

Marcos Gonçalves Rios

Documentação de Programas por Contrato

Tese de Doutorado apresentada ao Curso de Pós-Graduação em Ciência da Computação da Universidade Federal de Minas Gerais, como requisito parcial para a obtenção do grau de Doutor em Ciência da Computação.

Belo Horizonte

2001



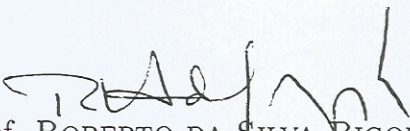
UNIVERSIDADE FEDERAL DE MINAS GERAIS

FOLHA DE APROVAÇÃO

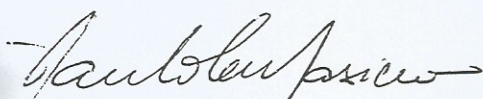
Documentação de Programas por Contrato

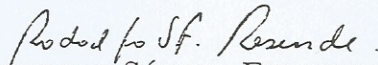
MARCOS GONÇALVES RIOS

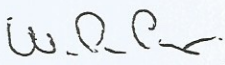
Tese defendida e aprovada pela banca examinadora constituída pelos Senhores:


Prof. ROBERTO DA SILVA BIGONHA - Orientador
Departamento de Ciência da Computação - ICEX - UFMG


Prof. JÚLIO CÉSAR SAMPAIO DO PRADO LEITE
Departamento de Informática - PUC-Rio


Prof. PAULO CÉSAR MASIERO
Instituto de Ciências Matemáticas e de Computação - USP - São Carlos


Prof. RODOLFO SÉRGIO FERREIRA DE RESENDE
Departamento de Ciência da Computação - ICEX - UFMG


Prof. WILSON DE PÁDUA PAULA FILHO
Departamento de Ciência da Computação - ICEX - UFMG

Belo Horizonte, 08 de maio de 2001.

À minha família.

Agradecimentos

- Ao Prof. Wilson Luiz de Almeida, Chefe do Departamento Acadêmico de Engenharia Industrial Mecânica do CEFET-MG, pelo apoio concedido.
- Ao Prof. Roberto da Silva Bigonha, orientador deste trabalho, por mostrar o caminho e apontar erros e acertos.
- Ao Lintz Veloso, meu colega e ex-aluno, pela ajuda na fase de validação da técnica proposta.
- À Ana Luiza Aragão Bracarense pelo apoio, incentivo e, também, pela ajuda na revisão do texto.
- À Profa. Mariza A. S. Bigonha pelas diversas críticas construtivas.

Sinopse

Este trabalho, que tem o engenheiro de software como audiência principal, objetiva oferecer uma nova técnica de produzir documentação consistente de programas para suporte à manutenção. Inicialmente, é feita uma exposição da importância de se possuir documentos consistentes e de fácil acesso, por meio da apresentação de resultados de pesquisas sobre práticas e custos de manutenção de software. Em seguida, é discutida a questão da geração automática de documentação, e algumas tecnologias relevantes são apresentadas para apontar vantagens e desvantagens das soluções existentes. Uma segunda geração de métodos de modelagem, baseados em técnicas mais precisas de especificação, é estudada demonstrando a tendência de utilização de um maior formalismo na especificação de software que favorece a geração de documentação. A solução do difícil problema de consistência de documentação proposta neste trabalho consiste na geração automática de documentação de programas desenvolvidos segundo os preceitos da técnica de *Design by Contract* [Meyer, 1999]. A origem da documentação a ser gerada é o próprio código fonte de acordo com o estilo *Literate Programming* [Knuth, 1992]. A documentação obtida é um *Hiperdocumento* [Martin, 1992], denominado aqui *Hipercontrato*, que estabelece as especificações de requisitos e as formas de tratamento de exceção. A contribuição oferecida por este trabalho consiste no tratamento da documentação de software como um contrato entre módulos, a exemplo do mundo jurídico. *Assinaturas Digitais* [Rivest, 1978] são, também, usadas para se assegurar a inviolabilidade do contrato. Por fim, o desenvolvimento de um conjunto de ferramentas é parte complementar desta tese com o objetivo de produzir um estudo de caso para validar esta nova forma de documentação.

Abstract

This work, that has the software engineer as its main audience, aims to offer a new technique of producing consistent software documentation for supporting the maintenance process. Initially, it highlights the importance of dealing with consistent and easily accessible documents, through the presentation of research results on practices and costs of software maintenance. In the sequel, the issue of the documentation automatic generation is discussed and some relevant technologies are presented to point out advantages and disadvantages of the existent solutions. A second generation of software methods, based on more precise techniques of software specification, is studied demonstrating the trend to stronger formalism use in the software specification that favors documentation generation. The solution proposed to the hard problem of documents consistency, in this work, is based on the documentation automatic generation through software developed according the rules of *Design by Contract* [Meyer, 1999]. The source of the documentation to be generated is the proper code source in accordance with the *Literate Programming* style [Knuth, 1992]. The obtained documentation is a *Hiperdocument* [Martin, 1992], denoted here *Hipercontract*, that establishes the specifications of requirements and the forms of exceptions handling. The contribution offered by this work consists of handling software documentation as a contract among modules, like an official legal agreement. Digital signatures [Rivest, 1978] is also used to assure integrity. Finally, the development of a tool set is a complementary part of this thesis in order to build a case study to validate this proposed form of software documentation.

Lista de Ilustrações

1.1	Divisão dos Custos de Manutenção.....	3
1.2	Ciclo de Vida de um Software Orientado por Objetos.....	8
2.1	Processo de Geração Automática de Documentação.....	14
2.2	<i>Literate Programming</i>	16
2.3	O Sistema Hyperpro.....	18
3.1	Especificação de Requisitos.....	26
3.2	Operações em Tipos de Coleção.....	34
4.1	Visão Estática do Contrato.....	42
4.2	Geração do Contrato por meio da Ferramenta Documentador.....	45
4.3	Diagrama de Classes e suas Diferentes Versões.....	46
5.1	Relação entre a unidade de compilação, o hipercontrato e o código objeto.....	51
5.2	Sistema convencional de criptografia.....	52
5.3	Sistema de chaves públicas.....	52
5.4	Assinatura Digital.....	53
5.5	Sistema de Criptografia Híbrido.....	54
5.6	Diagrama Funcional da Ferramenta de Assinatura – Tabelaio.....	56
5.7	Processo Básico de Alteração [Fiorini, 1998].....	59
6.1	Aplicação da Técnica Proposta.....	63
6.2	Diagrama de Classes da Companhia Real.....	65
6.3	Invariante da Classe Cliente.....	66
6.4	Relação entre os três artefatos.....	68
6.5	Interface do Documentador.....	71
6.6	Interface Primária do Documentador.....	72
6.7	Interface Secundária do Documentador.....	73

Sumário

Capítulo 1 - Documentação de Programas

1.1 Introdução.....	1
1.2 Propriedades da Documentação de Alta Qualidade.....	4
1.3 As Três Técnicas.....	6
1.3.1 <i>Literate Programming</i>	6
1.3.2 Projeto por Contrato.....	7
1.3.3 Controle de Versão e Assinatura Digital.....	7
1.4 Diretrizes para Obtenção da Solução.....	8
1.5 Conclusão.....	9
1.6 Organização da Tese.....	11

Capítulo 2 - Geração Automática de Documentação

2.1 Introdução.....	13
2.2 <i>Literate Programming</i> (LP).....	15
2.3 HyperPro e Thot.....	17
2.4 Documentu.....	19
2.5 Javadoc.....	20
2.6 Outras Ferramentas de Documentação.....	20
2.7 Conclusão.....	23

Capítulo 3 - Formalização de Requisitos

3.1 Introdução.....	25
3.2 Especificação de Programas.....	25
3.2.1 Requisitos.....	25
3.2.2 Métodos Formais.....	27
3.2.3 Cálculo de Refinamento.....	28
3.2.4 Linguagem Z e Object Z.....	29
3.2.5 <i>Viena Development Method</i> (VDM) e VDM ++.....	29
3.2.6 <i>Object Constraint Language</i> (OCL).....	30
3.3 Linguagens de Programação.....	31
3.3.1 Asserções em Eiffel.....	31
3.3.2 Asserções em Java.....	31
3.4 Asserções na Técnica Proposta.....	32
3.4.1 Asserções à Francesa.....	32
3.4.2 Asserções nos Métodos Estudados.....	33
3.4.3 Asserções em OCL.....	33
3.5 Conclusão.....	35

Capítulo 4 – Hipercontrato

4.1 Introdução.....	36
4.2 Projeto por Contrato.....	37
4.3 Outros Contratos.....	40
4.4 O Paradigma da Programação.....	41
4.5 O Modelo do Contrato.....	42
4.6 A Navegação no Contrato.....	43
4.7 Função das Asserções durante as Atividades de Manutenção.....	43
4.7.1 Quebras de Contrato.....	44
4.7.2 Tratamento de exceções.....	44
4.8 A Geração do Contrato: Documentador.....	45
4.9 Relacionamento do Documentador com o Ciclo de Vida.....	46
4.10 Visão Sistêmica.....	46
4.11 CMM e a Técnica Proposta: Hipercontrato.....	47
4.12 Conclusão.....	49

Capítulo 5 – Assinatura Digital

5.1 Introdução.....	50
5.2 Criptografia.....	51
5.2.1 Assinatura Digital.....	53
5.2.2 Sistemas Reais.....	54
5.3 Como a Assinatura é Implementada na Ferramenta.....	55
5.4 Contribuição ao Processo de gestão de Configurações.....	57
5.4.1 CMM e a Técnica Proposta: Tabelião.....	58
5.4.2 Comparação com Outras Alternativas.....	58
5.5 relacionamento do Tabelião com o Ciclo de Vida.....	60
5.6 Conclusão.....	60

Capítulo 6 - Validação da Técnica Proposta

6.1 Aplicação da Técnica de Forma Geral.....	62
6.2 Descrição do Exemplo: Companhia Real.....	64
6.2.1 Vantagens e Dificuldades na Aplicação da Técnica.....	66
6.3 Como a Técnica Auxilia a Manutenção.....	66
6.4 Como as propriedades de uma Boa Documentação são Atingidas.....	67
6.5 Análise Comparativa com Outras Técnicas.....	69
6.6 Avaliação das Ferramentas Desenvolvidas: tabelião e Documentador.....	70
6.6.1 Documentador.....	70
6.6.2 Tabelião.....	71
6.7 Conclusão.....	73

Capítulo 7 - Conclusão	
7.1 Introdução.....	75
7.2 Resultados Finais.....	75
7.3 Contribuições.....	76
7.4 Algumas Questões.....	77
7.5 Trabalhos Futuros.....	78
Apêndices	
A .1 Índice do Hipercontrato.....	80
A .2 Diagrama de Classes.....	81
A .3 Diagrama de Caso de Uso.....	82
A .4 Cláusula Cliente.....	83
A .6 Cláusula CartãoCliente.....	89
A .7 Cláusula Classe Abstrata de Interface de Transação.....	94
Bibliografia.....	96