



Fernando de Freitas Silva

**Uma nova abordagem de mineração de repositórios de
software utilizando ferramentas da Web Semântica**

Dissertação de Mestrado

Dissertação apresentada como requisito parcial para
obtenção do grau de Mestre pelo Programa de Pós-
graduação de Informática da PUC-Rio.

Orientador: Prof. Daniel Schwabe

Rio de Janeiro
Agosto de 2013



Fernando de Freitas Silva

**Uma nova abordagem de mineração de repositórios de
software utilizando ferramentas da Web Semântica**

Dissertação apresentada como requisito parcial para
obtenção do título de Mestre pelo Programa de Pós-
Graduação em Informática da PUC-Rio. Aprovada pela
Comissão Examinadora abaixo assinada.

Prof. Daniel Schwabe

Orientador

Departamento de Informática - PUC-Rio

Prof. Arndt Von Staa

Departamento de Informática - PUC-Rio

Prof. Alessandro Garcia

Departamento de Informática - PUC-Rio

Prof. José Eugenio Leal

Coordenador(a) Setorial do Centro Técnico Científico - PUC-Rio

Rio de Janeiro, 15 de agosto de 2013

Todos os direitos reservados. É proibida a reprodução total ou parcial do trabalho sem autorização da universidade, do autor e do orientador.

Fernando de Freitas Silva

Graduou-se em Bacharel em Sistemas de Informação pela Pontifícia Universidade Católica do Rio de Janeiro em 2009.

Ficha Catalográfica

Silva, Fernando de Freitas

Uma nova abordagem de mineração de repositórios de software utilizando ferramentas da Web semântica / Fernando de Freitas Silva ; orientador: Daniel Schwabe. – 2013.

178f : il. ; 30 cm

Dissertação (mestrado)–Pontifícia Universidade Católica do Rio de Janeiro, Departamento de Informática, 2013.

Inclui bibliografia

Informática – Teses. 2. Web semântica. 3. Manutenção de software. 4. Engenharia de software. 5. Semântica. 6. Repositórios de software. I. Schwabe, Daniel. II. Pontifícia Universidade Católica do Rio de Janeiro. Departamento de Informática. III. Título.

CDD: 004

Agradecimentos

Agradeço aos meus pais, Walquíria de Freitas Silva e Fernando Jorge Lopes Silva, por sempre me apoiarem, incentivarem e por terem me dado uma criação excepcional. Sem vocês eu não estaria onde estou hoje. Muito obrigado!

Agradeço a minha esposa e melhor amiga, Geyzyely, pelo apoio durante esta empreitada e por ter compreendido as madrugadas que tive que passar longe de você para estudar. Obrigado por estar ao meu lado e espero que eu possa te apoiar do mesmo modo que você me apoiou. Te amo.

Agradeço ao meu “amigão” e filho, Joaquim, que apesar de sua pouca idade entendia os momentos que eu não podia brincar ou estar com ele. Obrigado por ser um filho maravilhoso e que você alcance coisas maiores do que eu já imaginei.

Tenho muito a agradecer ao meu orientador Professor Daniel Schwabe pelo constante incentivo e apoio durante Mestrado. Sem suas observações e direcionamentos eu não teria chegado até aqui. Muito obrigado!

Agradeço ao Gustavo Robichez de Carvalho e ao Professor Carlos José Pereira de Lucena pelo apoio durante minha graduação e pela oportunidade de ingressar no Mestrado. Muito obrigado!

Agradeço aos meus colegas da PrimeUp, ao sócio Leandro Daflon e ao meu gerente João Manoel Silvestre pelo compreensão nos momentos que tive que me ausentar do trabalho durante meu Mestrado. Muito obrigado!

Agradeço à PUC-Rio que me formou Bacharel em Sistemas de Informação e me deu a oportunidade de me tornar Mestre em Informática sem custo algum.

Finalmente, gostaria de agradecer à CAPES pela bolsa concedida durante os dois anos do Mestrado. Este apoio foi imprescindível para que eu pudesse concluir meus estudos. Obrigado!

Resumo

Silva, Fernando de Freitas; Schwabe, Daniel. **Uma nova abordagem de mineração de repositórios de software utilizando ferramentas da Web Semântica**. Rio de Janeiro, 2013. 178p. Dissertação de Mestrado - Departamento de Informática, Pontifícia Universidade Católica do Rio de Janeiro.

A Mineração de Repositórios de Software é um campo de pesquisa que extrai e analisa informações disponíveis em repositórios de software, como sistemas de controle de versão e gerenciadores de issues. Atualmente, diversos trabalhos nesta área de pesquisa têm utilizado as ferramentas da Web Semântica durante o processo de extração a fim de superar algumas limitações que as abordagens tradicionais enfrentam. O objetivo deste trabalho é estender estas abordagens que utilizam a Web Semântica para minerar informações não consideradas atualmente. Uma destas informações é o relacionamento existente entre as revisões do controle de versão e as mudanças que ocorrem no Abstract Syntax Trees dos arquivos modificados por essas revisões. Adicionalmente, esta nova abordagem também permite modelar a interdependência entre os projetos de software, suas licenças e extrair informações dos builds gerados por ferramentas de integração contínua. A validação desta nova abordagem é demonstrada através de um conjunto de questões que são feitas por desenvolvedores e gerentes durante a execução de um projeto e que foram identificadas em vários trabalhos da literatura. Demonstramos como estas questões foram convertidas para consultas SPARQL e como este trabalho consegue responder às questões que não são respondidas ou são respondidas parcialmente em outras ferramentas.

Palavras-chave

Web Semântica; Manutenção de Software; Engenharia de Software Semântica; Repositórios de Software.

Abstract

Silva, Fernando de Freitas; Schwabe, Daniel (Advisor). **A New Approach for Mining Software Repositories using Semantic Web Tools**. Rio de Janeiro, 2013. 178p. MSc. Dissertation - Departamento de Informática, Pontifícia Universidade Católica do Rio de Janeiro.

The Mining of Software Repositories is a field of research that extracts and analyzes information available in software repositories, such as version control systems and issue trackers. Currently, several research works in this area have used Semantic Web tools during the extraction process to overcome some limitations that traditional approaches face. The objective of this work is to extend the existing approaches that use Semantic Web tools to mine information not considered in these works. The objective of this work is to extend these approaches using the Semantic Web to mine information not currently considered. One of these information is the relationship between revisions of version control and the changes that occur in the Abstract Syntax Trees of files modified by these revisions. Additionally, this new approach also allows modeling the interdependence of software projects, their licenses and extracting information from builds generated by continuous integration tools. The validation of this approach is demonstrated through a set of questions that are asked by developers and managers during the execution of a project and have been identified in various works in the literature. We show how these questions were translated into SPARQL queries and how this work can answer the questions that are not answered or are partially answered in other tools.

Keywords

Semantic Web; Software Maintenance; Semantic Software Engineering; Software Repositories.

Sumário

1	Introdução	18
1.1.	Limitações das Abordagens Atuais	21
1.2.	Objetivo	22
1.3.	Organização da Dissertação	24
2	Fundamentos	26
2.1.	Visão Geral das Ferramentas utilizadas no Ciclo de Vida de Desenvolvimento de Software	26
2.2.	Tecnologias da Web Semântica	29
2.2.1.	Resource Description Framework (RDF)	30
2.2.2.	RDF Schema (RDFS)	32
2.2.3.	SPARQL	34
2.2.4.	Web Ontology Language (OWL)	36
3	Ontologias e Vocabulários	39
3.1.	Description of a Project (DOAP)	39
3.2.	Friend of a Friend (FOAF)	40
3.3.	Dublin Core	42
3.4.	Creative Commons Rights Expression Language (CC Rel)	43
3.5.	Open Services for Lifecycle Collaboration (OSLC)	44
3.5.1.	OSLC Change Management Vocabulary (OSLC CM)	45
3.5.2.	OSLC Automation Management Vocabulary (OSLC Auto)	47
3.5.3.	OSLC Asset Management Vocabulary (OSLC Asset)	48
3.5.4.	OSLC Quality Management Vocabulary (OSLC QM)	50
3.5.5.	OSLC Software Configuration Management Vocabulary (OSLC SCM)	51
3.5.6.	As Novas Ontologias	52
4	A Plataforma de Extração	63
4.1.	Casos de Uso da Plataforma	63
4.1.1.	Cadastrar Requisição de Extração de Informações de Controle de Versão	64
4.1.2.	Cadastrar Requisição de Extração de Informações de Demandas e	

Defeitos	65
4.1.3. Cadastrar Requisição de Extração de Informações de Integração Contínua	65
4.1.4. Cadastrar Requisição de Extração de Informações de Versão de uma Biblioteca	65
4.1.5. Cadastrar Requisitar de Extração de Diferenças entre Versões de uma Biblioteca	65
4.1.6. Executar Consulta SPARQL	66
4.1.7. Pesquisar Requisições de Extração	66
4.1.8. Iniciar Requisição de Extração	66
4.2. Arquitetura da Plataforma	67
4.3. Arquitetura de Implementação	75
4.3.1. Tecnologias de Infraestrutura	75
4.3.2. Tecnologias para a persistência dos Dados RDF	83
4.3.3. Tecnologias de Apoio aos Conectores	84
5 Validação e Estudo de Caso	88
5.1. Introdução	88
5.2. Perguntas Frequentes	89
5.3. Exemplos de Novas Perguntas	94
5.4. Estudos de Caso	97
6 Conclusão e Trabalhos Futuros	104
6.1. Trabalhos Relacionados	104
6.2. Contribuições	105
6.3. Trabalhos Futuros	106
7 Referências Bibliográficas	108
Apêndice A – Módulos Conectores Comuns	111
Apêndice B – Módulos Transformadores RDF	135
Apêndice C – Módulos Auxiliares	153
Apêndice D – Módulos Coordenadores	156

Apêndice E – Consultas SPARQL “Perguntas Frequentes dos

Desenvolvedores”

171

Lista de figuras

Figura 1 Exemplo do Fluxo de Informações entre ferramentas utilizadas no desenvolvimento de software.....	27
Figura 2 Exemplos de Representação de uma mesma informação nas três modelagens	31
Figura 3 Exemplo da Notação RDF/XML representando as informações de pessoa Eric Miller.....	32
Figura 4 Exemplo de uma consulta SPARQL do tipo SELECT	35
Figura 5 Exemplo da representação das informações de um projeto utilizando DOAP	40
Figura 6 Exemplo da representação de informações utilizando a ontologia FOAF	41
Figura 7 Representação das informações de uma licença utilizando o vocabulário CC REL	43
Figura 8 Ilustração da sinergia de ferramentas aderentes aos padrões da OSLC.....	45
Figura 9 Representação das informações de um defeito utilizando o vocabulário OSLC CM	46
Figura 10 Representação do resultado de uma automação utilizando o vocabulário OSLC Automation.....	47
Figura 11 Exemplo de Representação das informações de um ativo utilizando o vocabulário OSLC Asset.....	49
Figura 12 Exemplo da representação de um Caso de Teste utilizando o vocabulário OSLC Quality Management	50
Figura 13 Exemplo de representação de uma classe chamada “org.example.Class1” na ontologia Evoont SOM e na ontologia EPR.....	53
Figura 14 Exemplo de representação do relacionamento entre um método e o tipo de seu retorno	54
Na tabela abaixo apresentamos os principais conceitos e propriedades que são definidas nesta ontologia. Já na Figura 15, demonstramos visualmente como estes principais conceitos se relacionam.....	54
Figura 15 Representação visual dos elementos da ontologia de Entidades, Propriedades e Relacionamentos	55

Figura 16 Representação da exclusão de um parâmetro de um método na ontologia EPR.....	56
Figura 17 Representação da relação de membro entre a pessoa “p1” e o comitê “c1”	57
Figura 18 Representação da nova entidade “membership” que foi criada para representar informações sobre o relacionamento da pessoa “p1” e o comitê “c1”	58
Figura 19 Representação do responsável pela nomeação da pessoa “p1” no comitê “c1”	58
Figura 20 Exemplo da Hierarquia de impactos.....	61
Figura 21 Representação da modificação da visibilidade de um método de “package” para “public”	62
Figura 22 Diagrama de Casos de Uso da Plataforma	64
Figura 23 Visão Geral dos Processos de Extração implementados na Plataforma de Extração	67
Figura 24 Visão Geral do Processo de Extração de Informações de Ferramentas de Controle de Versão	72
Figura 25 Visão Geral do Processo de Extração de Informações de uma Versão de uma Biblioteca	73
Figura 26 Visão Geral do Processo de Extração de Informações de Ferramentas de Integração Contínua	73
Figura 27 Visão Geral do Processo de Extração de Informações de Diferenças entre duas Versões de uma Biblioteca	74
Figura 28 Visão Geral do Processo de Extração de Informações de Ferramentas de Gerenciamento de Demandas/Defeitos	75
Figura 29 Arquitetura definida pela especificação OSGi	77
Figura 30 Interfaces que o módulo Conector de Ferramentas de Controle de Versão fornece	78
Figura 31 Exemplo de um MANIFEST.MF de um bundle.....	78
Figura 32 Exemplo da dinâmica entre dois bundles e um serviço fornecido por um deles	79
Figura 33 Código que registra novos serviços do tipo ISCMConnector.....	80
Figura 34 Código que recebe notificação da remoção de um serviço do tipo ISCMConnector e remove do SCMConnectorRegister.....	81
Figura 35 Trecho do Arquivo XML de Configuração do serviço fornecido pelo Módulo Conector Git	82

Figura 36 Parte dos Atores criados para o processo de Extração de Informações de Ferramentas de Controle de Versão.....	83
Figura 37 Visualização das Chamadas de um Método na IDE Eclipse	88
Figura 38 Visualização de todos os defeitos abertos e sem um responsável na ferramenta Atlassian Jira	88
Figura 39 Imagem do GitHub que demonstra o desenvolvedor Sean Owen como o desenvolvedor com maior número de commits no período de tempo....	99
Figura 40 Tabela com os métodos mais invocados no projeto.....	101
Figura 41 Imagem da Página do GitHub que indica os dois desenvolvedores retornados pela consulta como os principais contribuidores da classe	102
Figura 42 Interface ISCMConnector.....	111
Figura 43 A classe SCMConnectorRegister e a interface ISCMConnectorRegister	112
Figura 44 Diagrama de Estrutura Composta do Módulo Conector de Ferramenta de Controle de Versão.....	113
Figura 45 Diagrama de Sequência da integração entre o cliente o Módulo Conector de Ferramenta de Controle de Versão	113
Figura 46 Diagrama de Estrutura Composta do Módulo Conector Git.....	114
Figura 47 Interface ICIConnector	114
Figura 48 A classe CIConnectorRegister e a interface ICIConnectorRegister..	115
Figura 49 Diagrama de Estrutura Composta do Módulo Connector Comum de Ferramentas de Integração Contínua.....	116
Figura 50 Diagrama de Sequência da integração entre o cliente o Módulo Connector de Ferramentas de Integração Contínua.....	117
Figura 51 Diagrama de Estrutura Composta do Módulo Conector Jenkins	118
Figura 52 Interface IAssetConnector.....	118
Figura 53 A classe AssetConnectorRegister e a interface IAssetConnectorRegister.....	119
Figura 54 Diagrama de Estrutura Composta do Módulo Connector de Ferramentas de Gerenciamento de Artefatos	120
Figura 55 Diagrama de Sequência da integração entre o cliente o Módulo Connector de Ferramentas de Gerenciamento de Artefatos.....	120
Figura 56 Diagrama de Estrutura Composta do Módulo Conector Aether.....	121
Figura 57 Interface ICMConnector	121
Figura 58 A classe <i>CMConnectorRegister</i> e a interface	

<i>ICMConnectorRegister</i>	122
Figura 59 Diagrama de Estrutura Composta do Módulo Conector Comum de Ferramentas de Gerenciamento de Defeitos e Demandas.....	123
Figura 60 Diagrama de Sequência da integração entre o cliente o Módulo Conector de Ferramentas de Gerenciamento de Demandas e Defeitos	123
Figura 61 Diagrama de Estrutura Composta do Módulo Conector Jira	124
Figura 62 Interface IASTConnector.....	125
Figura 63 A classe <i>ASTConnectorRegister</i> e a interface <i>IASTConnectorRegister</i>	125
Figura 64 Diagrama de Estrutura Composta do Módulo Conector Comum de Linguagem de Programação	126
Figura 65 Diagrama de Sequência da integração entre o cliente o Módulo Conector de Linguagem de Programação	126
Figura 66 Diagrama de Estrutura Composta do Módulo Conector de Linguagem de Programação Java	127
Figura 67 Interface IDependencyConnector.....	128
Figura 68 A classe <i>DependencyConnectorRegister</i> e a interface <i>IDependencyConnectorRegister</i>	128
Figura 69 Diagrama de Estrutura Composta do Módulo Comum Conector de Ferramentas de Gerenciamento de Dependências	129
Figura 70 Diagrama de Sequência da integração entre o cliente o Módulo Conector de Ferramentas de Gerenciamento de Dependência.....	130
Figura 71 Diagrama de Estrutura Composta do Módulo Conector da Ferramenta Maven	131
Figura 72 Interface <i>ILicenseConnector</i>	131
Figura 73 A classe <i>LicenseConnectorRegister</i> e a interface <i>ILicenseConnectorRegister</i>	132
Figura 74 Diagrama de Estrutura Composta do Módulo Comum Connector de Licenças	133
Figura 75 Diagrama de Sequência da integração entre o cliente o Módulo Conector de Licenças.....	133
Figura 76 Diagrama de Estrutura Composta do Módulo Conector de Licenças Maven.....	134
Figura 77 Diagrama de Estrutura Composta do Módulo Transformador RDF de Controle de Versão	135

Figura 78 Diagrama de Estrutura Composta do Módulo Transformador RDF de Licença.	138
Figura 79 Diagrama de Estrutura Composta do Módulo Transformador RDF de Artefatos	139
Figura 80 Diagrama de Estrutura Composta do Módulo Transformador RDF de Código-Fonte	141
Figura 81 Diagrama de Estrutura Composta do Módulo Transformador RDF de Impactos	143
Figura 82 Diagrama de Estrutura Composta do Módulo Transformador RDF de Integração Contínua	145
Figura 83 Diagrama de Estrutura Composta do Módulo Transformador RDF de Demandas/Defeitos	147
Figura 84 Diagrama de Estrutura Composta do Módulo Transformador RDF de Testes.....	149
Figura 85 Diagrama de Estrutura Composta do Módulo FOAF	150
Figura 86 Diagrama de Estrutura Composta do Módulo DOAP	151
Figura 87 Diagrama de Estrutura Composta do Módulo de Verificação do Tipo de Arquivo.....	153
Figura 88 Diagrama de Estrutura Composta do Módulo de Análise de Impactos	155
Figura 89 Diagrama de Componentes dos Módulos utilizados pelo Módulo de Coordenação de Extração de Informações de Ferramentas de Controle de Versão	156
Figura 90 Diagrama de Atividades do Processo de Extração de Informações de Ferramentas de Controle de Versão	157
Figura 91 Diagrama de Atividades Tratar Arquivos.....	159
Figura 92 Diagrama de Atividades Tratar Arquivo de Licença.....	160
Figura 93 Diagrama de Atividades Tratar Arquivo de Dependências	160
Figura 94 Diagrama de Atividade Tratar Arquivo de Código-Fonte.	161
Figura 95 Diagrama de Atividade Tratar Impactos na AST do Arquivo	162
Figura 96 Diagrama de Componentes dos Módulos utilizados pelo Módulo Coordenador do Processamento da Extração de Informações de uma Versão de uma Biblioteca.	163
Figura 97 Diagrama de Atividades do Processo de Extração de Informações de uma Versão de uma Biblioteca.	163
Figura 98 Diagrama de Componentes dos Módulos utilizados pelo	

Módulo Coordenador de Extração de Informações de Ferramentas de Integração Contínua.	165
Figura 99 Diagrama de Atividades Extrair Informações de Integração Contínua.	165
Figura 100 Diagrama de Componentes dos Módulos utilizados pelo Módulo de Coordenação de Extração de Diferenças entre duas Versões de uma Biblioteca.	167
Figura 101 Diagrama de Atividades do Processo de Extração das diferenças entre duas versões de uma biblioteca.	167
Figura 102 Diagrama de Componentes dos Módulos utilizados pelo Módulo Coordenador de Extração de Informações de Ferramentas de Gerenciamento de Demandas/Defeitos.	168
Figura 103 Diagrama de Atividades Extrair Informações de Demandas/Defeitos.	169

Lista de tabelas

Tabela 1 Principais elementos do RDFS.....	34
Tabela 2 Elementos do vocabulário DOAP	40
Tabela 4 Principais termos do vocabulário Dublin Core	43
Tabela 6 Elementos vocabulário OSLC Change Management	47
Tabela 7 Elementos do vocabulário OSLC Auto	48
Tabela 8 Elementos do Vocabulário OSLC Asset Management	49
Tabela 9 Elementos do vocabulário OSLC Quality Management.....	51
Tabela 10 Elementos do vocabulário OSLC SCM.....	52
Tabela 11 Principais elementos da ontologia de Entidades, Propriedades e Relacionamentos	55
Tabela 12 Principais elementos da ontologia de Impactos.....	61
Tabela 13 Perguntas Frequentes dos Desenvolvedores.....	91
Tabela 14 Informações da execução do processo de extração.....	97
Tabela 15 Resultado da Consulta “Quem alterou classes que eu modifíco?”.....	99
Tabela 16 Resultado da Consulta “Quem está utilizando esta API?” para o método “org.apache.mahout.common.AbstractJob.getOption(String)”	101
Tabela 17 Resultado da Consulta “Quem está utilizando esta API?” para o método “org.apache.mahout.common.AbstractJob.addOption(String)”	101
Tabela 18 Resultado da Consulta “Para quem atribuir uma revisão de código”	102
Tabela 19 Listagem de desenvolvedores que já trabalharam como a biblioteca Lucene.....	103

Lista de quadros

Quadro 1 Consulta SPARQL que retornar entidades afetadas pela mudança de versão de uma biblioteca do projeto	95
Quadro 2 Consulta SPARQL que retorna os elementos afetados por defeitos corrigidos em uma nova versão de uma biblioteca utilizada no projeto	96
Quadro 3 Consulta SPARQL que retorna os elementos que podem ter sido afetados por defeitos devido à atualização da versão de uma biblioteca do projeto.....	97
Quadro 4 Consulta SPARQL que retorna o desenvolvedor que realizou o maior número de modificações nos elementos de código-fonte do projeto	98
Quadro 5 Consulta SPARQL que retorna os dois métodos mais invocados no projeto.....	100