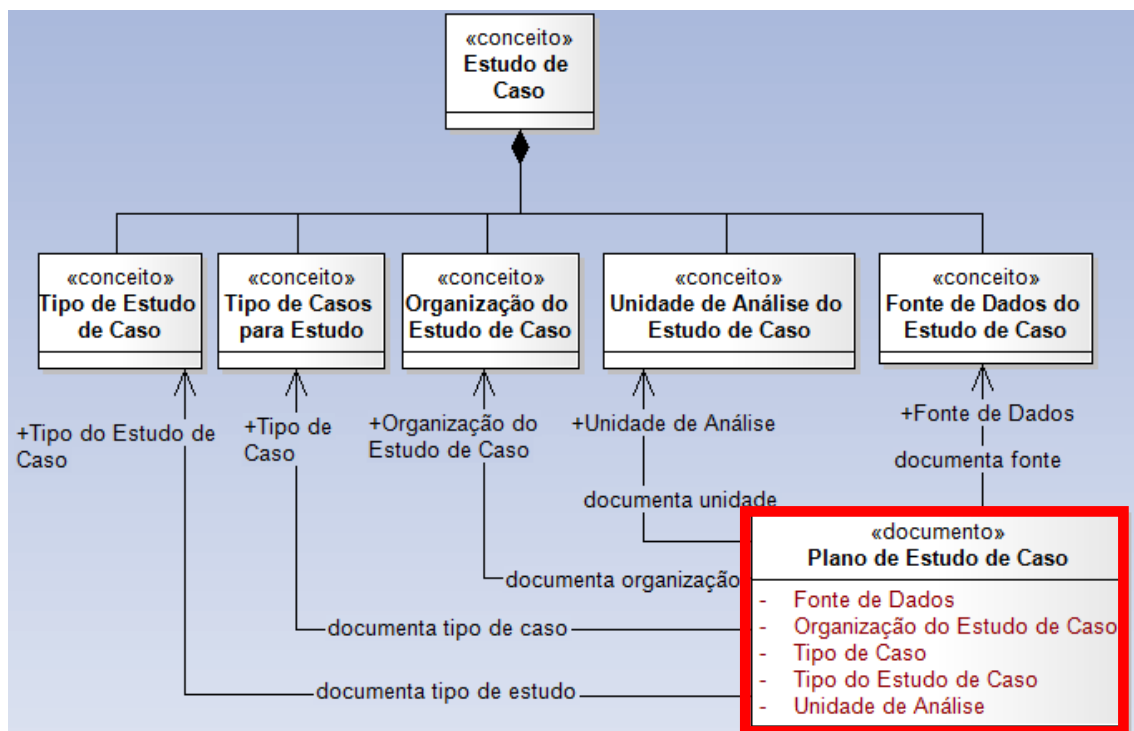


Figura 5-9. Subontologia de Documentos de Estudo de Caso

Os atributos das classes com estereótipo “documento” foram transformados em elementos *object property*, que podem ser “hasSecao” ou “hasSubsecao”. Caso a classe com estereótipo “documento” não esteja contida como seção em outro documento, o atributo da classe será transformado no *object property* “hasSecao”. Caso o documento constitua uma seção dentro de outro documento (exemplo da figura acima – “Plano do Estudo de Caso” é subseção de “Planejamento do Estudo”), os atributos são convertidos no *object property* “hasSubsecao”.

Na descrição OWL das classes de documento, são especificadas como sendo equivalentes (*equivalent classes*) as classes com restrições

<hasSecao/ hasSubsecao > SOME <classe OWL que corresponde à seção/subseção>

O objetivo da regra é estabelecer, assim como prevê a definição de *equivalent class* em OWL, que todos os indivíduos que possuem como seção ou subseção uma determinada classe são também considerados um documento. A figura 5-10 ilustra as regras de *equivalent classes* que indicam que TODO indivíduo que possui as subseções “Tipo de Casos para Estudo”, “Organização do Estudo de Caso” e “Unidade de Análise do Estudo de Caso” é também da classe de “PlanejamentoDoEstudoDeCaso”.

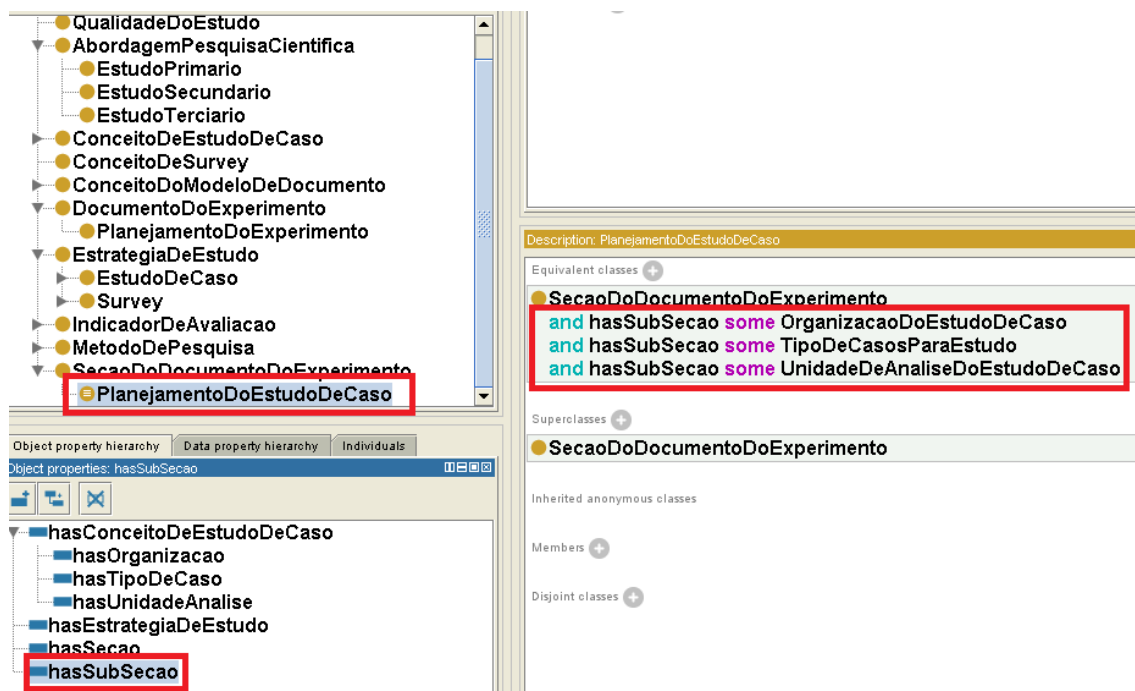


Figura 5-10. Regra de equivalent class para a classe "PlanejamentoDoEstudoDeCaso"

5.5.7. Associação

A heurística transforma ainda relacionamentos associativos em *object properties*. Nesta transformação, a classe UML que representa a origem do

relacionamento associativo é incluída no domínio do *object property*, enquanto que o alcance do *object property* é preenchido pela classe de destino. As classes OWL que representam as classes UML de origem da associação possuem uma restrição na superclasse no formato

<object property> SOME <classe destino da associação>

O nome do *object property* obedece a lei de formação “has” + nome da classe UML sem espaços e com as iniciais de cada palavra em maiúsculo.

5.6. Estratégia para Extração do Conhecimento da Ontologia

O conhecimento a ser extraído dos modelos visa a auxiliar o pesquisador durante as decisões realizadas no contexto da atividade de planejamento. Para disponibilizar esse conhecimento ao pesquisador, foi incluído o metadado “decisao” às classes OWL que possuem subclasses na hierarquia de classes. Neste caso, as classes OWL contempladas com esse metadado correspondem às classes UML com estereótipo “conceito” e que possuem classes especializadas. O metadado “decisão” é preenchido com um texto representando uma pergunta. Foram incluídos em todas as classes OWL os metadados “label” e “comment”, ambos oriundos da linguagem RDF, para explicitar respectivamente o nome completo e a definição da classe. Abaixo segue figura 5-11 que exhibe os metadados incluídos na classe OWL “AbordagemDePesquisaCientifica”.

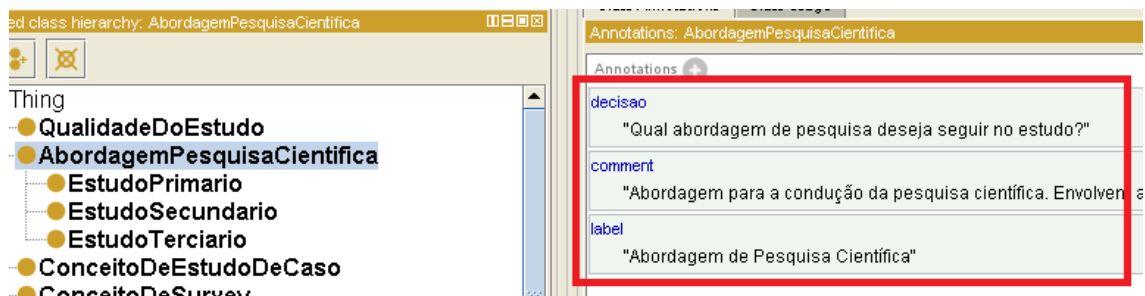


Figura 5-11. Metadados para a classe OWL “AbordagemDePesquisaCientifica”

A estratégia para extrair esse conhecimento dos modelos envolve um algoritmo que percorre pela hierarquia de classes do modelo OWL da Ontologia de Pesquisa Científica. O objetivo principal é que cada visita a uma classe obtenha, a partir dos metadados, qual conceito deve ser disponibilizado ao pesquisador na forma de uma decisão durante o planejamento do estudo. O algoritmo inicia a execução pela classe OWL “PesquisaCientifica”, que é inserida na pilha vazia. Segue abaixo o algoritmo em detalhes.

1. Enquanto a pilha não estiver vazia:
 - 1.1. Caso a classe sendo visitada possua o metadado “decisao”:
 - 1.1.1. Exibir a pergunta especificada no metadado “decisao” ao pesquisador
 - 1.1.2. Fornecer como opções de resposta as subclasses utilizando o metadado “label” de cada subclasse
 - 1.1.3. Remover a classe da pilha
 - 1.2. Obter todas as restrições *object property* da classe corrente. Para cada classe especificada no alcance (*range*) do *object property*:
 - 1.2.1. Incluir na pilha a classe
 - 1.2.2. Caso a classe sendo visitada possua o metadado “decisao”:
 - 1.2.2.1. Exibir a pergunta especificada no metadado “decisao” ao pesquisador
 - 1.2.2.2. Fornecer como opções de resposta as subclasses utilizando o metadado “label” de cada subclasse
 - 1.3. Para cada pergunta, obter a resposta do pesquisador
 - 1.3.1. Incluir na pilha a subclasse correspondente a resposta

O algoritmo acima percorre a hierarquia de classes para extrair da ontologia a decisão associada a cada classe OWL através do metadado “decisao”. A pergunta armazenada no metadado é exibida e as opções de resposta correspondem às subclasses. A navegação continua percorrendo as subclasses e as classes relacionadas através de *object properties*.

Após navegar por toda a Ontologia de Pesquisa Científica, um novo algoritmo é executado para percorrer a Ontologia do Pacote de Experimentação, na qual estão descritos os documentos constituintes do pacote. O objetivo é verificar cada classe desta Ontologia para determinar se alguma classe visitada no algoritmo anterior constitui o alcance (*range*) de uma *object property* “hasSecao” ou “hasSubsecão” de uma classe na Ontologia do Pacote de Experimentação. Isto indicará se uma classe visitada constitui uma seção/subseção dentro do documento cuja classe OWL está sendo visitada pelo algoritmo. No algoritmo é utilizado o mecanismo Fact++, nativo da própria API OWL. Abaixo segue o algoritmo:

1. Enquanto a pilha não estiver vazia:
 - 1.1. Insere na pilha as subclasses da classe corrente
 - 1.2. Caso a classe sendo visitada não possua subclasses e apresente como superclasse a classe “DocumentoDoExperimento” (uso de inferência com o Fact++)
 - 1.2.1. Gera o documento no formato DOC referente à classe corrente
 - 1.3. Caso contrário:
 - 1.3.1. Caso a classe corrente seja uma seção (uso de inferência com o Fact++):
 - 1.3.1.1. Insere no documento sendo construído a seção correspondente
 - 1.3.1.2. Caso a classe classe corrente possua subseções (uso de inferência com o Fact++):
 - 1.3.1.2.1. Insere no documento sendo construído a subseção correspondente

Cada documento é gerado no formato DOC do Microsoft Word contendo as respectivas seções de acordo com as decisões tomadas pelo pesquisador.

5.7. Exemplo de Utilização do Protótipo

Nesta seção descreveremos um exemplo de utilização do protótipo de ferramenta, que auxilia o pesquisador em tomadas de decisão na etapa de planejamento. Este apoio se dá pela navegação nos modelos ontológicos para recuperar quais conceitos estão associados a uma decisão, isto é, qual classe OWL possui um metadado do tipo “decisão” preenchido. A partir desta recuperação, a ferramenta prossegue a execução seguindo os algoritmos supracitados, gerando ao final um template de documento de planejamento com as seções e respectivas informações preenchidas em função das decisões tomadas.

O exemplo considera apenas um subconjunto de tomadas de decisão na etapa de planejamento. Para demonstrar como a ferramenta disponibiliza conhecimento ao pesquisador para auxiliá-lo na tomada de decisões, optamos por ilustrar sua aplicação

no planejamento de um estudo baseado na estratégia de *survey*. Esta escolha fará a ferramenta consultar os seguintes modelos nesta ordem:

- Ontologia de pesquisa científica;
- Subontologia de estratégia de estudo;
- Subontologia de *survey*;

As figuras 5-12 (abaixo), 5-15 e 5-17 destacam os conceitos dos modelos especificados na linguagem LINGO (MIAN, 2003) cuja classe OWL na ontologia implementada possui o metadado “decisão” preenchido. Para cada um desses conceitos, a ferramenta exibirá uma pergunta acompanhada da definição dos principais conceitos envolvidos com a pergunta. Ainda nas figuras seguintes, o número exibido junto ao conceito em destaque indica a ordem em que a decisão será exibida ao pesquisador na seqüência de perguntas.

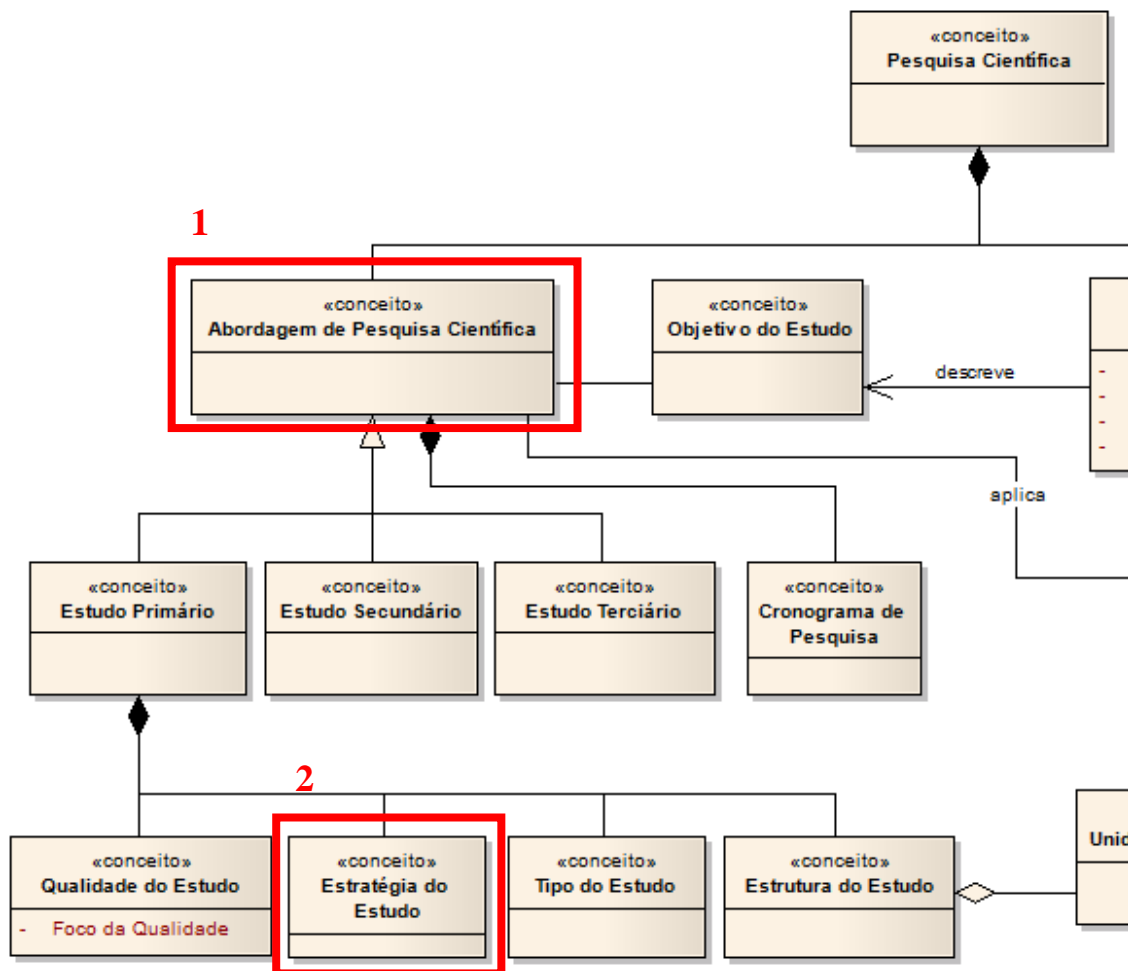


Figura 5-12. Conceitos associados a tomadas de decisão na Ontologia de Pesquisa Científica

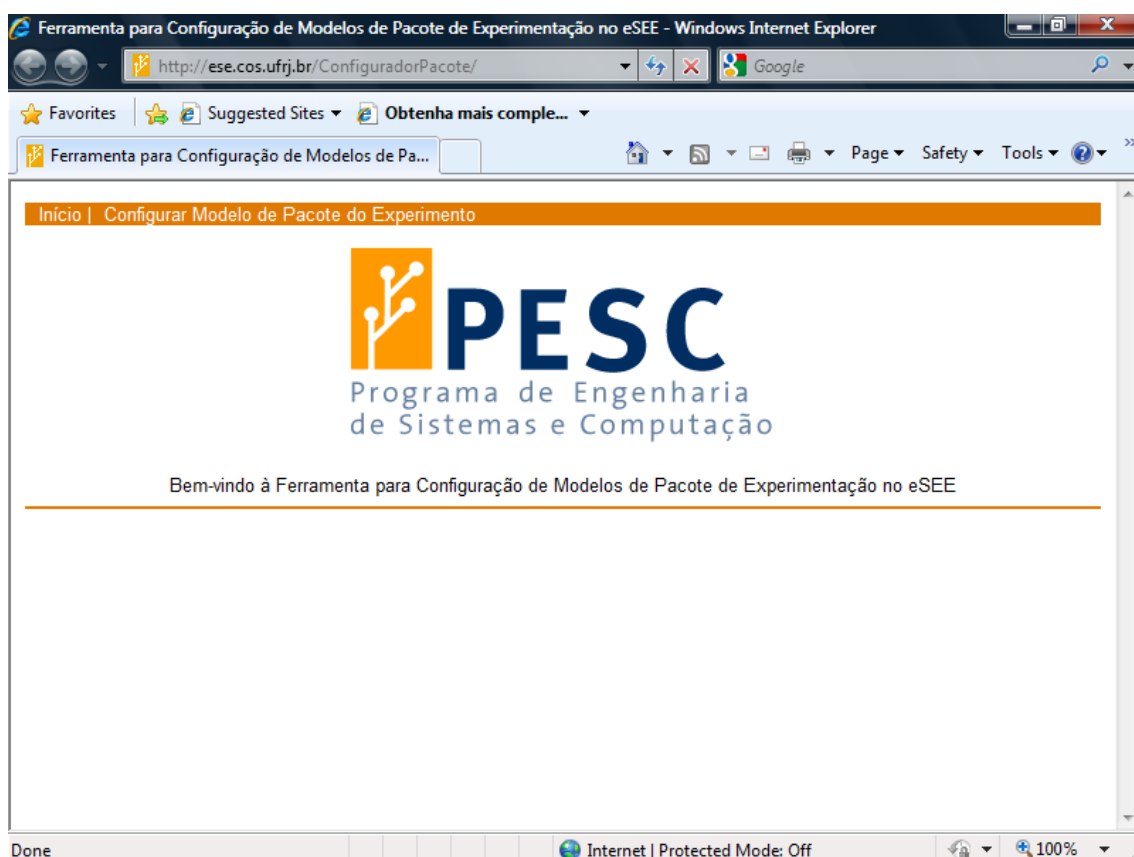


Figura 5-13 - Interface inicial da ferramenta

Na figura 5-12, duas decisões devem ser tomadas pelo pesquisador, que são sobre qual tipo de abordagem de pesquisa científica e estratégia de estudo deve ser adotada para o estudo sendo planejado. A decisão de número 1 está presente na figura 5-14, que ilustra a pergunta (conteúdo do metadado “decisão”) e as opções de resposta (subclasses da classe de abordagem de pesquisa, que são estudo primário, secundário e terciário). A definição do conceito de abordagem de pesquisa científica é recuperada da ontologia e disponibilizada ao pesquisador, de modo que possa tomar sua decisão ciente do que se trata o conceito envolvido.

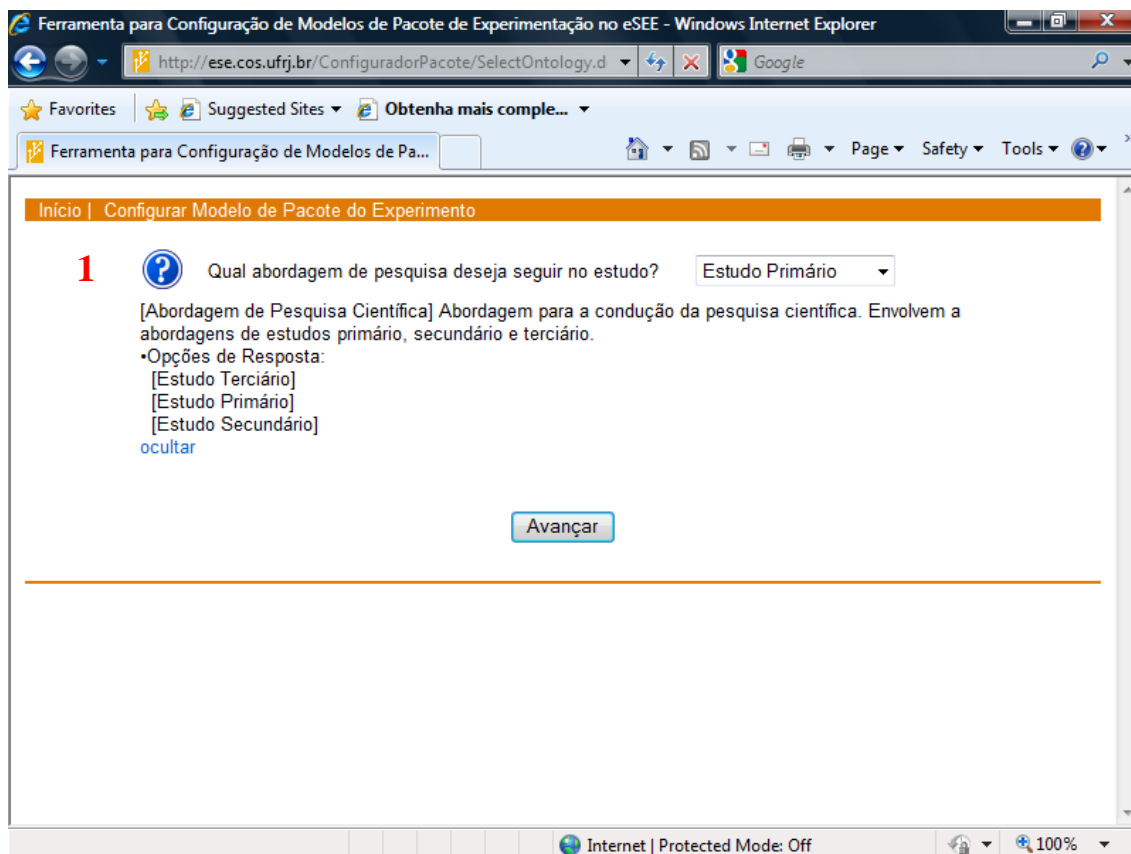


Figura 5-14 - Decisão sobre o tipo de abordagem de pesquisa científica

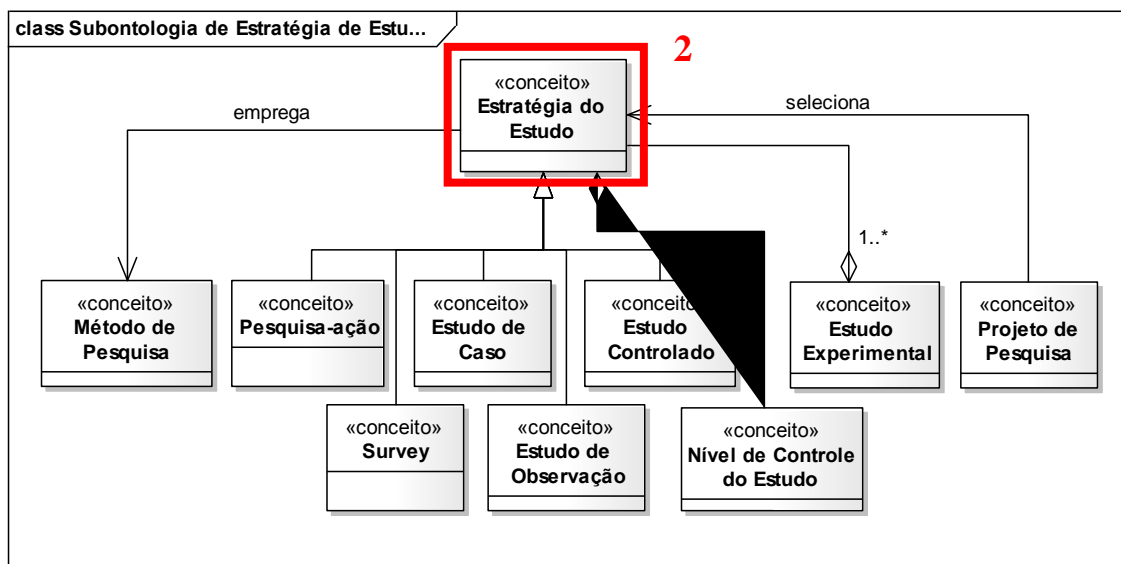


Figura 5-15 - Conceitos associados a tomadas de decisão na Subontologia Estratégia de Estudo

Na figura 5-15, o conceito de estratégia de estudo está associado à decisão sobre qual estratégia deve ser adotada. No exemplo de uso da ferramenta, será selecionada a opção de estratégia de *survey* (Figura 5-16).

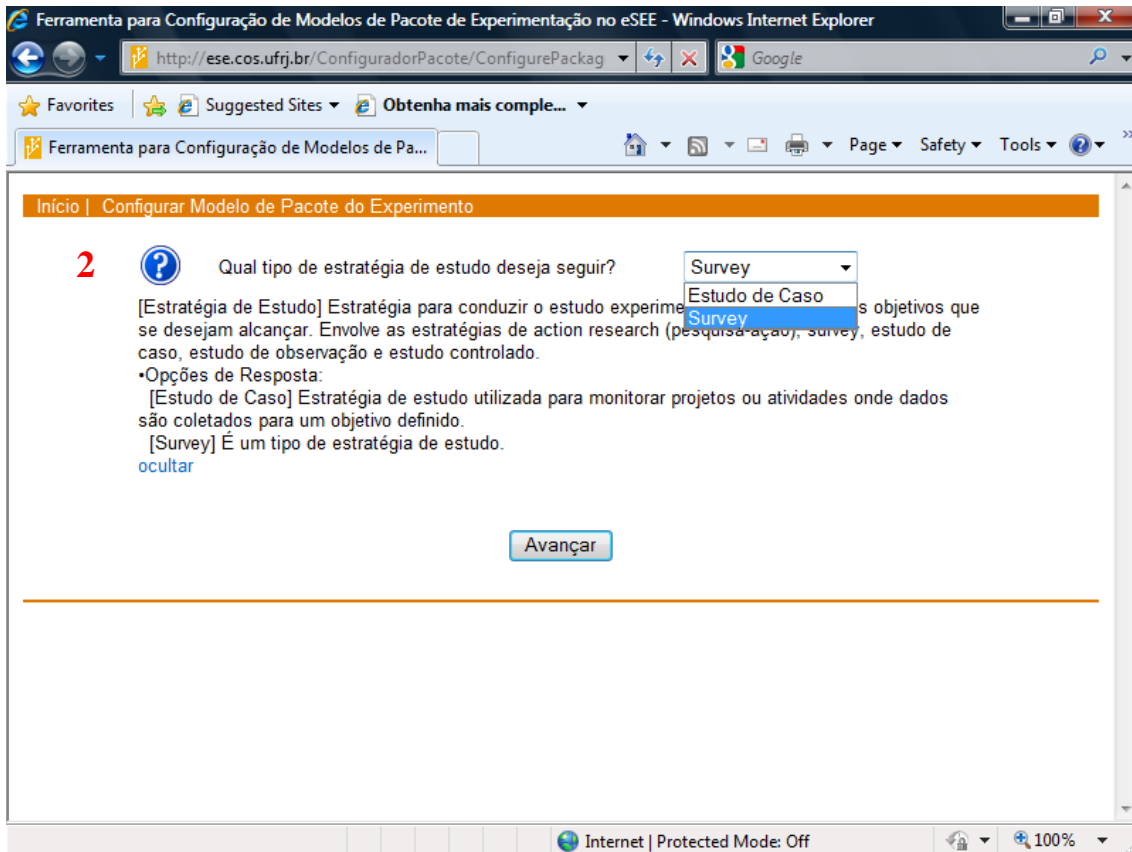


Figura 5-16 - Decisão sobre o tipo de survey

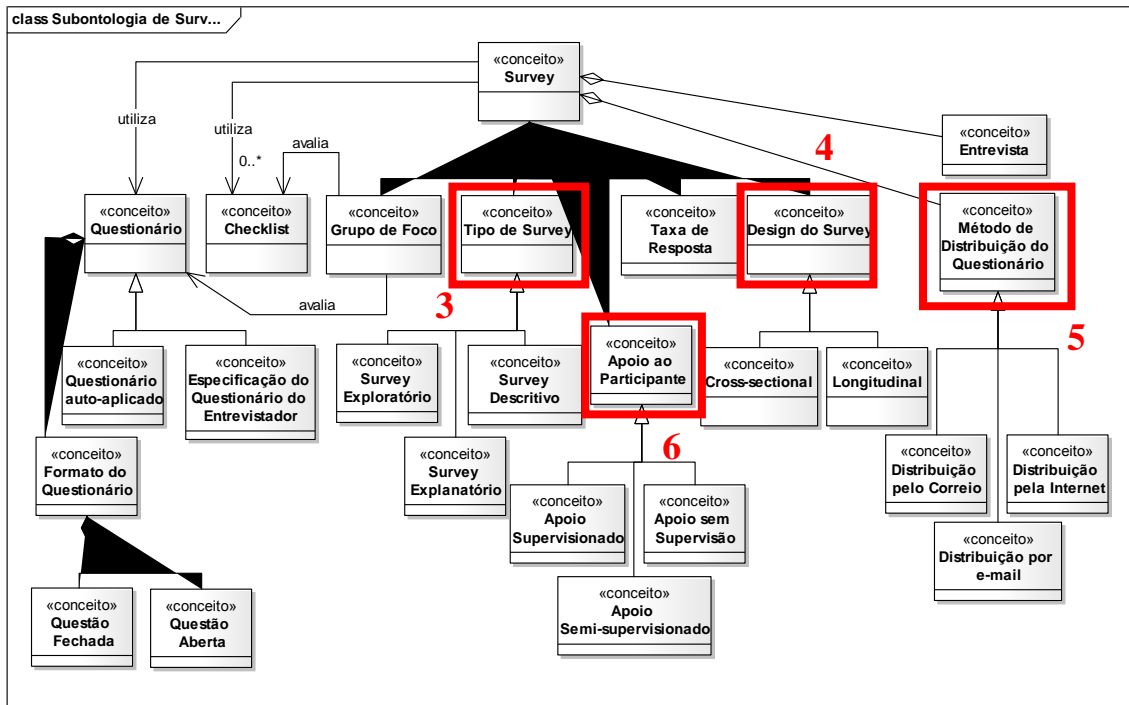


Figura 5-17 - Conceitos associados a tomadas de decisão na Subontologia de Survey

Na subontologia de *survey* (Figura 5-17), demonstraremos a decisão associada ao tipo de *survey*, design e apoio ao participante. Para cada conceito a ferramenta exibe

a decisão associada e as opções de resposta como sendo subclasses do conceito em questão. Todos os conceitos envolvidos possuem sua definição apresentada ao pesquisador, tal como pode ser observado nas figuras 5-18 e 5-19.

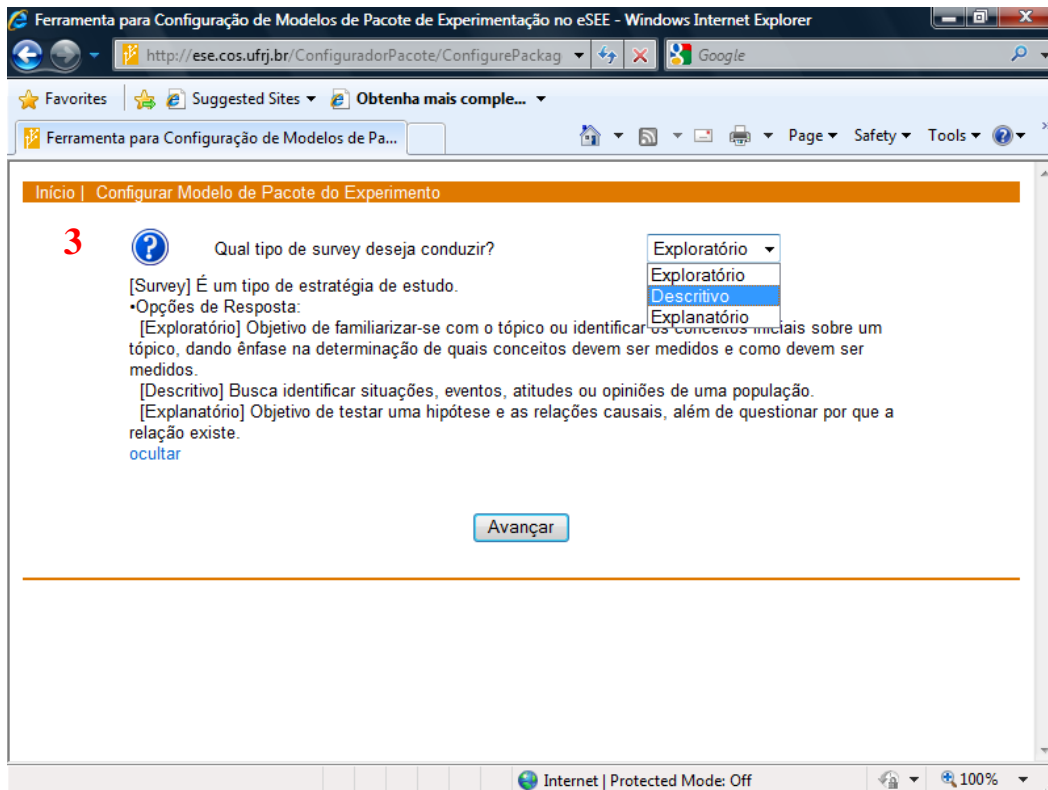


Figura 5-18 - Decisão sobre o tipo de survey

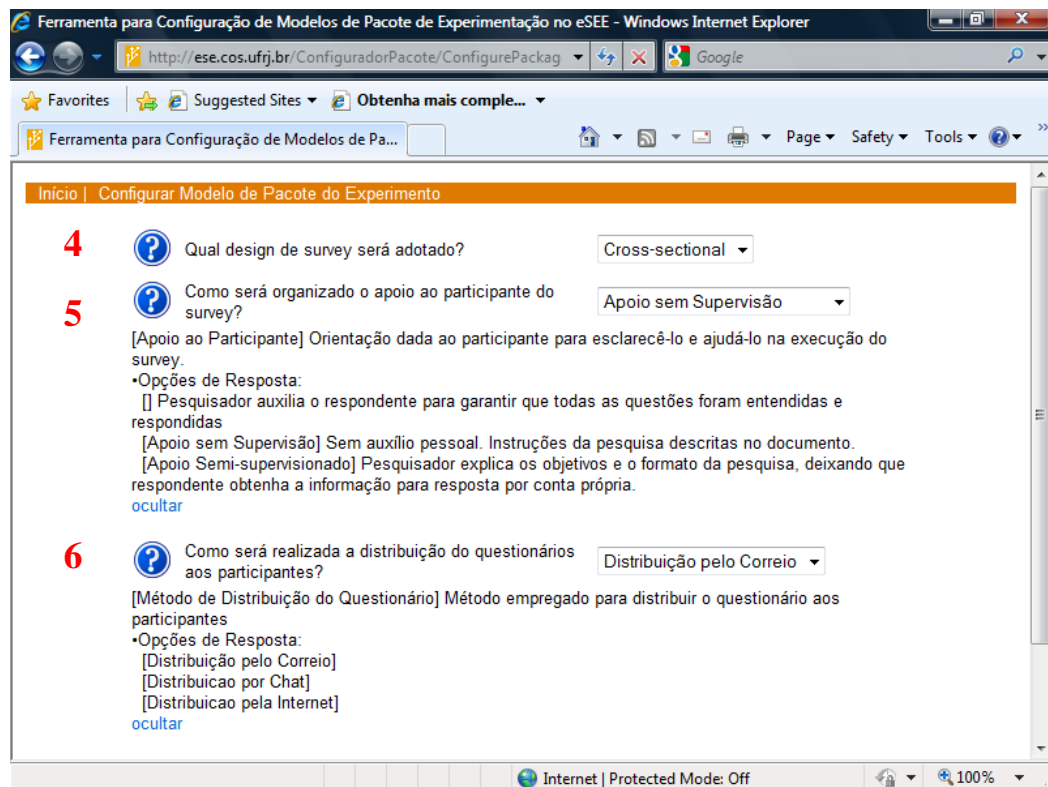


Figura 5-19 - Decisão sobre os tipos de design de survey, apoio ao participante e forma de distribuição de questionário

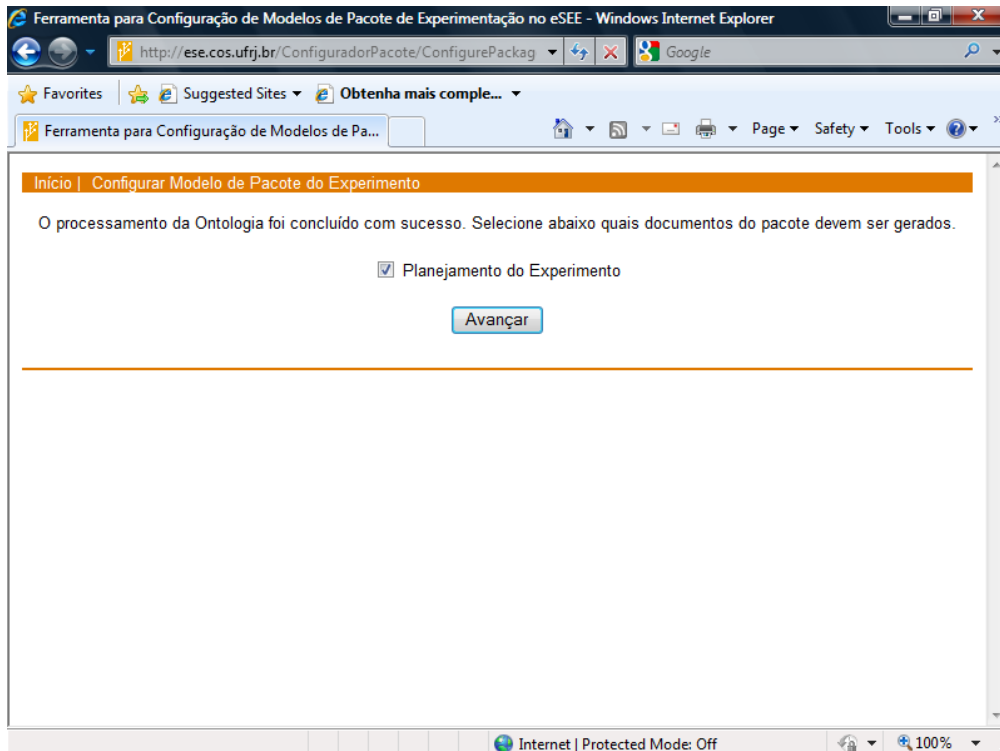


Figura 5-20 - Interface para seleção do documento do pacote de experimento que pode ser gerado pela ferramenta

Uma vez concluída a execução do algoritmo que navega pela ontologia, a ferramenta dá início a recuperação dos documentos que podem ser gerados a partir das respostas às perguntas. Como se trata de um protótipo, optamos por contemplar inicialmente apenas o documento de plano do experimento, uma vez que o próprio conteúdo inicial do repositório está focado no conhecimento relevante para a etapa de planejamento do estudo. Na figura 5-20, o usuário seleciona qual documento deve ser gerado, enquanto que na figura 5-21 é possível fazer o download do documento.

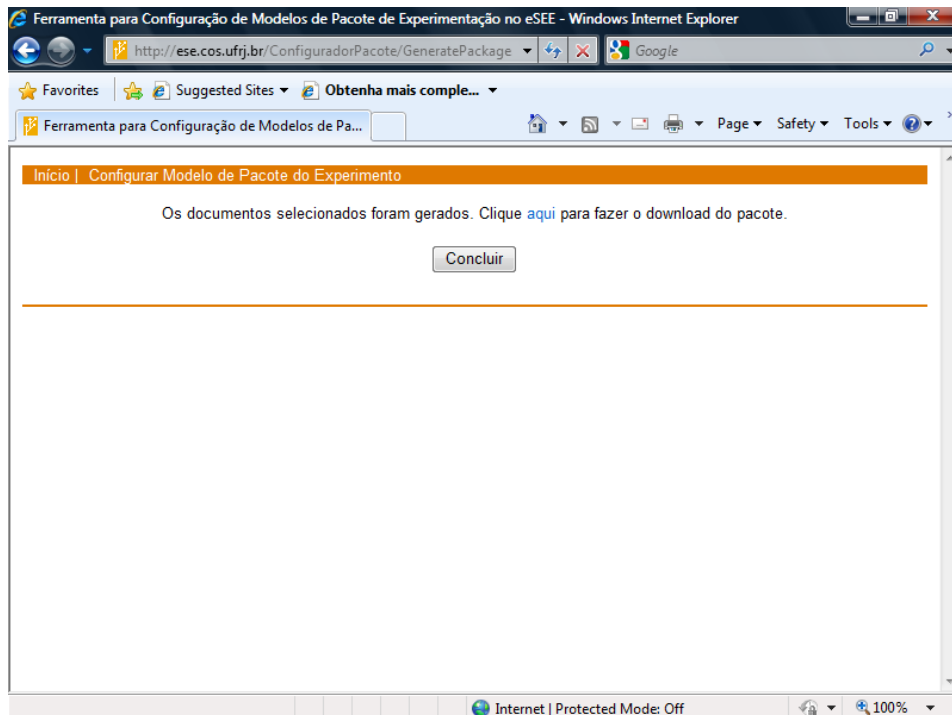


Figura 5-21 - Interface para download do documento selecionado para geração

O documento de plano do experimento é gerado (Figura 5-23) contendo a seção de “planejamento de survey” que, por sua vez, contém como subseções aquelas referenciadas pelas propriedades da classe “Plano do Survey”, conforme ilustra a figura 5-22. Neste caso, as subseções são tipo de survey, design do survey, método de distribuição do questionário e apoio ao participante. As respostas às perguntas são inseridas nas subseções, que podem ser acrescidas de novas informações pelo pesquisador.

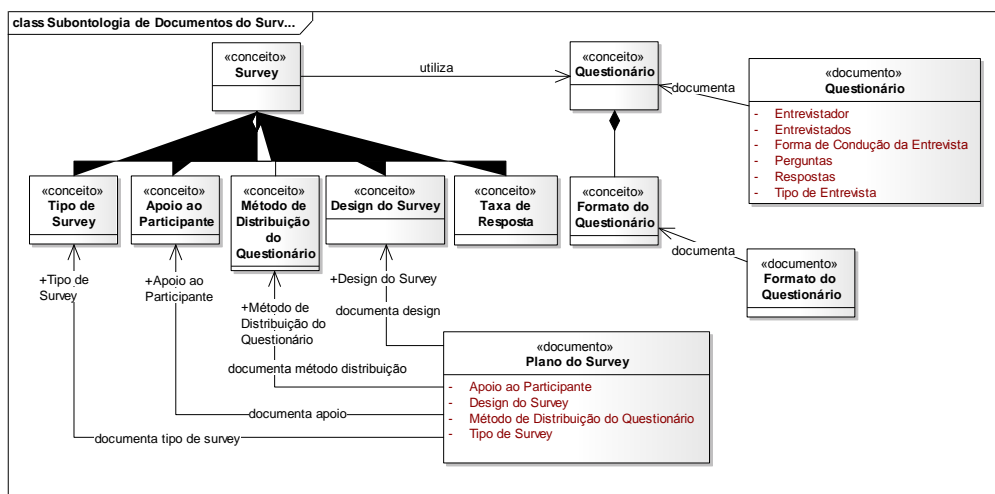


Figura 5-22 - Subontologia de Documento de Survey

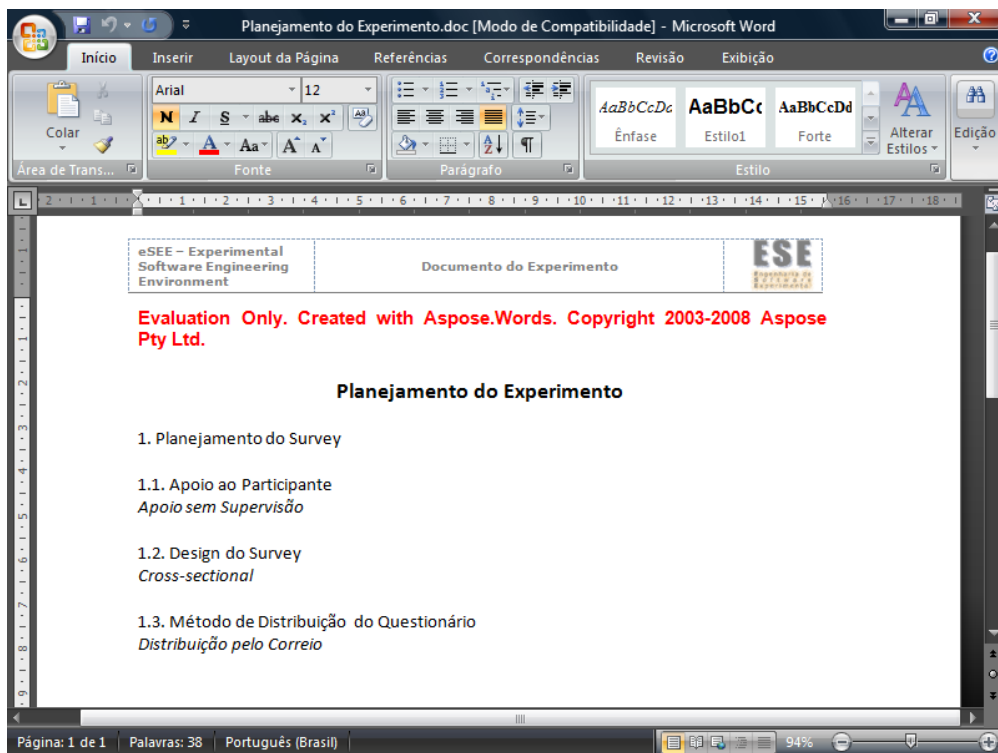


Figura 5-23 - Exemplo de documento de plano de estudo que pode ser gerado a partir da escolha da estratégia de survey

5.8. Conclusões

Este capítulo descreveu um protótipo de ferramenta que demonstra a viabilidade em recuperar conhecimento do repositório proposto e, utilizando-se de metadados associados a classes em modelos OWL, identificar pontos de tomada de decisão e disponibilizá-los ao pesquisador. Para construção da ferramenta, a atividade de implementação do processo de construção da ontologia codificou os modelos escritos na linguagem LINGO utilizando a OWL. Assim, os novos modelos passaram a ser processáveis computacionalmente, viabilizando a utilização de metadados e mecanismos de inferência.

Capítulo 6

Conclusão

Este capítulo apresenta as principais conclusões, contribuições, limitações e perspectivas de trabalhos futuro

6.1. Conclusões

Nesta dissertação foi discutida a complexidade acerca da definição da estrutura e construção de um repositório de conhecimento para um ambiente de apoio à Experimentação em larga-escala em Engenharia de Software. Dentre os objetivos relacionados à construção deste repositório destacam-se:

1. Estender o glossário de termos do eSEE com conhecimento acerca dos subdomínios de estratégia de estudo, método de pesquisa e ambiente de estudo, uma vez que o estágio do glossário anterior a este trabalho não contemplava este corpo de conhecimento;
2. Formalizar conhecimento sobre o domínio de Engenharia de Software Experimental através de ontologias, além de tornar explícitos os conceitos do subdomínio de estratégia de estudo. Isto atende a um dos desafios de SJØBERG *et al.* (2007) no tocante à especificação formal do conhecimento deste subdomínio;
3. Construir uma ferramenta inteligente que recupere conhecimento do repositório para disponibilizá-lo ao pesquisador em etapas importantes do processo de experimentação, atuando assim como uma facilidade provida pelo Ambiente ao mesmo tempo que representa uma prova de conceito para o repositório proposto.

O conteúdo do repositório foi avaliado a partir de inspeções conduzidas na disciplina de Engenharia de Software Experimental da COPPE/UFRJ. Importantes oportunidades de melhoria foram identificadas tanto no glossário, quanto nos modelos ontológicos, contribuindo assim para melhoria da completude e corretude de ambos.

Para demonstrar a viabilidade de uso do repositório como núcleo de recuperação de conhecimento, desenvolvemos ainda uma ferramenta capaz de analisar as ontologias codificadas em linguagem OWL, o que permitiu empregar mecanismos de inferência computacional. Esta facilidade é utilizada no contexto da etapa de planejamento, na qual é disponibilizado ao pesquisador o conhecimento acerca de subdomínios inerentes a esta etapa, tais como abordagem, estratégia e ambiente de estudo e método de pesquisa.

Decisões envolvendo esses subdomínios são apresentadas na forma de perguntas e respostas, juntamente com a definição dos conceitos. Ao final a ferramenta gera um template do documento de plano do estudo contendo as seções e informações aderentes às respostas de cada decisão apresentada.

6.2. Contribuições

As contribuições deste trabalho se relacionam a:

- Formalização do conhecimento sobre Engenharia de Software Experimental em ontologias, tornando explícitos definições e relacionamentos entre importantes conceitos dos subdomínios de abordagem, estratégia e ambiente de estudo e método de pesquisa, conhecimento este que passa a estar disponível no eSEE;
- Construção de ferramenta de configuração de modelos de pacote de Experimentação em Engenharia de Software, que utiliza o repositório como núcleo básico para recuperação do conhecimento formal. Assim, a ferramenta viabiliza a:
 - Flexibilização da elaboração dos documentos do pacote do experimento (plano do experimento) de acordo com a tomada de decisão do pesquisador durante o processo de planejamento do experimento e com base nos questionamentos gerados a partir do conteúdo do repositório;
 - Racionalização sobre o conhecimento formal para apoiar tomadas de decisão pelo pesquisador durante as etapas do processo de experimentação. Para tornar o conhecimento racionalizável, as ontologias foram codificadas na linguagem OWL utilizando a ferramenta *Protégé*. O protótipo então analisa o modelo codificado em OWL e percorre a árvore de relacionamentos entre os conceitos para identificar, através de meta-dados específicos, pontos de decisão e suas respectivas opções de resposta;
- Consolidação de um glossário de termos sobre Engenharia de Software Experimental do eSEE como uma proposta de terminologia consistente e abrangente para a área.

6.3. Limitações

A proposta do repositório de conhecimento do eSEE contempla a dependência entre dois mecanismos de representação de conhecimento: glossário e ontologias. Idealmente, o conjunto de termos do glossário, incluindo a definição e as referências de cada termo, deve refletir o conteúdo dos modelos ontológicos. Uma limitação frente a este cenário é que não foi objeto de estudo neste trabalho um mecanismo que garanta a sincronia entre o conteúdo de ambos. Ou seja, ao incluir, alterar ou excluir termos, a modificação correspondente deve ser realizada nos modelos ontológicos corretos. A recíproca também é verdadeira para a ontologia, na qual inserções, alterações e exclusões de conceitos demandam também ajuste manual no glossário. Assim, caso o ajuste no glossário/ontologias não seja realizado, o conteúdo de ambos pode não se manter sincronizado.

As limitações deste trabalho em relação ao protótipo da ferramenta estão principalmente relacionadas às questões de usabilidade e interface gráfica, que estavam fora do escopo deste trabalho. Além disso, a ferramenta, embora idealmente construída para configurar modelos de documentos do pacote de experimentação, apenas o faz para o documento de plano do experimento em razões de escopo.

Ainda em relação a ferramenta, pode ser apontada uma limitação sobre o fato de não ter havido tempo hábil de planejar e executar estudos para avaliar o emprego da ferramenta em um contexto real de estudo experimental. Entretanto, esta ferramenta foi apresentada e discutida entre os pesquisadores da Equipe de Engenharia de Software Experimental da COPPE/UFRJ e está aderente ao procedimento empregado na etapa de planejamento por este grupo de pesquisadores.

Outra limitação diz respeito à forma de uso da ferramenta proposta. Apenas modelos ontológicos em OWL e, sobretudo, construídos seguindo as heurísticas estabelecidas para conversão de UML para OWL podem ser utilizados. Modelos OWL que não seguem o padrão de representação dos meta-dados de decisão e de relacionamentos entre conceitos e documentos não são analisados corretamente pela ferramenta, o que pode gerar resultados desconhecidos.

6.4. Perspectivas Futuras

As ontologias foram construídas com a intenção de explicitar conhecimento dos subdomínios identificados por LOPES e TRAVASSOS (2008) como sendo relevantes

na etapa de planejamento do processo de experimentação. Entretanto, o domínio de Engenharia de Software Experimental não se restringe somente a esses subdomínios e, para melhorar a completude da ontologia e do glossário como um todo, outros subdomínios devem ser pesquisados para inclusão de novos termos e relacionamentos. Como consequência, novas rodadas de inspeção serão necessárias.

A ferramenta proposta pode ser evoluída no sentido de gerar o template dos demais documentos do pacote de experimento de AMARAL (2003). Para isto, o modelo OWL deve ser expandido e novos pontos de decisão devem ser identificados e marcados como meta-dados no ambiente *Protégé*.

A partir do repositório de conhecimento formal em Engenharia de Software Experimental, acreditamos que o eSEE poderá ampliar sua capacidade de apoiar Experimentação larga-escala em Engenharia de Software, uma vez que o processo de experimentação gera e consome grande volume de conhecimento (SHULL *et al.*, 2001) e ter um repositório para armazená-lo e viabilizar sua recuperação torna-se um requisitos fundamental. Assim, com a estruturação deste repositório, poderemos, no futuro, explorar a incorporação de atividades de Gestão do Conhecimento no processo de experimentação, visando melhorar a inclusão, alteração, exclusão e registro de proveniência do conhecimento consumido e produzido durante execução deste processo.

Referências Bibliográficas

- ABCKER, A., BERNARDI, A., HINKELMANN, K., KÜHN, O., SINTEK, M., 1998, “Toward a Technology for Organizational Memories”, *IEEE Intelligent Systems*, v. 13, n. 3 (May/Jun), pp. 40-48.
- ALAVI, M., LEIDNER, D., 1999, “Knowledge Management Systems: emerging Views and Practices from the Field”, In: *Proceedings of the 32nd Hawaii International Conference on Systems Sciences*, pp. 106-113, Maui, Hawaii, Janeiro.
- AMARAL, E.A.G.G., 2003, *Empacotamento de Experimentos em Engenharia de Software*, Dissertação de Mestrado, Programa de Engenharia de Sistemas e Computação, COPPE/UFRJ, Universidade Federal do Rio de Janeiro. Rio de Janeiro, RJ, Brasil.
- BASILI, V. R., CRUZES, D., CARVER, J. C., HOCHSTEIN, L. M., HOLLINGSWORTH, J. K., ZELKOWITZ, M. V., SHULL, F., 2001, “Understanding the High-Performance-Computing Community: A Software Engineer’s Perspective”, *IEEE Software*, v. 21, n. 3 (Agosto), pp. 23-31.
- BASILI, V.R., SHULL, F., LANUBILE, F., 1999, “Building Knowledge through Families of Experiments”, *IEEE Transactions of Software Engineering*, vol. 25, No. 4, Julho/Agosto de 1999, pp.456-473.
- BASKEVILLE, R. L., 1999, “Investigating Information Systems with Action Research”, *Communications of the Association for Information Systems*, Volume 2, Artigo 19, Outubro, pp.120-135.
- BECHHOFFER, S., HORROCKS, I., GOBLE, C., STEVENS, R., 2001, “OilEd: a Reasonable Ontology Editor for the Semantic Web”, In: *Proceedings of KI2001, Joint German/Austrian conference on Artificial Intelligence*, pp. 23-31, Vienna, Setembro.
- BREITMAN, K. K., (2005), “Web Semântica: A Internet do Futuro”, LTC, ISBN – 85-216-1466-7.
- BIOLCHINI, J.C.A., MIAN, P.G., NATALI, A.C.C, CONTE, T.U, TRAVASSOS, G.H., 2006, “Scientific Research Ontology to Support Systematic Review in Software Engineering”, *Advanced Engineering Informatics*, Novembro, v. 21, n. 4, pp.133-151.

- BIOLCHINI, J., MIAN, P.G., NATALI, A.C.C., TRAVASSOS, G.H., 2005, "Systematic Review in Software Engineering", Relatório Técnico, Programa de Engenharia de Sistemas e Computação - PESC, COPPE-UFRJ, ES-679/05.
- CAMPBELL, D.T., COOK, T.D. (eds.), 1979, **Quasi-Experimentation, Design and Analysis Issues for Field Settings**, 1 ed., Boston, USA, Houghton Mifflin Company.
- CHAPETTA, W.A., CONTE, T.U., TRAVASSOS, G.H., 2004, *Requisitos para um Ambiente de Experimentação em Engenharia de Software*, Relatório Técnico, Programa de Engenharia de Sistemas e Computação - PESC/COPPE/UFRJ, da Equipe de Engenharia de Software Experimental.
- CHAPETTA, W.A., 2006, *Uma Infra-estrutura para Planejamento, Execução e Empacotamento de Estudos Experimentais em Engenharia de Software*, Dissertação de Mestrado, Programa de Engenharia de Sistemas e Computação, COPPE/UFRJ, Universidade Federal do Rio de Janeiro. Rio de Janeiro, RJ, Brasil.
- COSTA, H.R., MIAN, P.G., TRAVASSOS, G.H., 2004, *Estendendo um Modelo de Processo e de Empacotamento de Estudos Experimentais*, Programa de Engenharia de Sistemas e Computação - PESC/COPPE/UFRJ, da Equipe de Engenharia de Software Experimental.
- DALY, J., EMAN, K.E., MILLER, J., 1997, "Multi-method research in Software Engineering", In: *IEEE Workshop on Empirical Studies Software Maintenance*, pp. 121-131, Bari, Itália, Outubro.
- DAVENPORT, T.H., PRUSAK, L. (eds.), 1998, ***Working Knowledge: How Organizations Manage What They Know***, Boston, USA, Harvard Business School Press.
- DIAS NETO, A.C., BARCELOS, R.A., C.W., SANTOS, P.S.M., MAFRA, S.N., TRAVASSOS, G.H., 2004, "Infrastructure for SE Experiments Definition and Planning". In: *1st Experimental Software Engineering Latin American Workshop - ESELAW'04*, pp. 34-44, Brasília, Brasil.
- DOMINGUE, J., MOTTA E., CORCHO O., *Knowledge modelling in webonto and ocml: A user guide*, 1999. Disponível em: <http://projects.kmi.open.ac.uk/webonto/user_guide.2.4.pdf>. Acesso em: 24 de novembro de 2009, 12:34.

- EASTERBROOK, S.; SINGER, J.; STOREY, M.; DAMIAN, D., 2008, "Selecting Empirical Methods for Software Engineering Research", In: SHULL, F., SINGER, J., SJØBERG, D. I. K. (eds.), 2008, *Guide to Empirical Software Engineering*, pp. 285-310, London, Springer.
- FALBO, R., MENEZES, C., ROCHA, A.R., (1998), "Using Ontologies to Improve Knowledge Integration in Software Engineering Environments", In: *Proceedings of the International Conference on Information Systems Analysis and Synthesis*, pp. 1-8, Berlin, Germany, October.
- FERNANDEZ, M., GOMEZ-PEREZ A., JURISTO, N., 1997, "Methontology: From ontological art towards ontological engineering", In: *Proceedings of the AAAI97 Spring Symposium Series on Artificial Intelligence*, pp. 33-40, Stanford, USA, March.
- GRUBER, T. R., 1993, "A Translation Approach to Portable Ontology Specifications", *Knowledge Acquisition*, pp. 199-220, ed. 1, Boston, USA, Springer.
- HOVE, S.E., ANDA, B., 2005, "Experiences from Conducting Semi-Structured Interviews in Empirical Software Engineering Research". In: *11th IEEE International Software Metrics Symposium*, pp. 23-33, Stanford, USA, September.
- HORRIDGE M., DRUMMOND N., GOODWIN J., RECTOR A., WANG H., 2006, "The manchester owl syntax", In: *Proceedings of the 2006 OWL Experiences and Directions Workshop (OWL-ED2006)*, pp. 34-44, Athens, Georgia, USA, November.
- HORROCKS I., SATTLER U., TOBIES S., (2000), "Practical reasoning for very expressive description logics", In: *Logic Journal of IGPL*, v. 21, n.10 (August), pp.120-132.
- ISERN, "ISERN basic terminology in Experimental Software Engineering", 2005, Disponível em: <<http://www.cs.umd.edu/projects/SoftEng/tame/isern/isern.definitions.html>>. Acesso em: 26 maio. 2010, 17:00:00.
- JURISTO, N., MORENO, A., 2001, *Basics of Software Engineering Experimentation*, Boston, USA, Kluwer Academic Press.
- KIM, W., GRAUPNER, S., SAHAI, A., LENKOV, D., CHUDASAMA, C., WHEDBEE, S., LUO, Y., DESAI, B., MULLINGS, H., WONG., P., 2002,

- "Web E-speak: facilitating Web-based e-Services", *IEEE Multimedia*, v. 9, n. 1 (Jan-Mar), pp. 43-55.
- KITCHENHAM, B.A., PFLEEGER, S.L., PICKARD, L.M., JONES, P.W., HOAGLIN, D.C., EMAM, K.E., ROSENBERG, J., 2002, "Preliminary Guidelines for Empirical Research in Software Engineering", *IEEE Transactions on Software Engineering*, v. 28, n. 8, pp. 721-734.
- KITCHENHAM, B., 2004, "Procedures for Performing Systematic Reviews", Technical Report, Keele University, Sydney, Australia.
- KITCHENHAM, B., PFLEEGER, S. L., 2008, "Personal Opinion Surveys", In: SHULL, F., SINGER, J., SJØBERG, D. I. K. (eds.), 2008, *Guide to Empirical Software Engineering*, pp. 63-92, London, Springer.
- LOPES, V.P., TRAVASSOS, G.H., 2008, "Infra-estrutura Conceitual para Ambientes de Experimentação em Engenharia de Software", Em: *Anais do Workshop Latino-americano em Engenharia de Software Experimental (ESELAW'08)*, pp. 33-44, Salvador, BH, Brasil, Outubro.
- LOPES, V.P., TRAVASSOS, G.H., 2009, "Experimentação em Engenharia de Software: Glossário de Termos", Em: *Anais do Workshop Latino-americano em Engenharia de Software Experimental (ESELAW'09)*, pp. 33-44, São Carlos, SP, Brasil, Novembro.
- MAEDCHE A., STAAB S., 2000, "The text-to-onto ontology learning environment", In: *Proceedings of the International Conference on Computational Science*, pp. 106-113, Maui, Hawaii, Janeiro.
- MCGUINNESS, D., FIKES, R., RICE, J., WILDER, S., 2002, "An Environment for Merging and Testing Large Ontologies", In: *Proceedings of the Seventh International Conference on Principles of Knowledge Representation and Reasoning (KR-2000)*, pp. 483-493, San Fransisco, USA, April.
- MEDIAWIKI, Mediawiki, 2010, Disponível em: <<http://www.mediawiki.org/wiki/MediaWiki>>. Acesso em: 26 maio. 2010, 18:00:00.
- MELIN, U., AXELSSON, K., 2007, "Action in Action Research – Illustrations of What, Who, Why, Where, and When from an E-Government Project", In: *Lecture Notes in Computer Science*, pp. 101-134, v. 4656. Berlin, Germany, Springer.

- MENDONÇA, C.C., 2004, *Uma Infra-estrutura para apoio ao Planejamento e Execução de Pesquisas de Opinião na Web*, Tese de Mestrado, Programa de Engenharia de Sistemas e Computação -PESC/COPPE, Universidade Federal do Rio de Janeiro -UFRJ, Rio de Janeiro, RJ, Brasil.
- MIAN, P. G., 2003, *ODEd: Uma Ferramenta de Apoio ao Desenvolvimento de ontologias em um Ambiente de Desenvolvimento de Software*, Dissertação de M.Sc., Departamento de Informática, UFES, Vitória, Brasil.
- MONTGOMERY, D.C., 2001, *Design and Analysis of Experiments*, ed. 1, Boston, USA, John Wiley & Sons.
- MONTONI, M. A., 2003, *Aquisição de Conhecimento: Uma aplicação no Processo de Desenvolvimento de Software*, Dissertação de Mestrado, Programa de Engenharia de Sistemas e Computação, COPPE/UFRJ, Universidade Federal do Rio de Janeiro. Rio de Janeiro, RJ, Brasil.
- MÜLLER, M., PFAHL, D., 2008, "Simulation Methods", In: *Guide to Empirical Software Engineering*, pp. 117-152, London, Springer.
- NONAKA, I., TAKEUCHI, H., 1995, *The Knowledge-Creating Company: How Japanese Companies Create the Dynamics of Innovation*, New York, USA, Oxford University Press.
- OGBORN, J., MILLER, R., 1994, "Computational Issues in Modelling". In: MELLAR, H.; BLISS, J.; BOOHAN, R.; OGBORN, J. & TOMPSETT, C. (eds.) (1994), *Learning With Artificial Worlds: Computer Based Modeling in the Curriculum*, pp.117-27, London, The Falmer Press.
- O'LEARY, D.E., 1998, "Using AI in Knowledge Management: Knowledge Bases and Ontologies", *IEEE Intelligent Systems*, v. 13, n. 3 (May/Jun), pp. 33-39.
- PERRY, D.E., SIM, S. E., EASTERBROOK, S., 2005, "Case Studies for Software Engineers", In: *NASA SW Engineering Workshop Tutorial*, pp. 10-15, Florida, USA, November.
- PERRY, D.E.; PORTER, A.A., VOTTA, L.G., 2000, "Empirical Studies of Software Engineering: A Roadmap. The Future of Software Engineering", In: *22nd International Conference on Software Engineering, Limerick*, pp. 85-109, Ireland, April.
- PFLEEGER, S.L., 1999, "Albert Einstein and Empirical Software Engineering". *IEEE Computer*, v. 32, n. 10 (Outubro), pp. 32-38.

- PFLEEGER, S.L., 2001, “Tutorial: Um curso rápido sobre estudos experimentais em Engenharia de Software”. In: *Anais do XV Simpósio Brasileiro de Engenharia de Software*, pp. 109-145, Rio de Janeiro, RJ, Brasil, Outubro.
- PROBST, G.J.B., RAUB, S., ROMHARDT, K. (eds), 2000, ***Managing Knowledge: Building Blocks for Success***, 1 ed., Boston, USA, Wiley.
- PROTÉGÉ, Protégé, 2010, Disponível em: <<http://Protégé.stanford.edu/>>. Acesso em: 26 maio. 2010, 17:00:00.
- PUNTER, T., CIOLKOWSKI, M., FREIMUT, B., JOHN, I., 2003, “Conducting on-line Surveys in Software Engineering”. In: *International Symposium on Empirical Software Engineering*, pp. 80-88, Frascati, Italy, October.
- ROBINSON, H., SEGAL, J. e SHARP, H., 2007, “Ethnographically-informed empirical studies of software practice”, ***Information and Software Technology***, v. 49, n. 6 (October), pp. 540-551.
- RUNESON, P., 2007, “Introduction to Case Study Research”, In: *IASESE (International Advanced School of Empirical Software Engineering)*, pp. 45-67, Madri, Setembro.
- SEAMAN, C. B., 2008, “Qualitative Methods”, In: SHULL, F., SINGER, J., SJØBERG, D. I. K. (eds.), 2008, ***Guide to Empirical Software Engineering***, pp. 35-62, London, Springer.
- SHULL, F., CARVER, J., TRAVASSOS, G.H., 2001, “An Empirical Methodology for Introducing Software Processes”. In: *8th European Software Engineering Symposium and 9th ACM SIGSOFT Symposium on the Foundations of Software Engineering (FSE-9) and 8th European Software Engineering Conference (ESEC)*, pp. 88-109, Vienna, Austria, September.
- SHULL, F., BASILI, V.R., CARVER, J., MALDONADO, J.C., TRAVASSOS, G.H., MENDONÇA, M., FABBRI, S., 2002, “Replicating Software Engineering Experiments: Addressing the Tacit Knowledge Problem.” In: *International Symposium on Empirical Software Engineering*, pp. 7-16, Nara, Japan, October.
- SHULL, F., MENDONÇA, M., BASILI, V., CARVER, J., MALDONADO, J. C., FABBRI, S., TRAVASSOS, G. H., FERREIRA, M. C., 2004, “Knowledge-Sharing Issues in Experimental Software Engineering”, In: *Empirical Software Engineering An International Journal*, v. 9, n. 1-2, p. 111-137, Boston, USA, September.

- SHULL, F., SINGER, J., SJØBERG, D. I. K. (eds.), 2008, *Guide to Empirical Software Engineering*, London, Springer.
- SJØBERG, D.I.K., DIBA, T., JORGENSEN, M., 2007, “The Future of Empirical Methods in Software Engineering Research”, In: *International Conference on Software Engineering (ICSE)*, pp. 123-156, Minneapolis, USA, May.
- SJØBERG, D.I.K., HANNAY, J.E., HANSEN, O., KAMPENES, V.B., KARAHASANOVIĆ, A., LIBORG, N.-K., REKDAL, A.C., 2005, “A Survey of 116 Controlled Experiments in Software Engineering”, *IEEE Transactions on Software Engineering*, v. 31, n. 9 (September), pp. 733-753.
- TICHY, W.F., 1998, “Should Computer Scientists Experiment More?”, *IEEE Computer*, v. 31, n. 5, p. 32-39. 1998.
- SANTOS, P.S.M., TRAVASSOS, G. H., 2008, “Colaboração entre Academia e Indústria: oportunidades para Utilização da Pesquisa-Ação em Engenharia de Software”, em: *Latino-americano em Engenharia de Software Experimental (ESELAW'08)*, pp. 46-58, Salvador, BH, Brasil, Outubro.
- TAYLOR, S.J., BOGDAN, R. (eds.), 1998, *Introduction to Qualitative Research Methods*, London, Wiley.
- TRAVASSOS, G.H., BARROS, M.O., 2003, “Contributions of In Virtuo and In Silico Experiments for the Future of Empirical Studies in Software Engineering”. In: *Proceedings of the WSESE03*, pp. 189- 200, Fraunhofer IRB Verlag, Roma, August.
- TRAVASSOS, G. H., SANTOS, P. S. M., MIAN, P. G., DIAS NETO A. C., BIOLCHINI, J., 2008, “A Environment to Support Large Scale Experimentation in Software Engineering”. In: *13th IEEE International Conference on Engineering of Complex Computer Systems*, pp. 92-100, Edinburgh, UK, April.
- TRIPP, D. 2005, “Action research: a methodological introduction”. Em: *Educação e Pesquisa*, v. 31, n. 3, São Paulo, Março.
- USCHOLD, M., GRUNINGER, M., 1996, “Ontologies: Principles, Methods and Applications”, *Knowledge Engineering Review*, Cambridge University Press, v. 11, n. 2, pp. 93-136.
- VIGDER M., VINSON, N. G., SINGER J., STEWART D., MEWS K., 2008, “Supporting Scientists’ Everyday Work: Automating Scientific Workflows”, *IEEE Software*, v. 25, n. 4 (August), pp. 21-44.

- YIN, R. K., 2003, *Case Study Research: Design and Methods*, 3 ed., London, Sage.
- W3C, w3c, 2010. Disponível em: <<http://www.w3.org/>>. Acesso em: 26 maio. 2010, 10:00:00.
- WIDMAN, L.E., 1989, “Expert system reasoning about dynamic systems by semi-quantitative simulation”, *Computer Methods and Program in Biomedicine - Artificial Intelligence in Medicine*, v. 6, n. 3, (June), pp. 229-247.
- WINCH, G., 1999, “Knowledge Management”, *Manufacturing Engineering*, v.3, n.4 (August), pp. 178-180.
- WÖHLIN, C., RUNESON, P., HÖST, M., OHLSSON, M., REGNELL, B., WESSLÉN A. (eds.), 2000, *Experimentation in Software Engineering – An Introduction*, 1 ed., London, Kluwer Academic Publishers.
- WOOLLARD, D., MATTMANN, C. A., MEDVIDOVIC, N., GIL, Y., (2008), “Scientific Software as Workflows: From Discovery to Distribution”, *IEEE Software*, v. 25, n. 4 (July), pp. 37-43.
- ZABOT, J. B. M., SILVA, L. C. M., 2002, *Gestão do Conhecimento: Aprendizagem e Tecnologia Construindo a Inteligência Coletiva*, 1 ed., São Paulo, Editora Atlas.
- ZELKOWITZ, M. V., WALLACE, D. R., BINKLEY, D.W., 2003, “Experimental Validation of New Software Technology”. In: *Lectures Notes on Empirical Software Engineering, Series on Software Engineering and Knowledge Engineering*, v. 12, n. 6 (August), pp. 34-67.

Anexo I

Modelos Ontológicos do Repositório de Conhecimento do eSEE

*Este anexo apresenta as os modelos conceituais formais
construídos para compor, juntamente com o glossário
de termos, o repositório de conhecimento do eSEE*

1.1. Introdução

Neste anexo são apresentados em detalhes os conceitos, suas definições e relacionamentos presentes na Ontologia de Pesquisa Científica e em suas principais subontologias. Para detalhes sobre demais subontologias, consulte o *link* <http://lens.cos.ufrj.br/esee>.

Na apresentação dos modelos ontológicos, adotamos a seguinte notação: as palavras em **negrito** referem-se a um determinado conceito de mesmo nome no modelo que está sendo apresentado. Nomes de relacionamentos estão destacados em *itálico*. Alguns conceitos estão presentes em mais de um modelo. A definição destes conceitos será apresentada somente uma vez e na primeira subseção que ocorrerem. Para visualizar em quais modelos um determinado conceito ocorre, basta acessar as facilidades de navegação disponibilizadas no *link* citado acima.

1.2. Ontologia de Pesquisa Científica

A Ontologia de Pesquisa Científica é o modelo central, do qual todas as demais subontologias derivam direta ou indiretamente. Ela contém os conceitos mais gerais em Engenharia de Software Experimental e vários de seus conceitos possuem uma subontologia que explicita conhecimento mais específico que, caso fosse representado na própria Ontologia de Pesquisa Científica, prejudicaria sensivelmente a legibilidade e navegação no modelo principal. A figura I-1 ilustra a Ontologia de Pesquisa Científica, que é descrita em detalhes no capítulo 3.

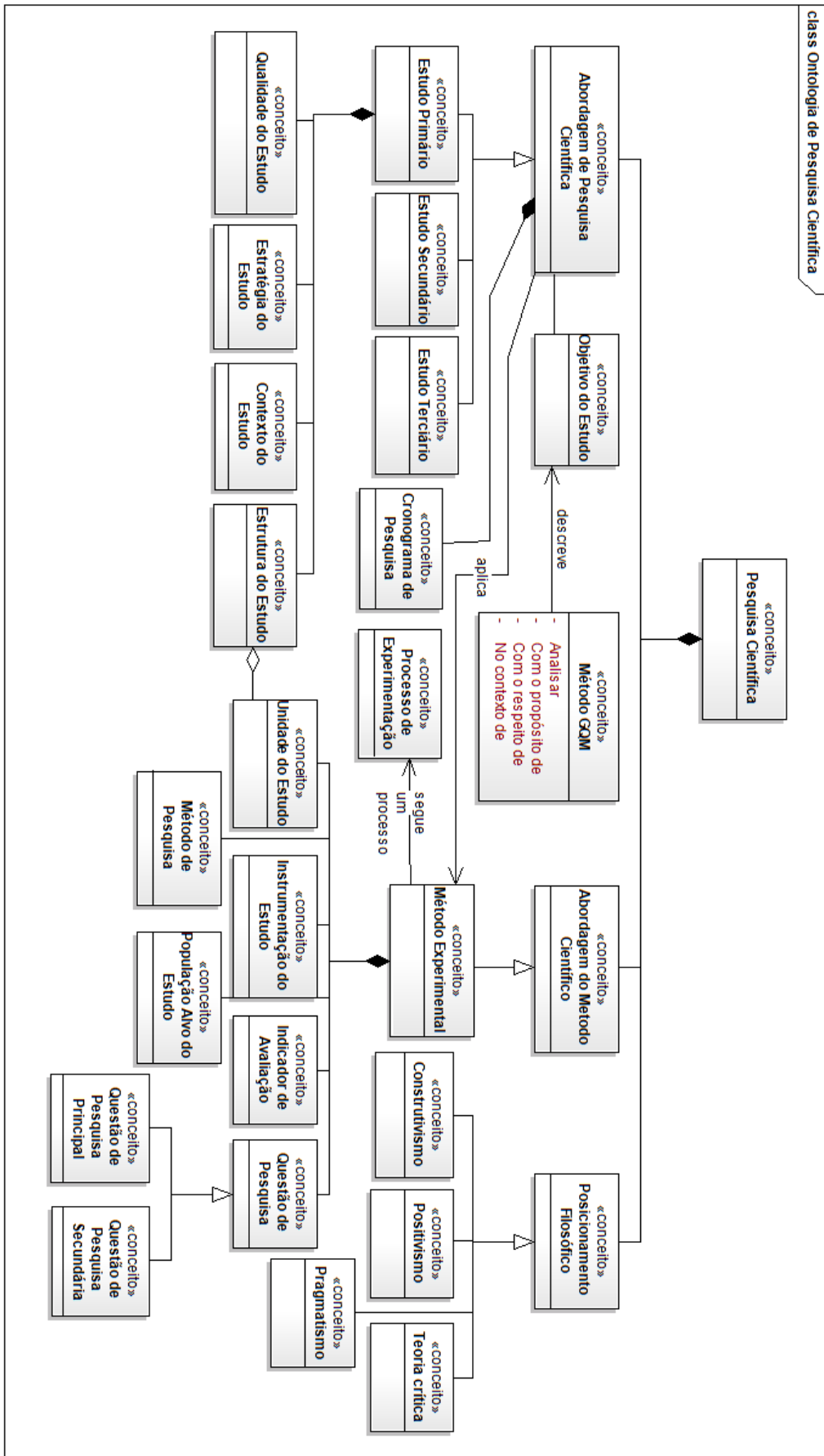


Figura III-1. Ontologia de Pesquisa Científica

A Ontologia de Pesquisa Científica original de Biolchini *et al.* foi formalizada utilizando a linguagem LINGO. Nesta formalização um novo corpo de conhecimento foi agregado sobre os subdomínios identificados por Lopes e Travassos (2008) como sendo importantes de serem considerados no planejamento de um estudo experimental. Em vista disto, para os domínios de **estratégia de estudo, método de pesquisa e ambiente de estudo**, cada qual com seu respectivo conceito na ontologia, novos modelos ontológicos foram construídos, de modo que modelos mais específicos expressem termos e relacionamentos pertinentes a um domínio em particular. Os novos modelos são denominados “subontologias” por designarem ontologias mais específicas e que estendem um modelo ontológico com conceitos mais gerais. Os modelos ontológicos construídos são:

- Subontologia de Estratégia de Estudo
 - Subontologia de Pesquisa-ação
 - Subontologia de Survey
 - Subontologia de Estudo Caso
 - Subontologia de Estudo de Observação
 - Subontologia de Estudo Controlado
- Subontologia de Método de Pesquisa
 - Subontologia de Método Qualitativo
 - Subontologia de Método Quantitativo
- Subontologia de Ambiente de Estudo
 - Subontologia de Estudo *In Vivo*
 - Subontologia de Estudo *In Vitro*
 - Subontologia de Estudo *In Virtuo*
 - Subontologia de Estudo *In Silico*

As subontologias relacionadas acima serão descritas nas subseções a seguir.

1.3. Subontologia de Estratégia de Estudo

Na Ontologia de Pesquisa Científica, o conceito de **estratégia de estudo** faz parte do domínio de **estudos primários** que, por sua vez, constitui uma **abordagem de estudo** (EASTERBROOK *et al.*, 2008) (JURISTO e MORENO, 2001). Uma estratégia de estudo, ou também chamado como método empírico, consiste em um conjunto de

práticas que definem como o objeto de estudo será avaliado; quais tipos de instrumentos serão utilizados; qual o rigor científico a ser empregado e entre outras características do projeto da pesquisa. A figura I-2 ilustra a subontologia de estratégia de estudo, descrita em detalhes no capítulo 3.

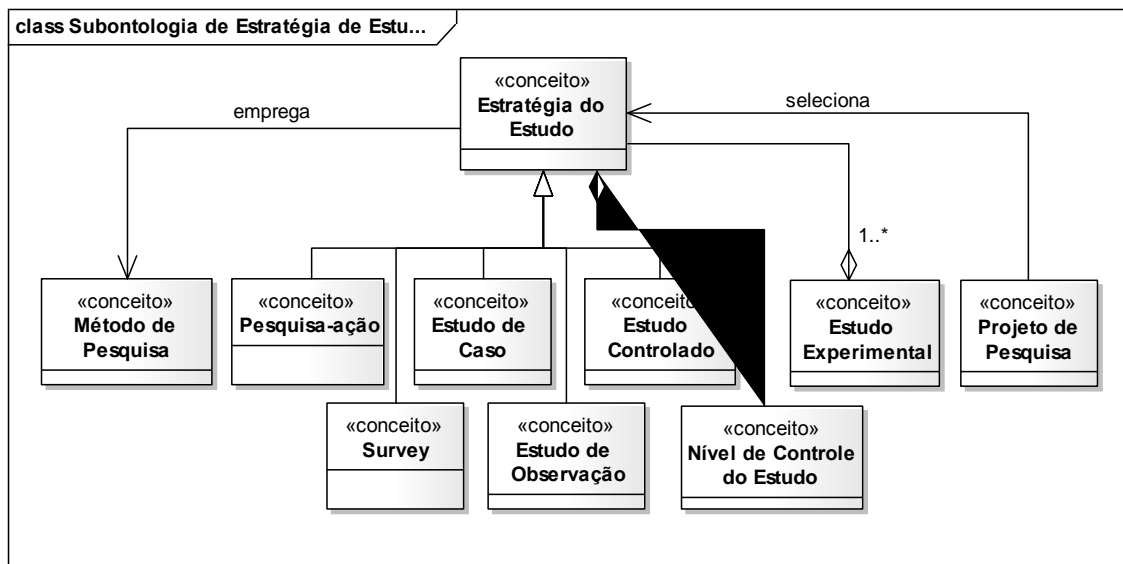


Figura III-2. Subontologia de Estratégia de Estudo

Constituem tipos de estratégia de estudo: **pesquisa-ação**, *survey*, **estudo de caso**, **estudo de observação** e **estudo controlado**. Para cada um desses conceitos foram construídos subontologias específicas, que são descritas a seguir.

1.3.1. Subontologia de Pesquisa-ação

A estratégia de **pesquisa-ação** tem como objetivo realizar de maneira simultânea pesquisa e ação. Está ligada a uma transformação em uma comunidade ou organização. A pesquisa envolve a melhor compreensão do fenômeno transformador pelo pesquisador. Em pesquisa-ação, pesquisador e participante colaboram ativamente no estudo. A partir da observação e avaliação do fenômeno observado, os participantes expõem suas impressões e dificuldades enfrentadas e o conhecimento sobre o processo transformatório é externalizado (EASTERBROOK *et al.*, 2008). Neste sentido, **pesquisa-ação gera aprendizado colaborativo**, no qual espera-se que tanto o pesquisador, quanto o participante, possam internalizar conhecimento ao final do estudo (TRIPP, 2005). A estratégia de **pesquisa-ação** é *conduzida sob* um pensamento filosófico denominado **teoria crítica**, uma abordagem teórica que, contrapondo-se à teoria tradicional, de tipo cartesiano, busca unir teoria e prática, ou seja, incorporar ao pensamento tradicional dos filósofos uma tensão com o presente. A **teoria crítica** julga

o conhecimento científico por sua habilidade para libertar as pessoas dos sistemas restritivos de pensamento (EASTERBROOK *et al.*, 2008). A subontologia de pesquisa-ação é ilustrada na figura I-3.

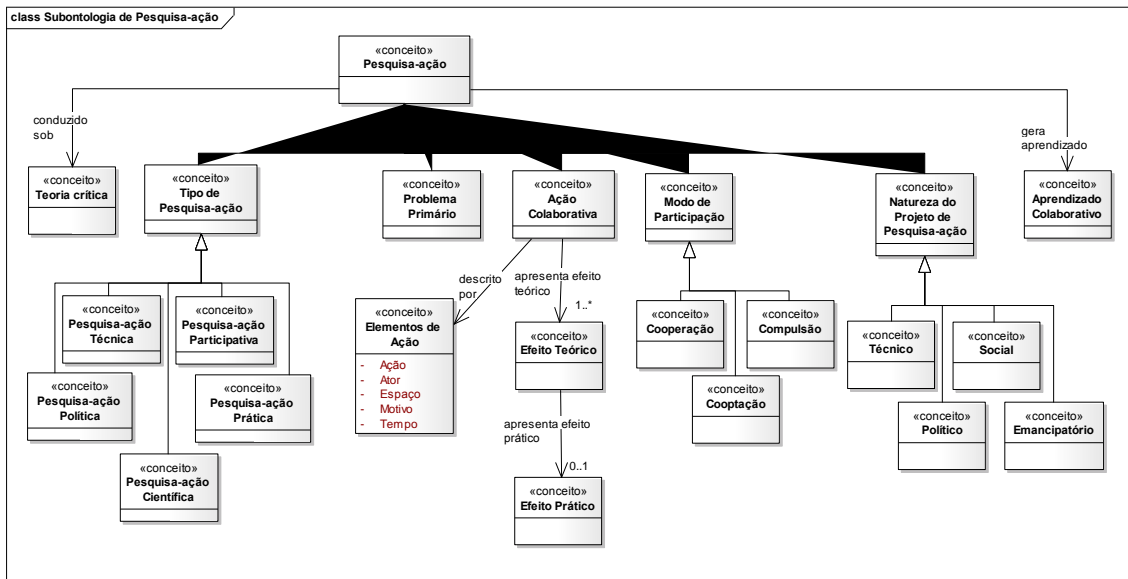


Figura III-3. Subontologia de Pesquisa-ação

São tipos de **pesquisa-ação**:

- **Pesquisa-ação técnica:** Tipo de pesquisa-ação em que as ações se baseiam numa prática pré-existente na literatura e que será aplicada à situação-problema pesquisada (TRIPP, 2005);
- **Pesquisa-ação participativa:** Tipo de pesquisa-ação que difere da pesquisa-ação convencional pelo fato dos participantes também atuarem como co-pesquisadores (BASKERVILLE, 1999);
- **Pesquisa-ação política:** Tipo de pesquisa-ação na qual a fonte de promoção de mudanças nas restrições sobre a prática advém das mais diversas formas de poder (TRIPP, 2005);
- **Pesquisa-ação prática:** Modalidade de pesquisa-ação em que o pesquisador escolhe ou projeta as mudanças (ações) a serem promovidas na situação social pesquisada (TRIPP, 2005);
- **Pesquisa-ação científica:** Também chamada de *action science*, é um tipo de pesquisa-ação que difere da pesquisa-ação convencional pelo fato de enfatizar as teorias em uso e tácitas que os participantes trazem para a prática (BASKERVILLE, 1999).

A **pesquisa-ação** possui um **problema primário**, que representa a causa do desejo de transformação que precisa ser aplicado na comunidade envolvida no estudo (BASKERVILLE, 1999). Além disto, possui uma **ação colaborativa**, que representa a essência deste tipo de estratégia: é a ação a ser realizada colaborativamente entre pesquisador e participante. Uma **ação colaborativa** *apresenta* um ou mais **efeitos teóricos**, cuja ocorrência é esperada pelos pesquisadores. Um efeito teórico pode ou não apresentar um **efeito prático**. Uma **ação colaborativa** é descrita em termos de **elementos de ação, que auxiliam** o tratamento dos dilemas da pesquisa por descrever a ação em termos dos atores envolvidos, o motivo da mesma, quando será aplicada e entre outras características (MELIN e AXELSSON, 2007).

São **modos de participação** no contexto de **pesquisa-ação** (TRIPP, 2005):

- **Cooperação:** O pesquisador seleciona os participantes, tendo estes como um parceiro em muitos aspectos (por exemplo, consultando o participante com frequência); contudo, o projeto continua sendo de responsabilidade do pesquisador. A maioria dos modos de participação é desse tipo;
- **Compulsão:** O participante não tem escolha em participar ou não do projeto, geralmente em função de uma restrição ou ordem de um superior;
- **Cooptação:** O participante é persuadido pelo pesquisador a ajudá-lo na pesquisa e aquele efetivamente aceita prover determinado serviço ao pesquisador.

A **natureza de um projeto de pesquisa-ação** constitui a intenção, seja científica, social ou política, na prática deste tipo de estratégia (TRIPP, 2005). Podem ser um dos tipos abaixo:

- **Técnico:** Pesquisador aplica uma prática existente em outro lugar e a implementa no seu contexto para melhorá-lo;
- **Social:** Participantes atuam com intenções que vão além de simplesmente melhorar a maneira de se realizar algo, podendo acarretar mudanças no pensamento do grupo social;
- **Político:** A participação dos envolvidos é mais profunda e pretende mudar "o sistema", isto é, quebrar restrições existentes;
- **Emancipatório:** Envolvidos pretendem mudar o status *quo* não apenas para um conjunto de pessoas, mas para todo um grupo social.

1.3.2. Subontologia de Survey

Um *survey* representa uma investigação realizada em retrospecto, quando, por exemplo, uma determinada tecnologia tem sido utilizada durante certo período de tempo (Shull *et al.*, 2008). Um survey permite capturar um "retrato instantâneo" da situação atual. Através de questionários ou entrevistas, dados são coletados de uma amostra de pessoas que representam um grupo específico. A figura I-4 ilustra a subontologia de survey. Para maiores detalhes sobre este modelo, consulte o capítulo 3.

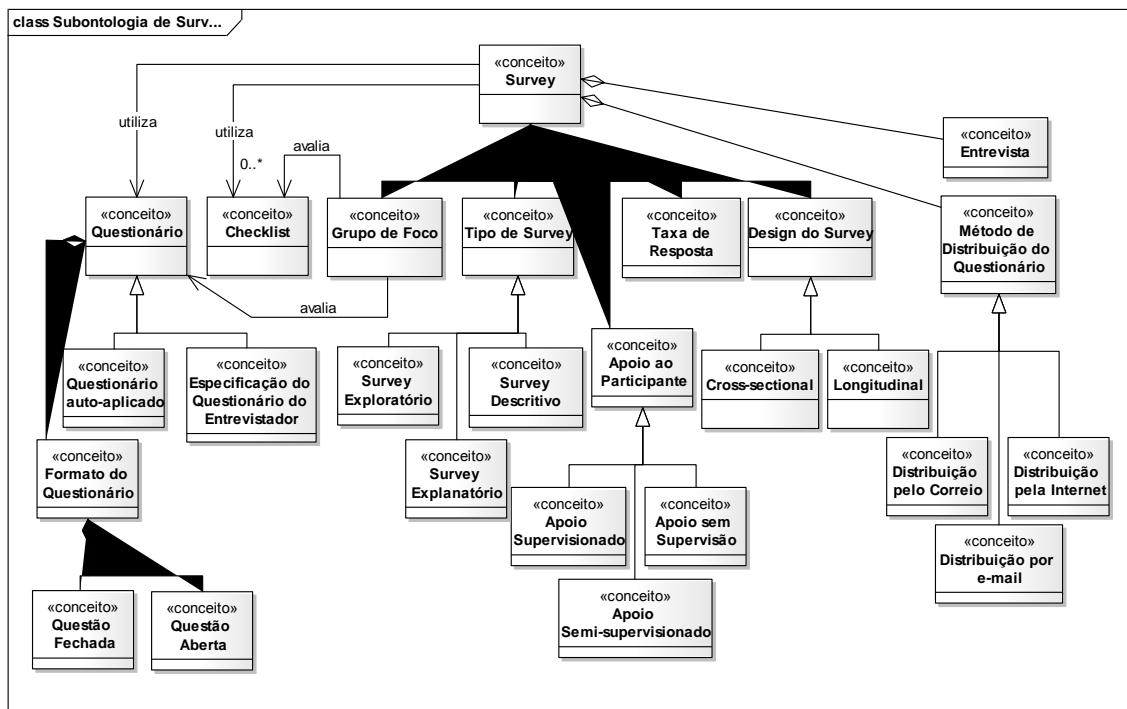


Figura III-4. Subontologia de survey

1.3.3. Subontologia de Estudo de Caso

Um **estudo de caso** representa uma estratégia que investiga um fenômeno contemporâneo dentro de um contexto real, especialmente quando os limites do fenômeno e do contexto não são claros (YIN, 2003). A subontologia de estudo de caso pode ser observada na figura I-5. Este tipo de estratégia *emprega* o conceito de generalização analítica, no qual as teorias desenvolvidas são utilizadas para comparar os resultados empíricos obtidos no estudo de caso (PERRY *et al.*, 2005).

Representam **tipos de estudo de caso**:

- **Estudo de caso exploratório:** Usado nas investigações iniciais de alguns fenômenos para *propor hipóteses* e construir teorias (EASTERBROOK *et al.*, 2008).

- **Estudo de caso confirmatório:** *Avalia teorias* existentes (EASTERBROOK *et al.*, 2008).
- **Estudo de caso descritivo:** Descreve uma seqüência de eventos ocorridos em campo (YIN, 2003).
- **Estudo de caso explanatório:** Ocorre julgamento entre duas explicações alternativas (YIN, 2003).
- **Estudo de caso causal:** Busca relacionamentos causais entre os conceitos (YIN, 2003).

Os **tipos de estudo de caso confirmatório, descritivo, causal e explanatório** se caracterizam por serem *conduzidos sob uma* perspectiva filosófica denominada **positivismo**. Este pensamento consiste em uma corrente teórica empenhada em explicar como a inteligência humana se desenvolve partindo do princípio de que o desenvolvimento da inteligência é determinado pelas ações mútuas entre o indivíduo e o meio. A idéia é de que nada, a rigor, está pronto, acabado, e de que, especificamente, o conhecimento não é dado, em nenhuma instância, como algo terminado. Ele se constitui pela interação do indivíduo com o meio físico e social, com o simbolismo humano, com o mundo das relações sociais; e se constitui por força de sua ação e não por qualquer dotação prévia, na bagagem hereditária ou no meio (EASTERBROOK *et al.*, 2008). O estudo de caso do tipo **exploratório** é **conduzido sob** a perspectiva do **construtivismo**, na qual o rejeita-se a idéia de que o conhecimento científico pode ser separado de seu contexto humano (EASTERBROOK *et al.*, 2008).

Segundo Easterbrook *et al.* (2008), os diferentes casos para estudo são melhor compreendidos como repetições do que meros membros de uma amostra. Para os **estudos de caso confirmatórios**, existem dois **tipos de casos de estudo: replicação literal e replicação teórica**. No primeiro, é esperado que os casos exibam os mesmo resultados, enquanto que no segundo a expectativa é que os casos tenham resultados diferentes por razões previsíveis (EASTERBROOK *et al.*, 2008).

A **unidade de análise** define o que é o "caso" do estudo de caso. Qual unidade de análise utilizar geralmente depende da questão de pesquisa. Depois de definida, a **unidade de análise** ainda pode ser modificada como resultado das descobertas feitas nos dados. Pode ser de dois tipos: **múltiplas unidades de análise** (estudo de caso embutido), nas quais O mesmo estudo de caso utiliza mais de uma unidade de análise. Neste caso, o estudo de caso denomina-se holístico. Um exemplo é um estudo de caso

sobre uma organização e que tira conclusões também sobre subunidades da mesma; **única unidade de análise** (estudo de caso holístico) (PERRY *et al.*, 2005).

O estudo de caso pode ser organizado sob quatro perspectivas (YIN, 2003):

- **Estudos com um único projeto:** examina os objetos entre uma única equipe e um único projeto;
- **Estudos com múltiplos projetos:** examina os objetos entre uma única equipe e vários projetos;
- **Estudos com projeto replicado:** examina os objetos entre várias equipes e um único projeto;
- **Estudos com participantes bloqueados:** examina os objetos entre várias equipes e vários projetos.

O relatório de resultados de um estudo de caso possui seis formas de estruturação: **analítico-linear, construção de teoria, suspensa, comparativa, cronológica, não-sequencial** (RUNESON, 2007).

Segundo Runeson (2007) as **fontes de evidência** das quais podem ser coletados dados para o estudo de caso podem variar conforme a decisão pode envolver ou não pessoas ou determinados tipos de artefatos. As fontes podem ser as seguintes (RUNESON, 2007):

- **Documentação:** Cartas, agendas, anúncios, relatórios de eventos, documentos administrativos e entre outros;
- **Observação Direta:** Visitas a campo. Em um estudo de caso conduzido com uma população de desenvolvedores de software, a observação direta pode facilitar, por exemplo, a caracterização de como os participantes trabalham;
- **Observação de Participante:** O observador não é passivo como na observação direta. Pode introduzir viés na condução do estudo pelo fato do observador participar ativamente;
- **Artefato Físico:** Ferramentas, instrumentos ou dispositivos. Artefatos podem ser utilizados como fonte de evidência durante uma visita a campo;
- **Registros de Arquivos:** registros pessoais, dados de *surveys* e outros registros organizacionais como mapas e gráficos;
- **Entrevistas:** podem ser entrevistas informais ou até mesmo um *survey* aplicado no contexto do estudo de caso com a elaboração de questionários e outros artefatos comuns a um *survey*.

1.3.4. Subontologia de Estudo de Observação

Em um **estudo de observação**, os participantes são observados enquanto desempenham uma determinada tarefa. Neste momento, a coleta de dados teve como objetivo fornecer informação sobre como uma determinada tarefa é executada utilizando a tecnologia estudada. Desta forma, os pesquisadores podem obter uma melhor compreensão sobre como uma nova tecnologia é aplicada e quais as eventuais dificuldades enfrentadas na prática. **Estudos de observação** são *conduzidos sob* a perspectiva filosófica do construtivismo. A subontologia de estudo de observação é ilustrada na figura I-6.

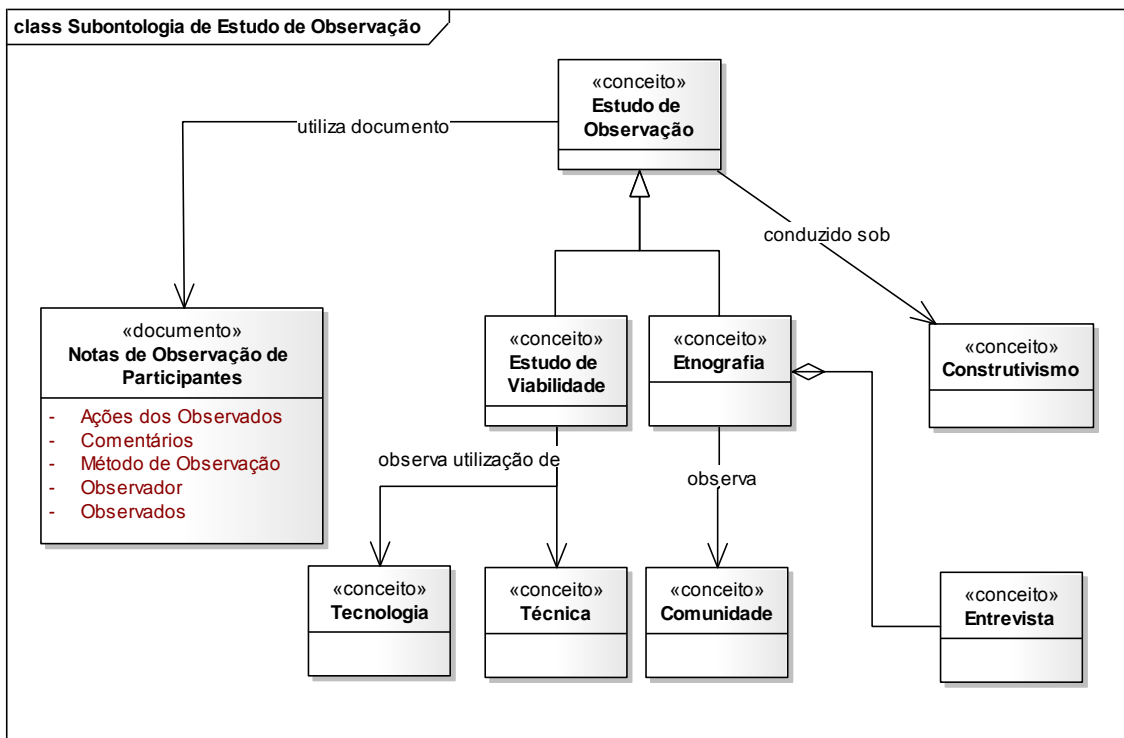


Figura III-6. Subontologia de estudo de observação

Na prática de **estudos de observação**, **etnografia** e **estudo de viabilidade** são os tipos mais recorrentes. Em **etnografia**, o pesquisador *observa* uma comunidade a fim de compreender as interações sociais existentes durante execução de uma determinada tarefa, sendo praticada comumente neste contexto **entrevistas**. Já em um **estudo de viabilidade**, o foco está em *observar a utilização* de uma técnica ou tecnologia, dando menor ênfase ao comportamento expresso pela comunidade (EASTERBROOK *et al.*, 2008).

1.3.5. Subontologia de Estudo Controlado

Estudos controlados objetivam manipular uma ou mais variáveis e controlar as demais a níveis fixos ou determinados. Em um estudo experimental, os participantes são atribuídos a diferentes tratamentos de forma aleatória. O objetivo é manipular uma ou mais variáveis e controlar todas as outras variáveis num valor fixo. O efeito da manipulação das variáveis é, então, medido e, com base nessa medição, análises estatísticas são conduzidas (WÖHLIN *et al.*, 2000). Este tipo de estratégia de estudo é **conduzido sob** a perspectiva filosófica do **positivismo**. A subontologia de estudo controlado é ilustrada na figura I-7.

Constituem tipos de estudos controlados (EASTERBROOK *et al.*, 2008):

- **Quasi-experimentos:** Estudos experimentais que tem tratamento, medidas de saída e unidades experimentais, mas não usam assinalamentos aleatórios para criar as comparações a partir das quais mudanças no tratamento e na causa são inferidas. A maioria dos estudos experimentais em Engenharia de Software se enquadra nesta categoria;
- **Experimentos aleatórios:** Caracterizado pelo uso de uma designação randômica inicial para inferir uma mudança na causa-tratamento. Neste experimento, os participantes são designados a receber o tratamento ou condição alternativa por meio de designação randômica.

Em estudos controlados, as **variáveis independentes**, também denominadas variáveis de entrada ou fatores, são manipuladas pelo pesquisador para que sejam observadas as repercussões desta manipulação no processo ou sistema. Esta observação ocorre sobre as **variáveis dependentes** (ou resultados), que serão medidas pelo pesquisador para avaliar o impacto das alterações nas **variáveis independentes**. Torna-se importante na execução do experimento a homogeneização das amostras da população-alvo, de modo a minimizar a influência das variáveis de contexto.

Para cada variável independente do experimento, o pesquisador define uma faixa de valores, que serão atribuídos à variável durante a execução do estudo. Cada combinação de valor e fator é chamada de **tratamento**.

Uma **teoria** é uma possível explicação de algum **fenômeno**, uma ocorrência observável no mundo real e que apresenta uma **causa** e um **efeito** (EASTERBROOK *et al.*, 2008). Qualquer teoria é composta por um conjunto de hipóteses. De acordo com PFLEEGER (2001), uma **hipótese** é uma tentativa teórica ou uma suposição do que se pensa, e que explica o comportamento do que se pretende explorar. Segundo PERRY *et*

al. (2000) as hipóteses definem as **questões de pesquisa** que esta procura responder, além das **variáveis dependentes**. A **questão de pesquisa** define a direção no desenvolvimento de estratégias de pesquisa. Essas perguntas podem ser agrupadas em perguntas de conhecimento – que são investigadas pela pesquisa empírica em engenharia de software e incluem perguntas exploratórias (perguntas de existência, perguntas de descrição e classificação, perguntas descritivas-comparativas), perguntas de base (perguntas de frequência e distribuição, perguntas processo-descritivas), perguntas de relacionamento e perguntas de causalidade (perguntas de causalidade, perguntas causalidade-comparativas, perguntas de interação causalidade-comparativas) – e perguntas de projeto, que são o foco da maior parte da pesquisa não-experimental em Engenharia de Software.

Deve-se pensar no estudo como um procedimento para fazer uma comparação. Desta forma, inicia-se em alto nível com questões abstratas e refina-se até chegar a um nível de questões concretas. As hipóteses abstratas são declarações em linguagem natural geralmente utilizadas nos termos diários, como por exemplo: “reunião é uma parte indispensável no processo de inspeção”. As hipóteses concretas são utilizadas em termos do projeto de estudo, como por exemplo: “equipes que aplicam inspeções com reuniões encontram mais defeitos do que equipes que aplicam inspeção sem reuniões”.

A hipótese principal de um experimento se chama **hipótese nula** e declara que não há nenhum relacionamento estatisticamente significante entre a causa e o efeito. O objetivo principal do experimento é, então, rejeitar a hipótese nula a favor de uma ou algumas **hipóteses alternativas**. A decisão sobre rejeição da hipótese nula pode ser tomada baseada nos resultados da sua verificação utilizando um teste estatístico. É aceito que a hipótese nula, empregada por conveniência para enunciar a hipótese de um experimento, não possa ser aceita pelos dados obtidos; podendo apenas ser rejeitada ou deixar de ser rejeitada. Situação semelhante acontece com hipóteses mais gerais, pois tecnicamente elas nunca são confirmadas, mas quando, por comodidade, usa-se essa expressão, entende-se que a hipótese foi submetida à rejeição e não rejeitada (CAMPBELL e COOK, 1979b).

O **objeto** é uma ferramenta usada para verificar o relacionamento causa-efeito numa teoria. Durante a execução do experimento, os tratamentos estão sendo aplicados a um conjunto de objetos e assim os resultados estão sendo avaliados. Os **participantes** de um experimento são os participantes que foram selecionados a partir da população de

interesse para conduzir o experimento. Para generalizar os resultados de um experimento a uma população desejada, o conjunto de participantes deve ser representativo para aquela população. Para atingir esta meta, parâmetros que influem no resultado do experimento, como o modo de seleção dos participantes e o tamanho do conjunto selecionado, devem ser considerados. A princípio, quanto maior é a variedade da população tanto maior deve ser o tamanho do conjunto de participantes. A combinação dos **objetos**, **participantes** e **tratamentos** se chama **rodada**, teste experimental ou ainda *trial* (WÖHLIN *et al.*, 2000).

No **projeto de estudos controlados** com mais de um fator, é comum a adoção de **projeto fatorial**, no qual são consideradas as interações entre estes fatores. O efeito sobre as variáveis dependentes pode ser dependente não apenas de cada fator separadamente, mas também das interações entre os fatores. Neste contexto, existem quatro tipos de **projeto de estudos controlados** (WÖHLIN *et al.*, 2000):

- **Projeto com mais de dois fatores com cada tratamento:** Aplica-se projeto fatorial 2^K ;
- **Projeto de dois fatores com dois tratamentos:** Aplica-se projeto fatorial dois à dois (projeto em dois estágios);
- **Projeto de um fator com dois tratamentos:** Projeto completamente aleatório, sendo comum a prática da comparação em pares;
- **Projeto de um fator com mais de dois tratamentos:** Projeto completamente aleatório.

O **projeto de estudos controlados** pode seguir três princípios fundamentais a depender do tamanho da população e do número de participantes formado a partir da mesma (WÖHLIN *et al.*, 2000):

- **Balanceamento:** Se forem designados os tratamentos de modo que cada tratamento tem igual número de indivíduos, tem-se um projeto balanceado. Balanceamento é desejável, pois, simplifica e reforça a análise estatística dos dados, porém não é necessário;
- **Randomização:** É um dos mais importantes princípios de projeto é a randomização. A randomização é aplicável em matéria de alocação dos objetos, os "participantes" e a ordem em que os testes serão realizados. Também usada para selecionar os participantes para que seja representativa da população de interesse;

- **Bloqueio:** Bloqueio é sistematicamente utilizado para eliminar o efeito indesejável na **comparação entre os tratamentos** através do bloqueio de **variáveis**. Dentro de um bloco, o efeito indesejado é o mesmo e nós podemos estudar o efeito dos tratamentos sobre esse bloco. Esta técnica aumenta a precisão do experimento, e pode ser usada quando o efeito do fator é conhecido e controlável.

O **método de atribuição de tratamentos aos participantes** varia conforme o **tipo do estudo controlado**. O **método não-aleatório**, empregado em **experimentos aleatórios**, caracteriza-se pela atribuição randômica dos tratamentos aos participantes sendo, portanto, viável apenas no caso população com tamanho representativo. Já o **método não-aleatório** é empregado em **quasi-experimentos** (WÖHLIN *et al.*, 2000).

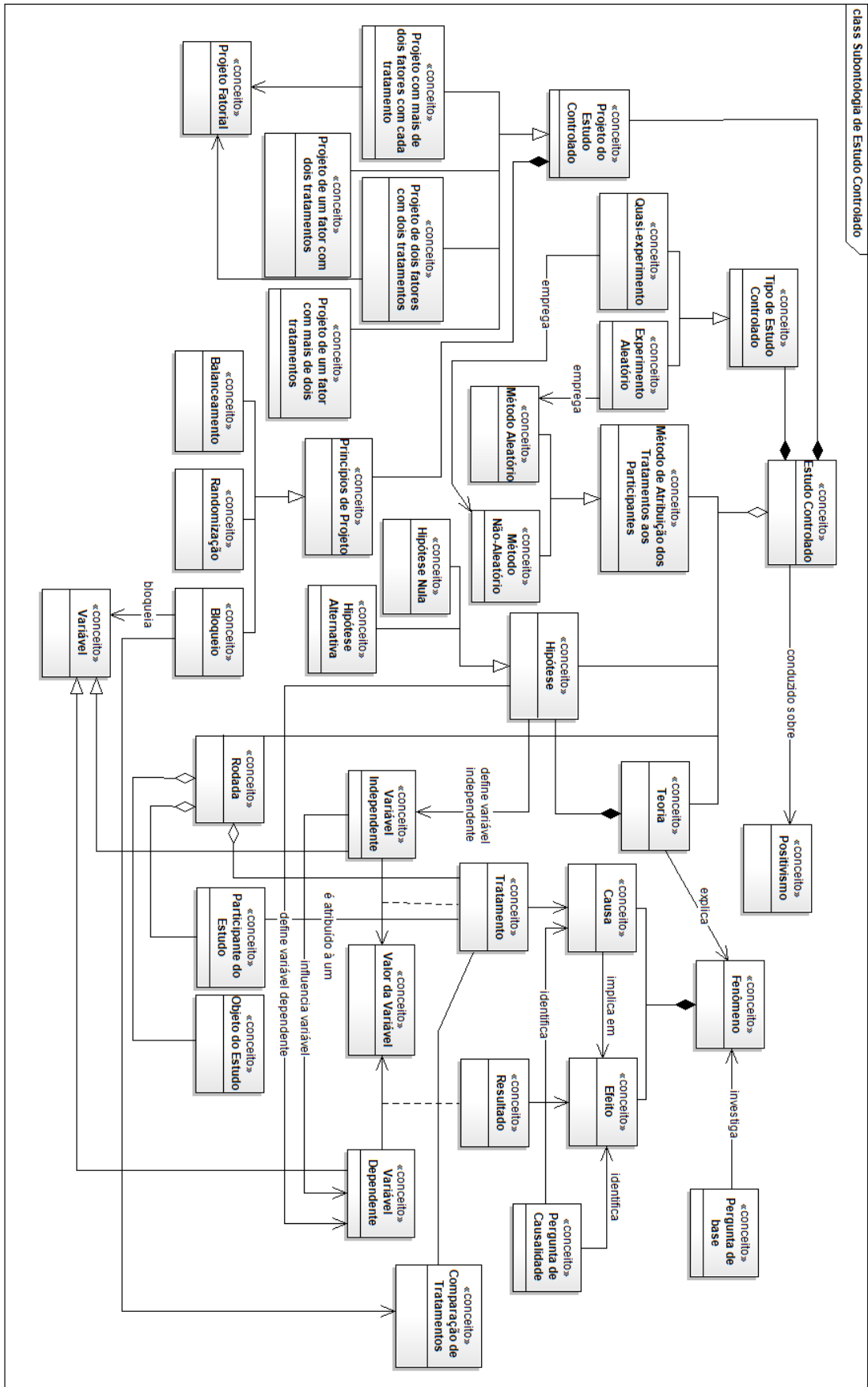


Figura III-7. Subontologia de estudo controlado

1.4. Subontologia de Método de Pesquisa

Um **método de pesquisa** corresponde à estratégia adotada para coleta e análise dos dados. Há três tipos de **métodos de pesquisa**: **quantitativo**, **qualitativo**, **semi-quantitativo** e **triangulação concorrente**. O capítulo 3 desta dissertação descreve detalhes a subontologia de método de pesquisa, ilustrada na figura I-8.

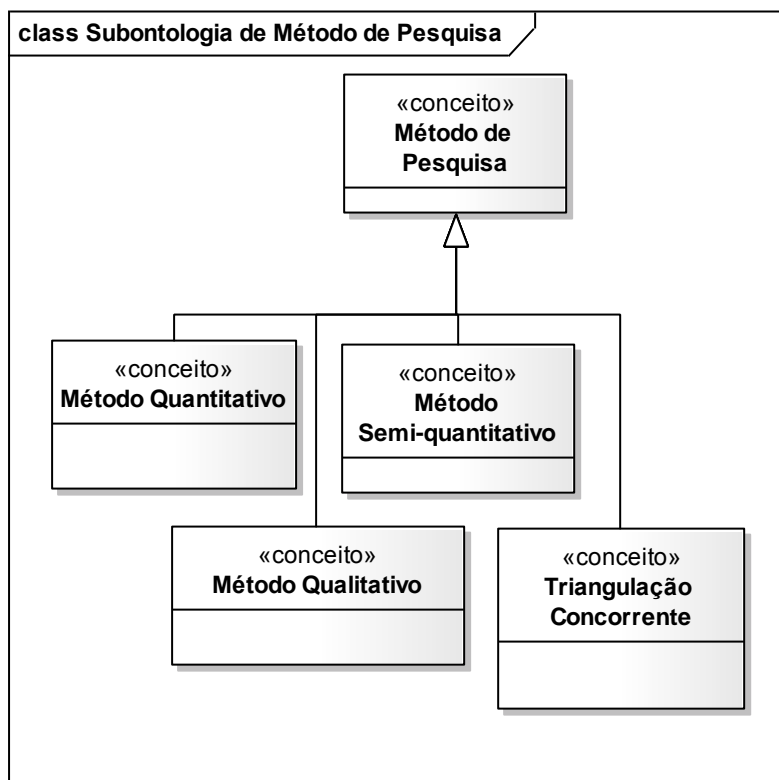


Figura III-8. Subontologia de Método de Pesquisa

1.4.1. Subontologia de Método Qualitativo

O **método qualitativo** busca focar na compreensão do fenômeno de pesquisa em seu contexto de ocorrência natural. De um modo geral, as principais técnicas aplicadas são a observação de participantes, entrevistas e análise de artefatos (TAYLOR e BOGDAN, 1998). As figuras I-9 e I-10 ilustram a subontologia em duas partes.

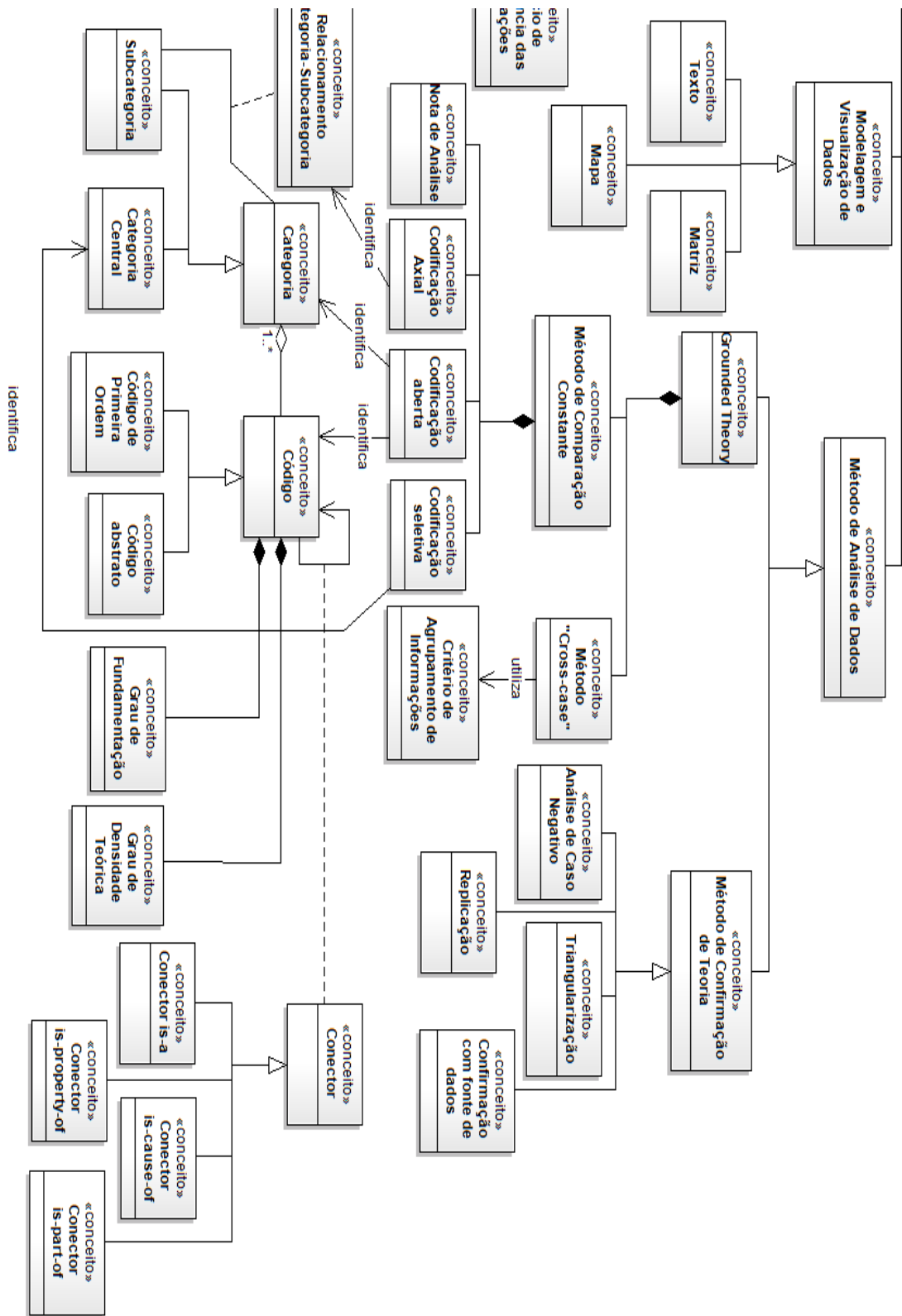


Figura III-10. Subontologia de Método Qualitativo (Parte 2)

Dentre os **métodos de coleta de dados** mais empregados, temos **entrevistas e observação de participantes**. Os **tipos de entrevistas** são **semi-aberta, semi-estruturada e estruturada**:

- **Semi-estruturada:** A entrevista geralmente segue um fluxo de conversação. Novas perguntas podem ser derivadas a partir de novas informações. Tipicamente, algumas questões abertas podem ser formuladas para melhorar a interação com o entrevistado (KITCHENHAM *et al.*, 2008).
- **Estruturada:** Uma lista fixa de perguntas rege a entrevista e são feitas exatamente como estão formulas, não ocorrendo, portanto, possíveis desvios. Dados de entrevistas estruturadas podem ser analisados estatisticamente (KITCHENHAM *et al.*, 2008).

Entrevistas podem ser conduzidas de forma **pessoal, por telefone ou por chat** (MENDONÇA, 2005):

- **Pessoal:** Do ponto de vista de padronização das perguntas e do potencial para transcrever as respostas, a aplicação pessoal de instrumentos é a mais problemática. Além de exigir treinamento para os aplicadores e para as pessoas transcreverem as respostas (especialmente as perguntas abertas), é o método mais demorado e mais caro. Sua vantagem é permitir acesso a informações mais delicadas;
- **Por telefone:** Do ponto de vista de padronização das perguntas e potencial para transcrever respostas, a entrevista por telefone – especialmente com apoio de computador – apresenta bons resultados. Embora também seja necessário o treinamento dos entrevistadores, reduz-se consideravelmente o uso do papel, visto que as perguntas são apresentadas na tela do computador para o entrevistador, que as lê para o entrevistado. A seqüência de perguntas pode ser programada de forma que, dependendo da resposta, uma ou outra pergunta seja indicada para ser a próxima. Admitindo que nem toda a população tem acesso ao telefone, é preciso atentar para a representatividade da população-alvo e da amostra atingida;
- **Por internet:** É possível acessar pessoas geograficamente distantes, entretanto a população-alvo atingível é mais restrita (GÜNTHER, 1999). Apresenta baixo custo em função da existência de facilidades gratuitas para condução de entrevistas.

Constituem o conceito de **entrevista** os conceitos elementares de **entrevistado**, **entrevistador**, **questionário** e suas **perguntas**, cada qual com uma respectiva **resposta**. Outro **método de coleta de dados** é a **observação de participantes**, cujos métodos de aplicação podem ser:

- **Pensar em voz alta (*think aloud*)**: Pesquisadores solicitam aos participantes a “pensar em voz alta” enquanto realizam uma determinada tarefa, dizendo claramente cada etapa o que estão fazendo. A tarefa pode ocorrer naturalmente no trabalho ou ser pré-determinada pelo pesquisador, que deve estar sempre lembrando os participantes de falar em voz alta cada passo realizado. O pesquisador não pode interferir no processo de resolução dos problemas. Sua aplicação é bastante comum para determinar ou verificar modelos cognitivos enquanto participantes, geralmente engenheiros de software, conduzem atividades de programação.
- ***Synchronized Shadowing***: O pesquisador segue o participante para onde ele for e registra suas atividades, podendo ocorrer enquanto o participante estiver disposto a participar. Neste tipo de método, o pesquisador observa apenas um participante por vez, ao contrário da observação tradicional.
- **Observação Tradicional**: Pesquisador observa um participante em seu trabalho ou executando uma atividade relevante para o estudo. O pesquisador pode fazê-lo em um grupo de participantes.

Após uma sessão de observação de participantes, os pesquisadores envolvidos podem se reunir para avaliar a consistência dos dados registrados através de um **exercício de concordância**, no qual as partes são convidadas a explicar o que foi observado e gerar discussões sobre os dados (KITCHENHAM *et al.*, 2008). Os dados de uma análise qualitativa podem ser modelados e visualizados na forma de **texto**, **matriz** ou **mapas**.

Os **métodos de análise de dados** podem ser de **confirmação de teoria** e **grounded theory**. O **método de confirmação de teoria** contribui para reforçar ou "confirmar" uma determinada proposição após coleta de dados. Baseia-se na identificação de ameaças à validade do estudo (SEAMAN, 2008). Pode variar entre cinco abordagens (SEAMAN, 2008):

- **Análise de caso negativo**: Importante método de análise para confirmar hipóteses. Consiste na procura exaustiva de evidências que venham a contradizer

logicamente a proposição gerada, conduzindo a uma revisão desta proposição contra dados existentes e recém-coletados, e então continuando com a busca por novas evidências contraditórias. Esta busca pode incluir a seleção de novos casos para estudo, de forma a aumentar representatividade, assim como novas fontes e tipos de dados (JUDD *et al.*, 1991 *appud* SEAMAN, 2008);

- **Triangularização:** Consiste na junção de diferentes tipos de evidências para apoiar uma proposição. As evidências podem ser provenientes de diferentes métodos e ter diferentes formas (entrevistas, observações, documentos e entre outros), ou ainda vir de outro estudo, o que denota que a triangularização inclui também princípios de replicação. Pode ainda incluir a execução de algum método quantitativo. Um exemplo clássico de aplicação de triangularização é combinar testes estatísticos de uma hipótese que foi gerada de forma qualitativa (SEAMAN, 2008);
- **Confirmação com fonte de dados:** Método aderente a estudos em Engenharia de Software. Envolve obter explicações e novas informações dos participantes dos quais os dados foram coletados. Neste sentido, os dados são apresentados, formal ou informalmente, de modo que eles possam acrescentar novas informações ou ainda auxiliar na interpretação dos mesmos. Desta forma, os participantes compreendem como os dados foram obtidos e se sentem parte do processo, podendo auxiliar nas conclusões finais. É um método cuja aplicação é comum em estudos de inspeção (SEAMAN, 2008);
- **Replicação:** Embora seja referido na literatura como um método de análise de dados, consiste na aplicação dos mesmos instrumentos e procedimentos de um estudo original, mas alterando a alocação de tratamentos aos participantes ou ainda alterando a questão de pesquisa. Os dados não precisam ser necessariamente coletados da mesma maneira que o estudo original (SEAMAN, 2008).

Outro **método de análise de dados** é o **Grounded Theory** (ou Teoria Fundamentada nos Dados), que utiliza um conjunto de procedimentos sistemáticos de análise dos dados para gerar, elaborar e validar teorias substantivas sobre fenômenos essencialmente sociais, ou processos sociais abrangentes (BANDEIRA-DE-MELLO e CUNHA 2003). Seus autores, Glaser e Strauss, afirmam que existem dois tipos básicos de teorias: as formais e as substantivas (BIANCHI e IKEDA 2006). O primeiro tipo é

composto das teorias conceituais e abrangentes, enquanto que o segundo tipo é específico para determinado grupo ou situação e não visa generalizar além da sua área substantiva. Segundo Bandeira e Cunha (2006), teoria é um conjunto integrado de proposições que explicam a variação da ocorrência de um fenômeno social subjacente ao comportamento de um grupo ou à interação entre grupos. A essência do método **Grounded Theory** é que a teoria substantiva emerge dos dados, ou seja, é uma teoria derivada de dados sistematicamente coletados e analisados. Embora a finalidade do método **Grounded Theory** seja a construção de teorias substantivas, sua utilização não necessariamente precisa ficar restrita apenas aos pesquisadores que tem esse objetivo de pesquisa. Segundo Strauss e Corbin (1998), o pesquisador pode usar apenas alguns de seus procedimentos para satisfazer seus objetivos de pesquisa. Existem dois tipos de abordagens para métodos baseados em **Grounded Theory**: **cross-case** e de **comparação constante**.

No método “**cross-case**”, os dados são agrupados em “casos” *utilizando* diferentes **critérios de agrupamento**. Cada caso é então analisado internamente, avaliando cada um de seus dados, e depois comparativamente, procurando por semelhanças e diferenças. Quanto aos critérios, os dados podem ser agrupados conforme o número de pessoas envolvidas, tipo de produto, fonte (entrevista, observação) ou simplesmente em pares aleatórios (SEAMAN, 2008).

No método de **comparação constante**, a idéia é de codificação (“coding”), que é o processo de analisar os dados. Durante a codificação são *identificados* **códigos e categorias**. Um **código** (ou conceito) dá nome a um fenômeno de interesse para o pesquisador; abstrai um evento, objeto, ação, ou interação que tem um significado para o pesquisador (STRAUSS e CORBIN 1998). **Categorias** são agrupamentos de conceitos unidos em um grau de abstração mais alto.

O processo de codificação pode ser dividido em três fases: **codificação aberta, axial e seletiva**. A **codificação aberta** envolve a quebra, a análise, a comparação, a conceituação e a categorização dos dados. Segundo Bandeira e Cunha (2006), nas fases iniciais da **codificação aberta**, o pesquisador explora os dados examinando minuciosamente aquilo que lhe parece relevante devido à leitura intensiva dos textos. Na fase de **codificação aberta** os incidentes ou eventos são agrupados em **códigos** através da comparação incidente–incidente.

Os **códigos** gerados podem ser classificados como: **códigos de primeira ordem**, diretamente associados às citações (chamados **códigos in vivo**); e **códigos abstratos** ou teóricos, associados a outros códigos, sem necessariamente estarem ligados a alguma citação. Também na **codificação aberta**, é realizada a criação de categorias que agregam os **códigos** para reduzir o número de unidades com que o pesquisador irá trabalhar.

Após a identificação de **categorias** conceituais pela **codificação aberta**, a **codificação axial** examina as relações entre as **categorias** que formam as proposições da teoria substantiva (BANDEIRA e CUNHA, 2006). Explicitam-se causas e efeitos, condições intervenientes e estratégias de ação, em proposições que devem ser testadas novamente nos dados. As relações entre os **códigos** (**conectores** segundo GLASER (1992)) podem ser definidas pelo próprio pesquisador. Na linha proposta por Strauss e Corbin (1998), essas relações formam o que os autores denominam de paradigma: condições causais, intervenientes, conseqüências e estratégias de ações/interações. Bandeira (2006) propõe, na linha proposta por Strauss e Corbin (1998), um conjunto de conectores:

- **Is-a:** O código-origem é um tipo, ou forma, do código-destino. É definido por um padrão de variação dimensional ao longo das propriedades da categoria (código-destino);
- **Is-property-of:** O código-origem é propriedade da categoria (código-destino);
- **Is-cause-of:** O código-origem (condição causal) causa a ocorrência do código-destino;
- **Is-part-of:** O código-origem é uma parte que compõe juntamente com outras partes o código-destino.

Por fim, a **codificação seletiva** refina todo o processo identificando a **categoria central da teoria**, com a qual todas as outras estão relacionadas. A **categoria central** (“core category”) deve ser capaz de integrar todas as outras categorias e expressar a essência do processo social que ocorre entre os envolvidos. Esta **categoria central** pode ser uma categoria existente, ou uma nova categoria pode ser criada.

1.4.2. Subontologia de Método Quantitativo

O conceito **método quantitativo** consiste na representação numérica de observações com propósito de descrever e explicar um fenômeno. A pesquisa

quantitativa inicia-se com a coleta de dados, seguida da aplicação de vários métodos estatísticos descritivos e de inferência (TAYLOR e BOGDAN, 1998). Uma vantagem desta abordagem é permitir comparações e análises estatísticas (WÖHLIN *et al.*, 2000). As figuras I-11 e I-12 ilustram a subontologia em duas partes.

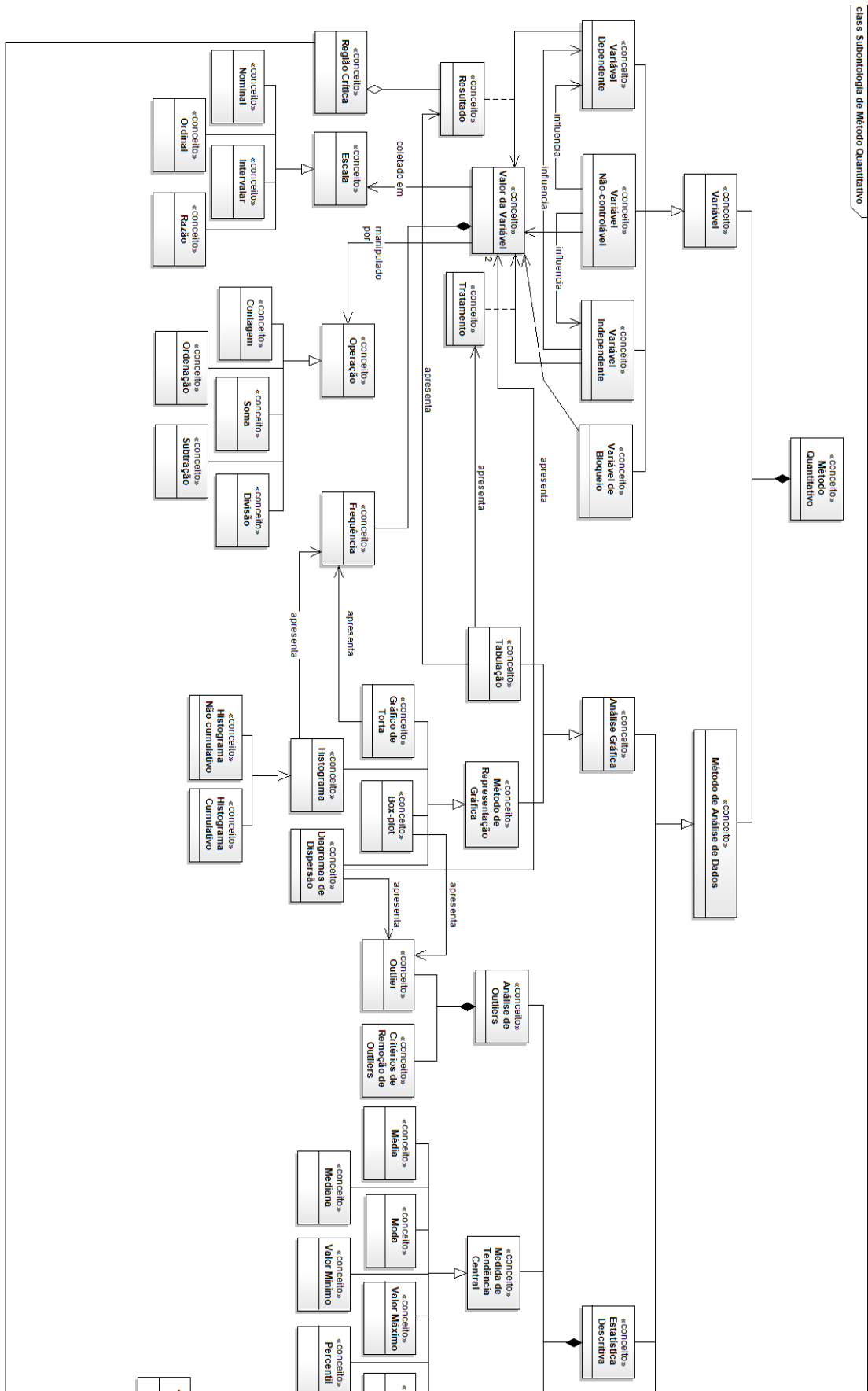


Figura III-11. Subontologia de Método Quantitativo (Parte 1)

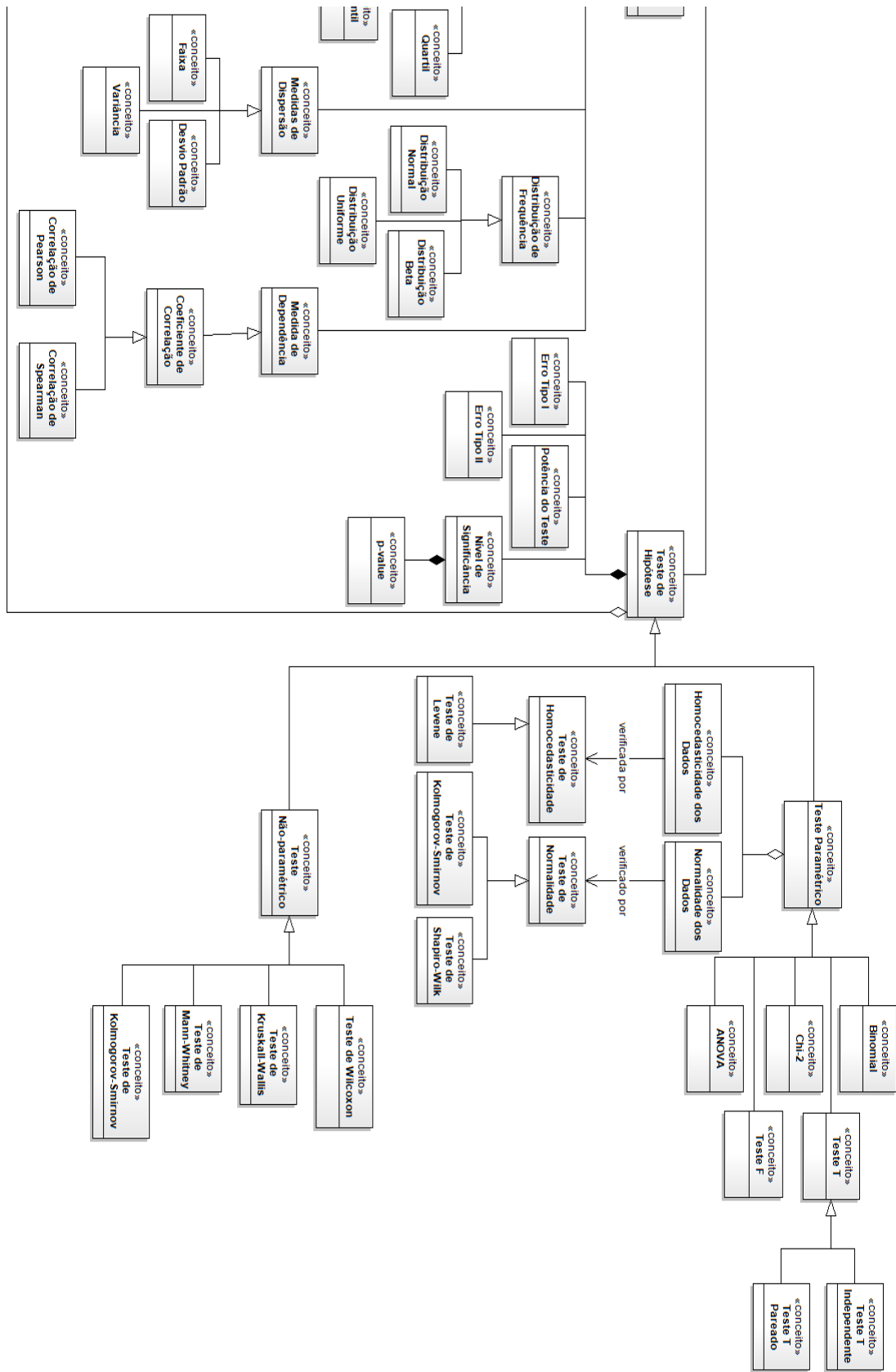


Figura III-12. Subontologia de Método Quantitativo (Parte 2)

No contexto de **método quantitativo**, as **variáveis** representam os elementos sob os quais se realizam transformações (**variáveis independentes**) ou para observação de efeitos representativos no contexto do estudo experimental (**variáveis dependentes**). Como os efeitos que desejamos observar estão relacionados às variáveis dependentes, seus valores constituem o **resultado** propriamente dito, que está numa **região crítica** específica. Uma **região crítica** é um conjunto de resultados de um teste estatístico que direciona para a rejeição da hipótese nula. Este teste estatístico também é referido como **teste de hipótese**, e verifica se é possível rejeitar a **hipótese nula**, de acordo com um conjunto de dados observados e suas propriedades estatísticas. Os testes existentes comparam médias entre grupos de participantes realizando tratamentos diferentes. Conceitos importantes envolvidos em **testes de hipótese** são o **erro tipo I** (indica um relacionamento entre causa e efeito e o relacionamento real não existe), **erro tipo II** (não indica o relacionamento entre causa e efeito, mas existe este relacionamento), **potência de teste** e **nível de significância**. A **potência do teste** indica a probabilidade de rejeitar a **hipótese nula** quando esta é falsa, ou seja, a probabilidade de decisão correta baseada na **hipótese alternativa**. O **nível de significância** indica a probabilidade de cometer um **erro tipo-I**, sendo que, no contexto de **nível de significância**, há o **p-value**, que representa o menor nível com que se pode rejeitar a hipótese nula (WÖHLIN *et al.*, 2000).

Há dois tipos de **testes de hipótese**: **paramétricos** e **não-paramétricos**. Os testes paramétricos utilizam fórmulas fechadas, derivadas de propriedades de distribuições de frequência conhecidas (tais como equação da curva, da curva acumulada, simetria e entre outras). Já os testes não paramétricos devem ser utilizados quando os dados coletados não atendem aos pressupostos esperados pelos **testes paramétricos**. São menos poderosos que estes, mas não presumem distribuições de probabilidade nos dados. Utilizam rankings dos valores observados ao invés dos valores propriamente ditos. Há cinco tipos de **testes paramétricos**: **binomial**, **chi-2**, **ANOVA**, **teste T (independente** ou **pareado)** e **teste F** (MAXWELL, 2002).

Ainda no tocante aos **testes paramétricos**, temos os conceitos de **homocedasticidade** e **normalidade dos dados**. A existência de homocedasticidade nos dados implica em variância constante entre os conjuntos de dados que serão testados, ou seja, a variância de um subgrupo não é maior que a de outro. Pode ser verificada por testes específicos, tais como o de **Levene**. A normalidade dos dados diz respeito a

valores que se concentram simetricamente em torno de uma média e quanto maior a distância desta média, menor a frequência das observações. Para verificar a normalidade, os testes de **Kolmogorov-Smirnov** e de **Shapiro-Wilk**, sendo que o primeiro é freqüentemente utilizado para identificar normalidade em variáveis com pelo menos 30 valores (MAXWELL, 2002).

Dentre os testes não-paramétricos que podem ser executados, temos os **testes de Wilcoxon, Kruskal-Wallis, Mann-Whitney e Kolmogorov-Smirnov**. O teste de **Kruskal-Wallis** é uma alternativa não-paramétrica para a análise de variância. Como grande parte dos testes paramétricos, este teste se baseia na substituição dos valores por seus rankings no conjunto de todos os valores. A hipótese nula é rejeitada com nível de significância α se K for maior que o valor da distribuição Chi-2 com n graus de liberdade para α . O teste de **Mann-Whitney** é uma alternativa não paramétrica para o teste T. Requer que as amostras sejam independentes, com dados contínuos e nas escalas ordinal, intervalar ou razão. Para a realização do teste as observações das amostras são reunidas em um único grupo, que é ordenado. As amostras são transformadas em rankings dentro do grupo e calcula-se o somatório dos rankings da menor amostra (MAXWELL, 2002).

Sobre os tipos de **variáveis** existentes, há ainda as **variáveis não-controláveis** e as **de bloqueio**. As **variáveis não-controláveis** são variáveis de contexto que, de algum modo, perturbam o processo ou sistema, mas sobre as quais não se tem controle adequado (WÖHLIN *et al.*, 2000).

Teste de hipótese constitui um dos tipos de **métodos de análise quantitativa de dados**. Há ainda os métodos por **análise gráfica e estatística descritiva**. A **análise gráfica** permite ao pesquisador um melhor "entendimento" dos dados antes de proceder com a análise propriamente dita. Pode ser feita via **tabulação**, que *apresenta os tratamentos e os resultados*, ou por um **método de representação gráfica**, que pode ser dos tipos (MAXWELL, 2002):

- **Gráfico de torta:** Apresenta a frequência relativa (ou percentual) de ocorrência dos dados, dividindo estes em um conjunto de classes distintas e apresentando-os como fatias de um círculo.
- **Box-plot:** Apresentam a distribuição de dados quantitativos.
- **Histograma:** Método comum de apresentação de dados numéricos e em qualquer escala, pois envolve apenas contagem. Há os **histogramas**

cumulativos (Apresenta os valores observados para uma variável de interesse no domínio da frequência.) e **não-cumulativos** (Apresenta os valores observados para uma variável de interesse no domínio da frequência.).

- **Diagramas de dispersão:** Representação dos valores observados para duas ou mais variáveis através de gráficos cartesianos.

Estatística descritiva apóia a análise inicial dos dados, medindo as dependências e relacionamentos entre eles. Transmite uma visão geral de como o conjunto de dados está distribuído. Em **estatística descritiva**, um importante conceito é o **medida de tendência central**, que indica o meio do conjunto de valores observados. Representam medidas de tendência central (MAXWELL, 2002):

- **Média:** Considerada como o centro de gravidade dos dados coletados. É calculada pelo somatório dos valores coletados, dividido por sua quantidade.
- **Mediana:** Valor do meio de um conjunto de dados, ou seja, o número de valores coletados que está abaixo da mediana deve ser o mesmo que está acima. É calculada colocando os valores em ordem crescente ou decrescente e selecionando o elemento central. Em caso de número par de valores, é calculada pela média dos valores centrais.
- **Moda:** representa o valor mais comum dentre o conjunto de valores coletados. É calculada pela contagem do número de ocorrências (frequência) de cada valor, selecionando o mais comum. Se dois ou mais valores ocorrem com a maior frequência, os valores coletados possuem diversas modas.
- **Valor máximo:** Representa o maior valor entre os dados que foram coletados.
- **Valor mínimo:** Representa o menor valor entre os dados que foram coletados.
- **Percentil:** É o caso geral da mediana, que é conhecida como percentil 50%. Em uma amostra de 100 elementos, o percentil X% é o valor que divide a amostra em X valores menores que ele e (100-X) valores maiores que ele.
- **Quartil:** São os valores que representam o percentil 25% (ou primeiro quartil), a mediana (segundo quartil) e o percentil 75% (terceiro quartil).

Os dados a serem analisados possuem uma **distribuição de frequência**. Podemos citar a **distribuição uniforme**, **beta** e **normal**. A **distribuição normal** possui o formato de um sino, com as pontas se estendendo a direita e esquerda do centro. A curva é simétrica em relação a sua média e a largura do sino é proporcional ao seu

desvio padrão. Assim, a curva pode ser descrita matematicamente apenas com base em sua **média e desvio padrão**.

Na **estatística descritiva**, há o conceito de **outliers**, que são valores extremos observados e que estão muito distantes dos demais valores observados. Estes dados podem representar erros no conjunto de valores observados e usualmente são removidos deste conjunto antes de se aplicar a técnicas de inferência estatística. Os **outliers** podem ocorrer por problemas de aplicação da sistemática prevista no projeto do estudo, por erros de digitação, problemas de interpretação ou motivação dos participantes. A **análise de outliers** praticada na **estatística descritiva** consiste no procedimento de verificação das origens de cada **outlier** e decisão sobre sua remoção, ou não, do conjunto de dados, utilizando para isto critérios de que podem considerar a distância da média ou da mediana (MAXWELL, 2002).

Outras medidas importantes são as **medidas de dispersão**, que medem o quanto os valores coletados estão dispersos ou concentrados em torno de seu valor central. São **medidas de dispersão** (MAXWELL, 2002):

- **Faixa:** É a diferença entre o maior e o menor valor dentre os valores coletados.
- **Variância:** Soma do quadrado da diferença entre cada valor e a média dos valores coletados, dividida pelo número de valores coletados menos 1.
- **Desvio padrão:** Raiz da variância, sendo a medida de dispersão mais comumente utilizada.

Quando duas ou mais variáveis estão relacionadas em um estudo, é útil calcular o grau de dependência entre elas. Neste caso, consideram-se **medidas de dependência**. Elas determinam a força e direção do relacionamento entre duas ou mais variáveis avaliadas quantitativamente. Um tipo de **medida de dependência** é o **coeficiente de correlação**, que consiste na correlação entre duas variáveis varia entre -1 e 1. A correlação -1 indica que um valor alto em uma variável normalmente ocorre em conjunto com um valor baixo da segunda variável. A correlação 1 indica que um valor alto em uma variável normalmente ocorre em conjunto com um valor alto da segunda variável. A correlação próxima de zero indica que não podemos inferir nenhum relacionamento entre as variáveis. São tipos de **coeficientes de correlação** usualmente empregados em **estatística descritiva** (MAXWELL, 2002):

- **Correlação de Pearson:** Quantifica a força de associação linear entre duas variáveis e descreve o quanto uma linha reta se ajustaria através da

representação cartesiana de seus valores. O coeficiente assume que os valores das variáveis seguem aproximadamente distribuições normais. Devido à forma da distribuição normal, esta condição é indicada pela formação de uma nuvem em forma de elipse em um gráfico de dispersão que apresente estes valores.

- **Correlação de Spearman:** Baseia-se no ranking dos valores coletados em seu conjunto, não nos valores propriamente ditos. Pode ser aplicado sobre valores em uma escala ordinal (não apenas intervalar e razão).

1.5. Subontologia de Ambiente de Estudo

A subontologia de ambiente de estudo (Figura I-13) ilustra os conceitos que representam os tipos de contextos de estudo, que são: **estudo *in vivo***, ***in vitro***, ***in virtuo*** e ***in silico*** (TRAVASSOS e BARROS, 2003). As subseções a seguir explicitam os conceitos relacionados a cada tipo de contexto de estudo.

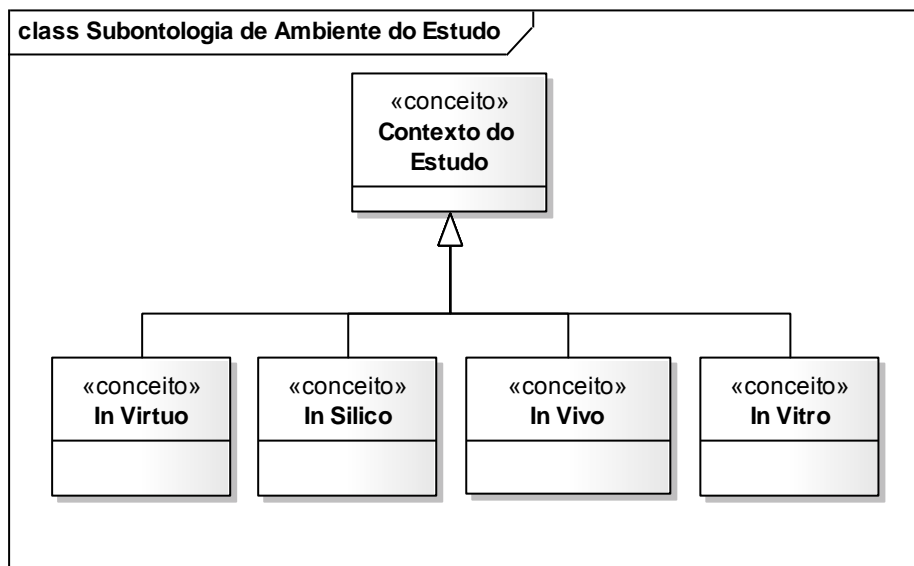


Figura III-13. Subontologia de Ambiente de Estudo

1.5.1. Subontologia de Estudo *In Vivo*

Estudos ***in vivo*** são aqueles que envolvem as pessoas em seus próprios ambientes de trabalho. São, portanto, *executados em ambiente de indústria*. Tais estudos experimentais ocorrem durante o processo de desenvolvimento em circunstâncias reais. A figura I-14 ilustra a subontologia.

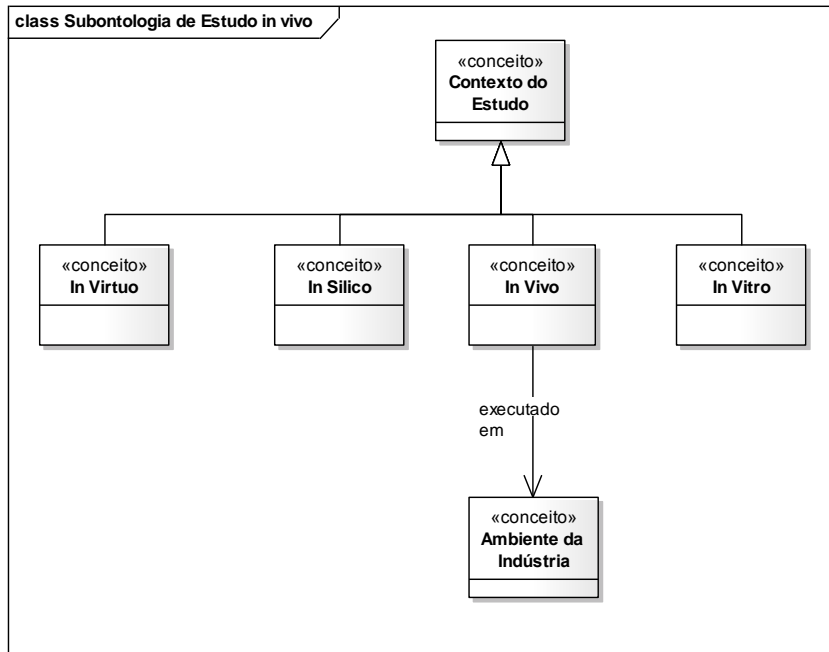


Figura III-14. Subontologia de estudo in vivo

1.5.2. Subontologia de Estudo *In Vitro*

Estudos *in vitro* são aqueles executados e controlados em ambientes tais como laboratórios ou comunidades controladas. Nestes estudos, pesquisadores criam situações próximas às reais em ambientes de laboratório para observar como os participantes executam uma determinada tarefa (TRAVASSOS e BARROS, 2003). A figura I-15 ilustra a subontologia.

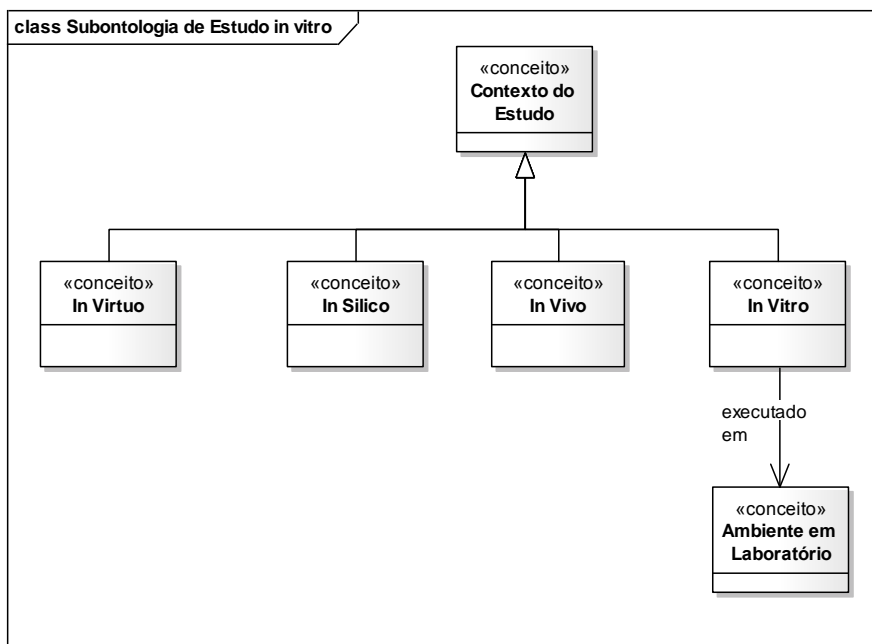


Figura III-15. Subontologia de estudo in vitro

1.5.3. Subontologia de Estudo *In Virtuo*

Estudos *in virtuo* são conduzidos em um ambiente virtual, composto por modelos numéricos que são manipulados por pessoas. O comportamento do modelo é analisado através de simulação. Em estudos experimentais *in virtuo*, um modelo computacional representa o objeto em estudo e, em alguns casos, o ambiente onde o estudo experimental é realizado. As pessoas que interpretam os resultados obtidos pela simulação do modelo podem manipulá-lo através de um programa de computador. A figura I-16 ilustra a subontologia.

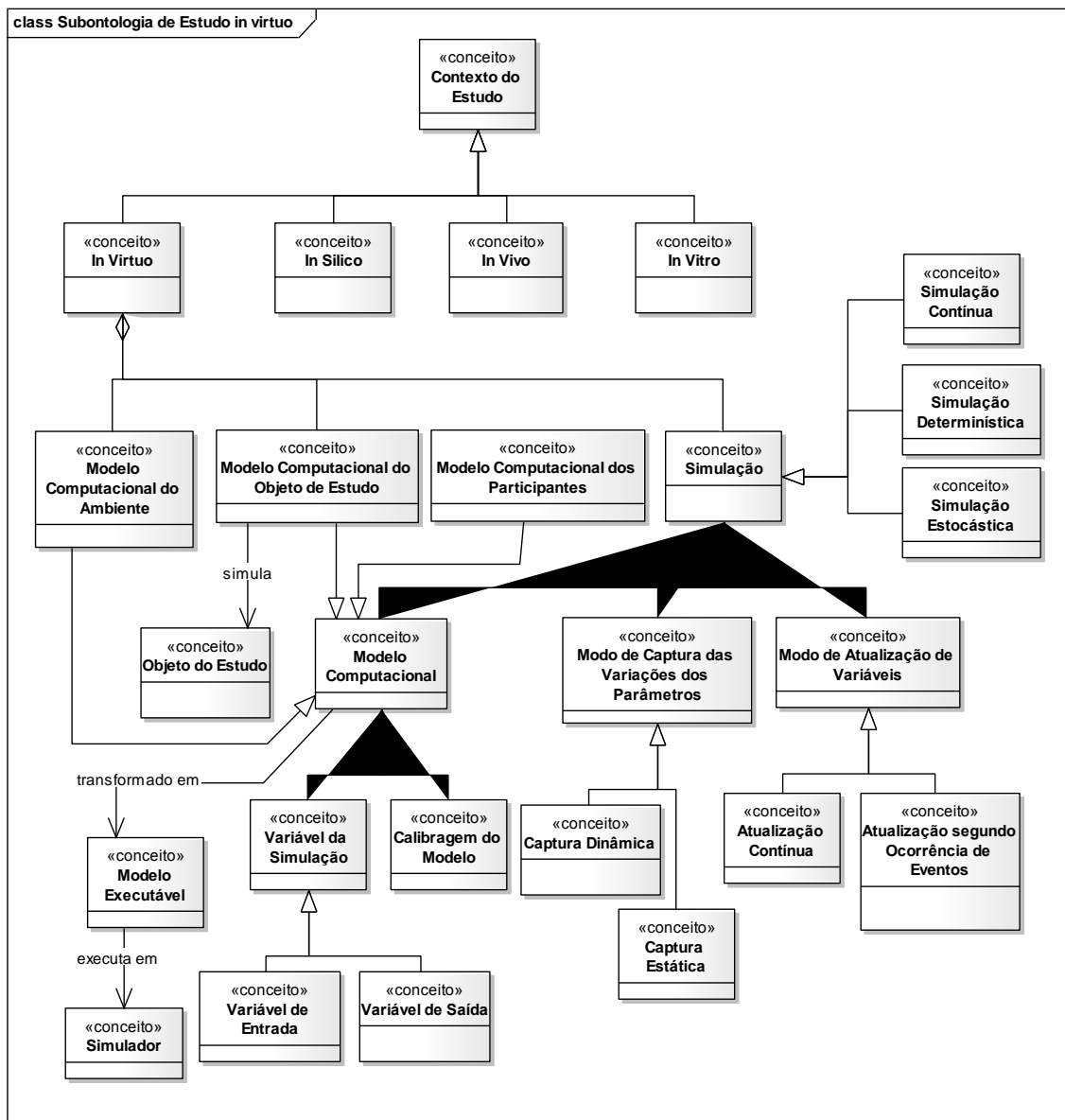


Figura III-16. Subontologia de Estudo *in virtuo*

Estudos **in virtuo**, por definição, simulam computacionalmente o ambiente e o **objeto do estudo**. Logo, tornam-se necessários **modelos computacionais** para descrevê-los:

- **Modelo computacional do ambiente:** Descreve computacionalmente as características do ambiente, assim como os eventos que ocorrem neste cenário.
- **Modelo computacional do objeto de estudo:** Modelo computacional que descreve as características do objeto de estudo que será simulado.

Em **modelos computacionais** há as **variáveis da simulação**, que podem ser de **entrada** (para as quais é estipulado uma faixa de valores de modo a serem manipuladas) e de **saída**, cujos valores são observados pelo pesquisador e possuem um significado. **Modelos computacionais** passam por procedimentos de ajustes para compatibilizar os dados de saída com aqueles do ambiente real. Estes procedimentos constituem a **calibração do modelo**. O **modelo computacional** é transformado em um **modelo executável**, que é um modelo codificado para ser *executado* em um **simulador** específico.

Na simulação de **modelos computacionais**, as variações dos parâmetros podem ser capturadas de forma **dinâmica** (intervalos de tempo previamente especificados) ou **estática** (instante específico de tempo). As variáveis podem ainda ser atualizadas de forma **contínua** ou segundo a **ocorrência de eventos**. Na **atualização dinâmica**, os valores são atualizados em intervalos de tempos fixos e segundo equações pré-definidas.

São tipos de simulação (MÜLLER e PFAHL, 2008):

- **Simulação Estocástica:** O resultado da simulação pode variar ainda que um mesmo conjunto de dados de entrada seja fornecido. Neste cenário ocorre influência de fatores probabilísticos;
- **Simulação Determinística:** O resultado da simulação será sempre igual a partir de um mesmo conjunto de dados de entrada. Não há influência de fatores probabilísticos;
- **Simulação Contínua:** É um tipo de simulação onde o modelo de simulação atualiza os valores das variáveis do modelo baseados em um conjunto fixo de equações bem definidas em intervalos de tempos equidistantes.

1.5.4. Subontologia de Estudo *In Silico*

Estudos *in silico* são aqueles em que, além do objeto e do ambiente a serem estudados, o comportamento dos indivíduos envolvidos também é descrito com modelos computacionais (TRAVASSOS e BARROS, 2003). A diferença fundamental entre os estudos *in virtuo* e *in silico* está no fato de que este trabalha também com a simulação do comportamento do participante, razão pela qual demanda também um modelo que descreva computacionalmente este comportamento. A figura I-17 ilustra a subontologia.

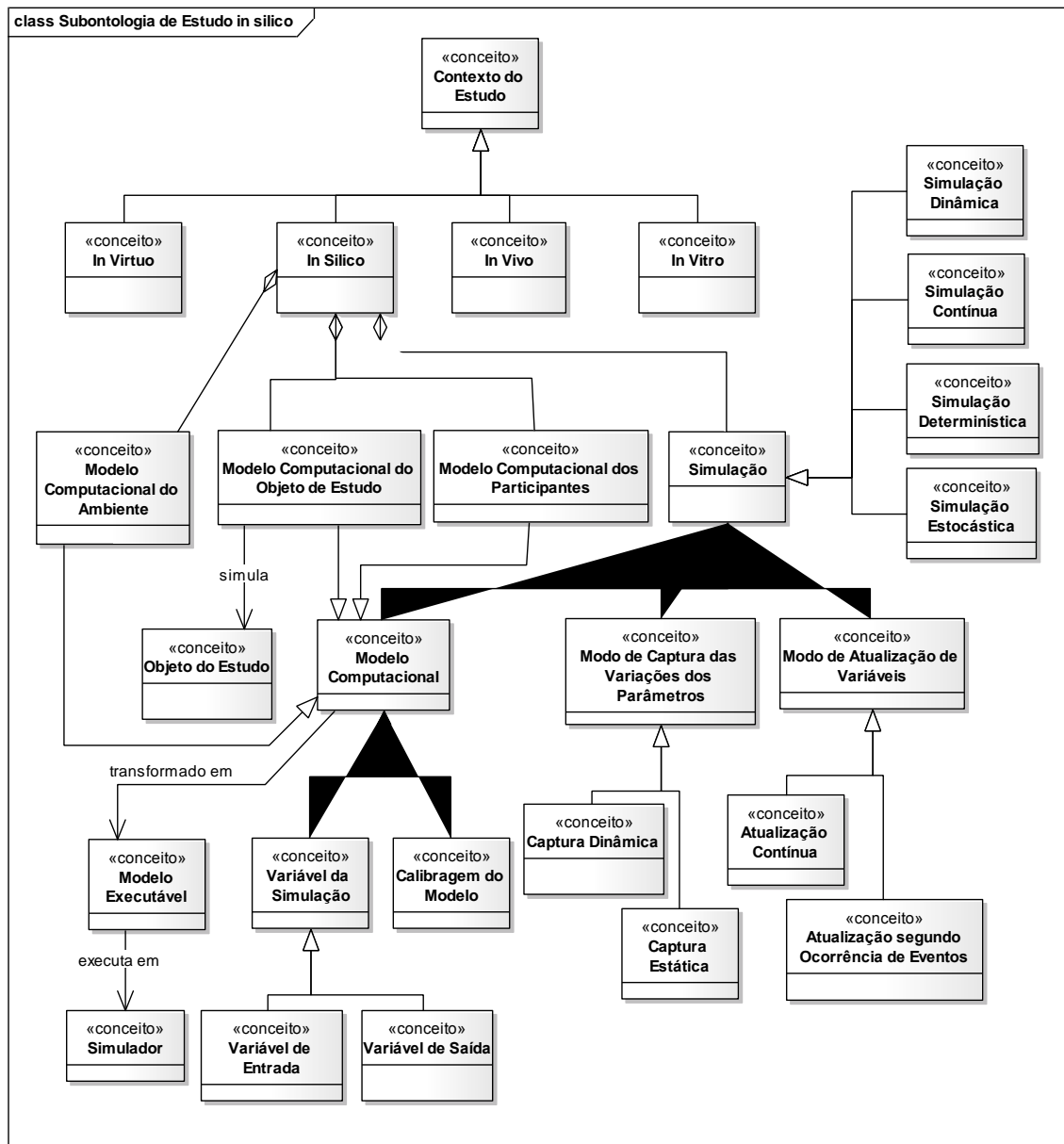


Figura III-17. Subontologia de estudo *in silico*

1.1. Referências bibliográficas

- BANDEIRA-DE-MELLO, R., (2006), “Softwares em Pesquisa Qualitativa”, em: *Pesquisa Qualitativa em Estudos Organizacionais: Paradigmas, Estratégias e Métodos*, Capítulo 15, São Paulo, Saraiva.
- BANDEIRA-DE-MELLO, R., CUNHA, C., (2006), “Grounded Theory”, em: *Pesquisa Qualitativa em Estudos Organizacionais: Paradigmas, Estratégias e Métodos*, , Capítulo 8, São Paulo, Saraiva.
- BASKEVILLE, R. L., (1999), “Investigatin Information Systems with Action Research”, *Communications of the Association for Information Systems*, Volume 2, Artigo 19, Outubro.
- BIANCHI, E. M. P. G., IKEDA, A. A., (2006), “Analisando a Grounded Theory em Administração”, IX SEMEAD - Seminários em Administração. São Paulo, Brazil.
- CAMPBELL, D.T., COOK, T.D., (1979), *Quasi-Experimentation, Design and Analysis Issues for Field Settings*, Houghton Mifflin Company.
- EASTERBROOK, S.; SINGER, J.; STOREY, M.; DAMIAN, D., (2008) “Selecting Empirical Methods for Software Engineering Research” in *Guide to Advanced Empirical Software Engineering*, Springer. pp. 285-310.
- GLASER, B., (1992), “Basics of grounded theory analysis”. Mill Valley CA, The Sociology Press.
- GÜNTHER, H., (1999), “Como elaborar um questionário”, em: *Instrumentos psicológicos: manual prático de elaboração*. 1ª edição, Brasília: LabPAM/IBAPP, p. 231-258.
- JUDD C. M., SMITH E. R., KIDDER L. H., (1991), “Research Methods in Social Relations”, sixth edition, Harcourt Brace Jovanovich, Fort Worth.
- JURISTO, N., MORENO, A. (2001), “Basics of Software Engineering Experimentation”, Kluwer Academic Press, 1ª edição.
- KITCHENHAM, B., PFLEEGER, S. L., (2008), “Personal Opinion Surveys”, em: *Guide to Empirical Software Engineering*, Springer London, pp 63-92.
- MELIN, U., AXELSSON, K., (2007), “Action in Action Research – Illustrations of What, Who, Why, Where, and When from an E-Government Project”, *Lecture Notes in Computer Science*, Volume 4656. Springer Berlin/Heidelberg.

- MAXWELL, K. D., (2002), “Applied Statistics for Software Managers”, Prentice Hall PTR.
- MENDONÇA, C.C., (2004), “Uma Infra-estrutura para apoio ao Planejamento e Execução de Pesquisas de Opinião na Web”, Tese de Mestrado, Programa de Engenharia de Sistemas e Computação -PESC/COPPE, Universidade Federal do Rio de Janeiro -UFRJ, Rio de Janeiro, RJ, Brasil.
- MÜLLER, M., PFAHL, D., (2008), “Simulation Methods”, Guide to Empirical Software Engineering, Springer London, pp 117-152.
- PERRY, D.E.; PORTER, A.A., VOTTA, L.G., (2000), “Empirical Studies of Software Engineering: A Roadmap. The Future of Software Engineering”. 22nd International Conference on Software Engineering, Limerick, Ireland, April.
- ROBINSON, H., SEGAL, J. e SHARP, H., (2007), “Ethnographically-informed empirical studies of software practice”, Information and Software Technology, 49(6), 540-551.
- RUNESON, P., (2007), “Introduction to Case Study Research”, IASESE (*International Advanced School of Empirical Software Engineering*), Madri, Setembro.
- SEAMAN, C. B., (2008), “Qualitative Methods”, Guide to Empirical Software Engineering, Springer London, pp 35-62.
- STRAUSS, A., CORBIN, J., (1998), “Basics of Qualitative Research: Techniques and Procedures for Developing Grounded Theory”, 2ª edição, Londres, SAGE Publications.
- TAYLOR, S.J., BOGDAN, R., (1998), “Introduction to Qualitative Research Methods. Wiley”, ISBN 0471168688.
- TRAVASSOS, G.H., BARROS, M.O., (2003), “Contributions of In Virtuo and In Silico Experiments for the Future of Empirical Studies in Software Engineering”. Proceedings of the WSESE03, Fraunhofer IRB Verlag, Roma.
- TRIPP, D. (2005), “Action research: a methodological introduction”, Educação e Pesquisa, volume 31, número 3, São Paulo, DOI: 10.1590/S1517-97022005000300009
- WÖHLIN, C., RUNESON, P., HÖST, M., OHLSSON, M., REGNELL, B., WESSLÉN A., (2000), “Experimentation in Software Engineering – An Introduction”, Kluwer Academic Publishers.

Anexo II

Processo de Construção de Ontologias - *Methodology*

Este anexo descreve o processo Methodology, adotado na construção da ontologia de Experimentação em Engenharia de Software

1.1. Conceitos Básicos

Desenvolvido no laboratório de Inteligência Artificial do Politécnico de Madri, o *Methodology* é um framework que apóia a construção de ontologias de forma sistemática e organização. Baseia-se no processo-padrão IEEE para desenvolvimento de software e, portanto, especifica um conjunto de atividades com propósitos, produtos e insumos específicos (FERNANDÉZ *et al.*, 1997). Contempla atividades de três tipos:

- **Atividades de gerenciamento de ontologias:** planejamento da construção em termos de custo, tempo alocado a cada atividade, definição de participantes e entre outras questões.
- **Atividades de desenvolvimento de ontologias:** especificação, conceitualização, formalização, integração, implementação e manutenção.
- **Atividades de suporte:** avaliação, documentação, gestão de configuração e entre outras.

1.2. Processo de Construção de Ontologias Methodology

1.2.1. Planejamento

Nesta atividade, são estimados os custos envolvidos no processo e o tempo previsto para cada atividade. Ferramentas necessárias são identificadas e disponibilizadas aos executores. Além disso, as atividades de suporte têm sua execução planejada dentro do processo, uma vez que podem ser realizadas de forma flexível (FERNANDÉZ *et al.*, 1997).

1.2.2. Especificação

A etapa de especificação envolve definir a razão para a construção e a forma de utilização da ontologia, além dos possíveis usuários.

1.2.3. Conceitualização

Nesta etapa são identificados os termos da ontologia. Diversas técnicas podem ser empregadas, inclusive aquelas comumente empregadas em desenvolvimento de software ligadas à elicitação de requisitos. Modelos semi-formais podem ser construídos para expressar o relacionamento entre estes conceitos. Estes modelos são preferíveis no sentido de que são mais formais que uma lista de termos em linguagem natural.

1.2.4. Formalização

Na etapa de formalização o modelo semi-formal da etapa de conceitualização é gerado utilizando uma linguagem formal de representação de ontologias.

1.2.5. Integração

Realiza-se integração da ontologia sendo construída com outras de domínio aderentes àquele sendo considerado.

1.2.6. Implementação

Na etapa de implementação, a ontologia é implementada utilizando linguagens como, por exemplo, OIL, DAML+OIL e OWL. A ontologia implementada permite que seja processada automaticamente por ferramentas inteligentes (ABECKER *et al.*, 1999).

1.2.7. Documentação

A etapa de documentação, sendo uma atividade de suporte, deve ter sua execução planejada de forma a atender a demanda de cada atividade de desenvolvimento. O andamento do processo é continuamente documentado, sendo registradas informações sobre os artefatos consumidos e produzidos em cada atividade.

1.2.8. Avaliação

Na etapa de avaliação foram conduzidas sessões de inspeção do glossário e das ontologias, visando identificar eventuais defeitos. O planejamento e a execução da avaliação são discutidos na subseção 4.5.2.

1.2. Referências bibliográficas

FERNANDEZ, M., GOMEZ-PEREZ A., JURISTO, N., (1997), “Methontology: From ontological art towards ontological engineering”, *Proceedings of the AAAI97 Spring Symposium Series on Artificial Intelligence*.

Anexo III

Modelo de Pacote de Estudos Experimentais do eSEE

Este anexo descreve resumidamente o modelo de pacote de estudos experimentais do eSEE, que especifica o conjunto de documentos consumidos e produzidos durante a execução do Processo de Experimentação

1.1. Modelo de Pacote de Estudos Experimentais

Em AMARAL (2003) são definidos um pacote de laboratório (ou Pacote de Experimento) e um Processo de Experimentação evoluído a partir do processo proposto em (WÖHLIN *et al.*, 2000). O processo de AMARAL (2003) está fortemente focado em guiar o pesquisador no registro das informações e artefatos, ou seja, no empacotamento de um estudo experimental. Para isto, o pacote de experimento definido segue um modelo de pacote, composto por uma taxonomia de documentos. Cada documento segue um modelo específico. Todos os modelos de documentos reunidos constituem a estrutura de uma base de conhecimento, na qual o conhecimento produzido ao longo do processo de experimentação é registrado em documentos específicos. Cada documento segue um modelo pré-definido dentro do modelo de pacote do experimento.

Cada modelo de documento especifica as informações que devem ser armazenadas ou empacotadas. As informações estão relacionadas aos resultados do processo de experimentação, e vão desde as informações iniciais da definição do experimento, tais como: objeto de estudo, propósito, foco, perspectiva e em que contexto o estudo vai ser realizado; passando por informações de planejamento e execução, tais como: população a ser estudada, recurso empregado, hipóteses que visa verificar (ou não), variáveis utilizadas, dentre outros; até chegar às informações de coleta e análise dos resultados, tais como os dados crus obtidos, os dados analisados, resultados e comentários da análise e questões provenientes do estudo realizado.

Os tipos de documentos que devem ser armazenados durante a execução de um experimento, as pessoas envolvidas na execução e os papéis que elas desempenham, suas responsabilidades, relacionamentos e artefatos produzidos na execução do experimento foram descritos por AMARAL (2003) em modelos UML (*Unified*

Modeling Language). Todos esses modelos constituem o que denomina modelo de pacote do experimento.

Para representar as diferentes abstrações conceituais, AMARAL (2003) explorou a utilização de um diagrama de pacotes UML. Um pacote, neste contexto, agrega as informações relacionadas a conceitos de mesmo nível de abstração, podendo assumir a representação de um conjunto de documentos/informações relacionados a um experimento ou de um conjunto de participantes que necessita ter acesso a estes documentos. Desta forma, este modelo (Figura II-) foi dividido em três conjuntos de documentos, representado por: Organização do Experimento, Instrumentos, Formulários de Resultados, e em um conjunto de participantes representado pela Comunidade de Engenharia de Software. A seguir as subseções apresentam os indivíduos que compõem esta comunidade e os modelos de documentos constituintes dos pacotes de organização do experimento, instrumentos e formulário de resultados.

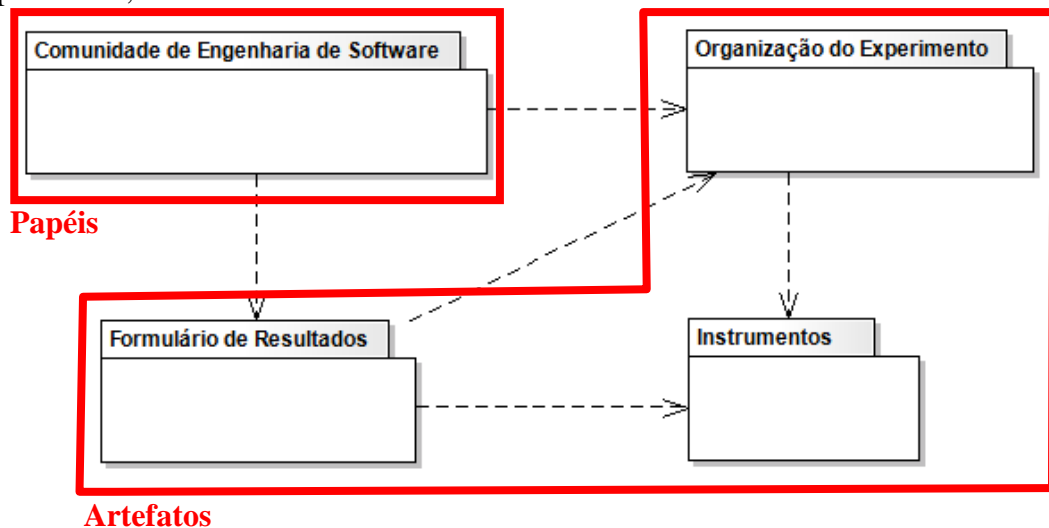


Figura II-1. Modelo de Pacote do Experimento

1.1.1 Comunidade de Engenharia de Software

O objetivo deste conjunto é representar os diferentes participantes existentes na Comunidade de Engenharia de Software e descrever os papéis que podem vir a assumir dependendo das atividades desempenhadas na execução de um experimento, Isto influencia, por exemplo, as permissões de acesso de cada participante a um determinado documento. A Tabela sintetiza as pessoas e papéis especificados no pacote de comunidade de Engenharia de Software.

Tabela II-1. Pessoas e Papéis da Comunidade de Engenharia de Software

Pessoas/papéis	Descrição
----------------	-----------

Engenheiro de Software	Representa os Engenheiros de Software participantes da comunidade, podendo ser pesquisadores.
Profissional da Prática	Representa uma pessoa interessada em saber que tipos de tecnologias foram validadas experimentalmente. Esta pessoa não está interessada em repetir um experimento, mas precisa ter livre acesso a seus resultados, avaliando se estes resultados preenchem os requisitos do seu ambiente de desenvolvimento.
Repetidor	Responsável por repetir um experimento.
Pesquisador	Representa os Engenheiros de Software que desenvolvem pesquisa na área.
Agregador	Combina os resultados de uma família de experimentos similares ou repetidos.
Projetista do Experimento	Projeta e executa um novo experimento.
Visitante	Descreve os participantes interessados em Engenharia de Software que possam estar navegando pela Internet/Intranet e lendo parte do conteúdo dos pacotes.
Leitor Casual	Aprende ou se atualiza em técnicas de experimentação ou em certo experimento.
Executores do Experimento	Representa os participantes que executam os experimentos. Eles podem ser de três tipos: Profissional, Estudante ou Desenvolvedor da Indústria.
Profissional Pesquisador	Refere-se aos profissionais que participam de um experimento.
Estudante	Descreve os estudantes que podem participar de um determinado experimento.
Desenvolvedor da Indústria	Descreve os profissionais desenvolvedores da área industrial que podem participar de um determinado experimento.
Indivíduo	Descreve os participantes que participam de um experimento planejado.

1.1.2. Documentos de Organização do Experimento

O conjunto Organização do Experimento é uma generalização dos documentos: plano do experimento, descrição dos exercícios, notas do curso de treinamento, procedimento do experimento e descrição dos parceiros. Cada um destes documentos forma o documento organização do experimento. A finalidade de cada um destes documentos é descrita em detalhes na Tabela II-1.

Este conjunto de documentos agrega informações necessárias à caracterização e planejamento de um experimento. A importância deste pacote se deve ao fato que com estas informações um pesquisador pode saber que questionários devem ser aplicados, a

ordem de sua aplicação, que informações estão definidas no projeto experimental, quais são os parceiros envolvidos na pesquisa, dentre outros.

Tabela II-1. Conjunto de Documentos de Organização do Experimento

Documento	Descrição
Documento do Experimento	Representa os diferentes tipos de documentos utilizados no experimento. Pode ser de um dos seguintes: Organização do Experimento, Formulários de Resultado ou Instrumentos.
Pacote do Experimento	Representa o pacote contendo as informações e documentos relativos ao experimento ou ao conjunto de experimentos.
Descrição da Área Temática	Descreve informações sobre temas gerais e sobre áreas técnicas específicas.
Organização do Experimento	Representa os conceitos relacionados à descrição dos documentos de um experimento.
Plano do Experimento	Descreve informações sobre o estudo experimental, como participantes, objetos, diretrizes, hipóteses, variáveis dependentes e independentes, questões e métricas, questões não respondidas, questões em aberto e caminhos de validação, dentre outros.
Descrição dos Exercícios	Descreve informações sobre os exercícios usados através do experimento.
Notas do Curso de Treinamento	Descrevem conceitos, termos relevantes e informações de notas de aulas, cursos e treinamentos.
Procedimento do Experimento	Descreve os procedimentos e ações que devem ser realizados para execução de um experimento.
Descrição dos Parceiros	Descreve informações sobre os parceiros que participam na realização de um experimento.

Amaral (2003) definiu para os documentos de cada pacote quais são as informações neles registradas. A definição completa dessas informações para um dado documento constitui o seu respectivo modelo. A título de exemplo, apresentamos na tabela 2-3 as informações que constituem o modelo do documento de Plano do Experimento, que possui no trabalho de Amaral (2003) o maior volume de informações registradas se comparado com os demais documentos do pacote. Cada informação presente no documento também pode ser interpretada como sendo uma seção dentro deste documento. Vale ressaltar que, embora (AMARAL, 2003) explicita as informações de cada modelo, não foi alvo deste trabalho defini-las.

Tabela II-2. Informações registradas no documento de Plano do Experimento

Informações registradas no documento de Plano do Experimento

introdução
trabalhoAnterior
objetivoGeral
objetivoEspecífico
hipóteseNula
hipóteseAlternativa
hipótesesVariáveis
hipótesesCritérioDivisão
variávelIndependente
escalaMedidaVariávelIndependente
faixaVariávelIndependente
níveisEspecíficosVariávelIndependente
variávelDependente
escalaMedidaVariávelDependente
faixaVariávelDependente
Participantes
Grupos
experiênciasParticipantes
variaçãoParticipantes
variaçãoEventosAções
técnicasProbabilísticasAmostragem
técnicasNãoProbabilísticasAmostragem
Objetos
Medições
Diretrizes
recursoSoftware
recursoHardware
questõesMétricas
questõesNãoPodemSerRespondidas
questõesEmAberto
oQueDizerEOQueNãoDizerAosParticipantes
diretrizesTratamentos
participantesGruposTratamentosNosObjetos
validadeInterna
validadeExterna
validadeConclusão
validadeConstrução

1.1.3. Documentos de Instrumentos

Os instrumentos de um estudo experimental descrevem os meios de executá-lo e monitorá-lo, sem afetar o controle do experimento. O conjunto de documentos de instrumento é formado pelos documentos artefato do experimento e pelo artefato de software.

O artefato do experimento apresenta os documentos que guardam informações sobre formatos da lista de defeito, formato do questionário e formato da nota de

observação. Apresenta ainda documentos referentes à medição e às diretrizes utilizados em um experimento.

O documento artefato de software é composto pelos documentos componente, código e documentação. O documento documentação refere-se aos documentos do usuário, documentos de requisitos, documentos da análise, documentos do projeto, documentação do sistema, modelo do domínio do negócio, modelo do processo do software e caso de teste de dados. O documento Código é composto pelos documentos texto do programa, função e interface. A tabela 2-4 relaciona todos os documentos de instrumentos.

Tabela II-3. Documentos de Instrumentos

Documentos	Descrição
Instrumentos	Descreve os instrumentos usados em um experimento. Podendo ser de dois tipos: Artefato de Software ou Artefato do Experimento
Artefato do Experimento	Descreve os artefatos do experimento usados em um experimento. Podendo ser de três tipos: Formato, Medição ou Diretrizes.
Formato	Descreve os formatos usados em um experimento. Podendo ser de três tipos: Formato da Lista de Defeito, Formato da Nota de Observação ou Formato do Questionário.
Formato da Lista de Defeito	É um padrão utilizado para descrever os defeitos encontrados em um experimento.
Formato da Nota de Observação	É um padrão utilizado para descrever as observações relativas a um experimento.
Formato do Questionário	É um padrão utilizado para descrever as questões de um experimento.
Medição	Descreve as métricas usadas para validar os dados coletados de forma manual ou através de entrevistas.
Diretrizes	Descreve padrões de processos, <i>checklists</i> e métodos são usados em um experimento.
Artefato de Software	Descreve os artefatos de software utilizados em um experimento.
Descrição do Domínio	Descreve o domínio da aplicação para o qual os artefatos foram construídos.
Componente	Descreve informações sobre componentes como bibliotecas e componentes de prateleira, usados no experimento.
Código	Descreve informações sobre os programas, funções, e interfaces usados no experimento.
Texto do programa	Descreve os programas utilizados em um artefato de software para produzir um experimento.

Função	Descreve as funções utilizadas em um artefato de software para produzir um experimento.
Interface	Descreve as interfaces utilizadas em um artefato de software para produzir um experimento.
Documentação	Descreve informações sobre os artefatos não executáveis.
Documentos da Análise	Descreve informações de Análise como os Casos de Uso da UML e comentários.
Documentos do Projeto	Descreve informações de Projeto como Diagramas de Classe da UML.
Modelo do Domínio do Negócio	Descreve informações sobre o Modelo de Negócios.
Modelo do Processo de Software	Descreve como o processo de software foi modelado.
Documentos dos Requisitos	Descreve as características que o sistema deve ter em acordo com os requisitos do usuário. Representa as características funcionais e não-funcionais.
Documentação do Sistema	Responsável pelas informações sobre o sistema que servem de apoio para os pesquisadores.
Documentação do Usuário	Responsável pelas informações sobre o sistema que servem de apoio para os usuários.
Casos de Teste dos Dados	Descreve os casos de teste definidos para o Sistema.

1.1.4. Documentos de Resultados

Este conjunto descreve os resultados na forma de dados e relatórios, obtidos a partir da execução de um experimento. Estes documentos viabilizam que os resultados e conclusões de um experimento sejam devidamente armazenados. O documento formulários de resultados é formado pelos documentos dados e relatório.

O documento dados é formado pelos documentos resultados agregados, resultados refinados do experimento e resultados crus do experimento. As informações do documento resultados crus do experimento são utilizadas pelo documento resultados refinados do experimento, que por sua vez fornece informações para o documento resultados agregados. O documento resultados crus do experimento é formado pelos documentos lista de defeitos, notas de observação e questionários.

O documento relatório é formado pelas informações dos documentos lembretes, lições aprendidas, questões que não podem ser respondidas, relatório final e questões em aberto. A tabela 2-5 relaciona os documentos presentes no pacote de documentos de resultados.

Tabela II-4. Documentos de Resultados

Documentos	Descrição
Formulário de Resultado	Descreve informações sobre os documentos resultantes de um experimento.
Dados	Descreve informações sobre os dados resultantes de um experimento.
Resultados Agregados	Define os objetivos e hipóteses para combinar os resultados de uma família de experimentos repetidos ou similares.
Resultados Refinados do Experimento	Descreve os dados analisados provenientes dos resultados crus (sem análise) e que são sintetizados em Resultados Agregados.
Resultados Crus do Experimento	São os resultados crus (sem análise), que podem estar armazenados em uma planilha ou em um banco de dados.
Lista de Defeitos	Descreve os defeitos encontrados no experimento, através de uma lista.
Notas de Observação	Descreve observações sobre o experimento.
Questionários	Descreve as questões aplicadas através do experimento.
Relatório	Descreve informações sobre os resultados de um experimento através de relatórios.
Lembretes	Descreve as informações frequentemente perguntadas sobre o experimento.
Lições aprendidas	Descreve as lições aprendidas sobre os resultados do experimento.
Relatório Final	Explica o método utilizado no experimento, bem como a técnica utilizada e o que pode ser melhorado.
Questões em aberto	Descreve as perguntas que representam futuras perspectivas de pesquisa.
Questões que não podem ser respondidas	Descreve as questões que não se consegue responder apenas realizando o experimento, mas que seria bastante interessante poder respondê-las, como por exemplo: não podemos ter o caso ideal, é impraticável testar isto, dentre outras.

1.2 Referências bibliográficas

AMARAL, E.A.G.G., 2003, *Empacotamento de Experimentos em Engenharia de Software*, Dissertação de Mestrado, Programa de Engenharia de Sistemas e Computação, COPPE/UFRJ, Universidade Federal do Rio de Janeiro. Rio de Janeiro, RJ, Brasil.

CHAPETTA, W. A., 2006, *Uma Infra-estrutura para Planejamento, Execução e Empacotamento de Estudos Experimentais em Engenharia de Software*, Dissertação de Mestrado, Programa de Engenharia de Sistemas e Computação, COPPE/UFRJ, Universidade Federal do Rio de Janeiro. Rio de Janeiro, RJ, Brasil.