

Frederico Silva Guimarães

**Um Sistema Multi-Agentes para
Monitoramento e Aquisição em
Tempo Real**

DISSERTAÇÃO DE MESTRADO

**DEPARTAMENTO DE INFORMÁTICA
Programa de Pós-graduação em
Informática**

Rio de Janeiro
Abril de 2006

Frederico Silva Guimarães

**Um Sistema Multi-Agentes para
Monitoramento e Aquisição em Tempo Real**

Dissertação de Mestrado

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

Orientador: Prof. Carlos José Pereira de Lucena

Rio de Janeiro
Abril de 2006



Frederico Silva Guimarães

**Um Sistema Multi-Agentes para
Monitoramento e Aquisição em Tempo Real**

Dissertação apresentada como requisito parcial para obtenção do grau de Mestre pelo Programa de Pós-graduação em Informática do Departamento de Informática do Centro Técnico Científico da PUC-Rio. Aprovada pela Comissão Examinadora abaixo assinada.

Prof. Carlos José Pereira de Lucena

Orientador

Departamento de Informática — PUC-Rio

Prof. Arndt Von Staa

PUC-Rio

Prof. Firmo Freire

PUC-Rio

Prof. José Eugenio Leal

Coordenador Setorial do Centro Técnico Científico —

PUC-Rio

Rio de Janeiro, 5 de Abril de 2006

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

Frederico Silva Guimarães

Graduou-se em Engenharia Computação na Pontifícia Universidade Católica do Rio de Janeiro (Brasil, Rio de Janeiro).

Ficha Catalográfica

Guimarães, Frederico Silva

Um Sistema Multi-Agentes para Monitoramento e Aquisição em Tempo Real/ Frederico Silva Guimarães; orientador: Carlos José Pereira de Lucena. — Rio de Janeiro : PUC–Rio, Departamento de Informática, 2006.

v., 93 f: il. ; 29,7 cm

1. Dissertação (mestrado) - Pontifícia Universidade Católica do Rio de Janeiro, Departamento de Informática.

Inclui referências bibliográficas.

1. Design by Contract. 2. Sistemas de Tempo Real. 3. Software Embarcado. 4. Sistemas Multi-Agentes. 5. Sistemas Orientados à Recuperação. 6. Tolerância a Falhas. I. Lucena, Carlos José Pereira de. II. Pontifícia Universidade Católica do Rio de Janeiro. Departamento de Informática. III. Título.

CDD: 510

Agradecimentos

A execução e finalização deste trabalho não se dariam sem a participação de algumas pessoas para as quais gostaria de mostrar aqui meu agradecimento.

Aos meus pais, Frederico e Letícia, e a minha família, pela compreensão, amor e carinho de sempre. À Juliana, pelo apoio irrestrito, por toda a compreensão, cumplicidade e incentivo.

Ao meu orientador, professor Carlos J. P. de Lucena, por diversas vezes clarear dúvidas em minha mente e minimizar os erros durante todo o trabalho de pesquisa. Ao João, "co-orientador" extra-oficial, que muito me apoiou com idéias e incentivos durante esta caminhada. Aos professores Arndt e Firmo pela atenção, pelas idéias e pelo apoio na definição deste trabalho.

À todos os meus amigos, em especial o pessoal da PUC (LF, Mauro Lual, Paulinha, Carol, Alexandre, Clara, Felipe, ...) e da MindsAtWork (Bruno, Ricardo, Rodrigo, Hevert, Guilherme, ...), pela alegria, companheirismo e pelos momentos de descontração, combustíveis para inspiração e criatividade. Aos colegas do laboratório TecComm, pela amizade e ajuda. Obrigado pelos ensinamentos e incentivo que muito colaboraram para a realização deste trabalho!

À todos os professores do Departamento de Informática que, durante a minha graduação, compartilharam comigo seus conhecimentos, sem os quais teria sido impossível realizar esse trabalho. À todos os funcionários do Departamento de Informática e da Fundação Padre Leonel Franca que, de alguma forma, contribuíram para a realização desse trabalho.

À CAPES e ao Departamento de Informática pelo apoio financeiro que possibilitou a realização deste trabalho.

Resumo

Guimarães, Frederico Silva; Lucena, Carlos José Pereira de. **Um Sistema Multi-Agentes para Monitoramento e Aquisição em Tempo Real**. Rio de Janeiro, 2006. 93p. Dissertação de Mestrado — Departamento de Informática, Pontifícia Universidade Católica do Rio de Janeiro.

Sistemas supervisores envolvendo software embarcados são encontrados com frequência e são responsáveis pela supervisão de equipamentos que vão desde máquinas industriais e eletrodomésticos, a celulares e PDAs. Muitos possuem severos requisitos de confiabilidade e tolerância a falhas, bem como características de sistemas de tempo real. Esta pesquisa investiga o uso de tecnologias de ponta como *Design by Contract*, Agentes de Software, *Mock Objects* e Componentes de Software no auxílio ao desenvolvimento de sistemas de monitoramento e aquisição em tempo real. Explora-se, ainda, o conceito de sistemas orientados à recuperação. Para realizar tal análise é utilizado como estudo de caso um sistema de inspeção de dutos, que foi implementado utilizando tais tecnologias. De maneira resumida, pode-se dizer que neste estudo de caso percebeu-se que o uso dessas tecnologias fez com que o sistema fosse desenvolvido em um curto espaço de tempo, sendo que questões relevantes não foram prejudicadas, como por exemplo o fato do sistema ser robusto e tolerante a falhas. É importante ressaltar que o sistema apresentado se encontra em uso em inspeções de dutos reais.

Palavras-chave

Design by Contract; Sistemas de Tempo Real; Software Embarcado; Sistemas Multi-Agentes; Sistemas Orientados à Recuperação; Tolerância a Falhas.

Abstract

Guimarães, Frederico Silva; Lucena, Carlos José Pereira de. **A Multi-Agent System for Real Time Monitoring and Data Acquisition.** Rio de Janeiro, 2006. 93p. MSc. Dissertation — Departamento de Informática, Pontifícia Universidade Católica do Rio de Janeiro.

Supervisory systems allied to embedded software, a common subject in the literature, are responsible by the supervision of equipments like industrial machines, household-electric devices, cellular phones and PDAs. Many of them have severe requisites of fault tolerance and reliability as well as real time systems features. This research investigates the use of state-of-the-art technologies such as Design By Contract, Mock Objects, software agents and components in the development of real-time monitoring and data-acquisition systems. Recovery oriented systems concepts are also explored. The case study carried out to the research is a pipeline inspection software system, which uses such technologies. In a nutshell, it can be said that this case study showed strong evidence that the use of these technologies allowed the development in a very short time, without any loss of robustness, fault tolerance and reliability. The developed system is currently being used in real pipeline inspections.

Keywords

Design by Contract; Real Time Systems; Firmware; Embedded Software; Multi-Agent Systems; Recovery Oriented Systems; Fault Tolerance.

Conteúdo

1	Introdução	11
1.1	Contextualização	11
1.2	O Estudo de Caso	13
1.3	Organização da Dissertação	16
2	Tecnologias Envolvidas	17
2.1	Software Embarcado	17
2.2	Software de Monitoramento e Aquisição de Dados	18
2.3	Sistemas de Tempo Real	18
2.4	Sistemas Orientados à Recuperação	19
2.5	Agentes de Software	21
2.6	<i>Design by Contract</i>	23
2.7	<i>Bluetooth</i>	25
2.8	Biblioteca Log4Cxx	26
2.9	Biblioteca SQLite	30
2.10	Biblioteca Qt	33
2.11	<i>Mock Objects</i>	37
2.12	Componentes de Software	39
3	Trabalhos Relacionados	41
4	Arquitetura Adotada	45
4.1	O Software Embarcado	46
4.2	O Software Supervisor	46
4.3	A Comunicação	49
4.4	Tecnologias Utilizadas	50
5	Implementação	51
5.1	O Software Embarcado	51
5.2	O Software Supervisor	54
5.3	Uso de <i>Mock Objects</i>	56
5.4	Usando Assertivas	59
5.5	<i>Design by Contract</i> e <i>Mock Elements</i>	70
5.6	Banco de Dados	73
5.7	Arquitetura das Janelas de Visualização	79
6	Conclusões e Trabalhos Futuros	83
6.1	Conclusões	83
6.2	Trabalho Futuros	87
	Referências Bibliográficas	89

Lista de Figuras

1.1	Dutos de transporte de petróleo	13
1.2	Duto com corrosão	13
1.3	PIG sendo inserido em um duto	14
1.4	Inspetor externo de dutos (IED)	15
1.5	Obstáculos em um duto	15
2.1	Herança dos <i>Loggers</i>	29
2.2	Pré-processamento do Qt	34
2.3	Conectando <i>slots</i> a <i>signals</i>	35
4.1	Arquitetura geral do sistema	45
4.2	Arquitetura do software embarcado	47
4.3	Arquitetura do software supervisor	48
5.1	Diagrama de colaboração dos agentes - Software Embarcado	53
5.2	Diagrama de colaboração dos agentes - Software Supervisor	55
5.3	Diagrama - <i>Mock Tool</i>	57
5.4	Diagrama - <i>Mock Interpreter</i>	59
5.5	Exemplo de mensagem exibida pela macro <i>ABORT</i>	61
5.6	Exemplo de mensagem exibida pela macro <i>ASSERT</i>	61
5.7	Exemplo de mensagem exibida pela macro <i>ASSERT_VALID</i>	63
5.8	Outro exemplo de mensagem exibida pela macro <i>ASSERT_VALID</i>	63
5.9	Diagrama de classes simplificado do <i>AssertionHandler</i>	68
5.10	Fluxo de execução do <i>AssertionHandler</i>	69
5.11	Diagrama de classes dos <i>Beans</i>	75
5.12	Alcance horário das ferramentas de inspeção	77
5.13	Diagrama de classes das <i>Views</i>	80

Lista de Tabelas

5.1	Exemplo de tabela de configuração do <i>AssertionHandler</i>	68
6.1	Estatísticas do Sistema	85

Lista de Códigos Fonte

2.1	Configuração do log4cxx	27
2.2	Registrando mensagens no log4cxx	28
2.3	Função sqlite3_exec	31
2.4	Exemplo de função de callback	32
2.5	Exemplo QObject - <i>Signals</i> e <i>Slots</i>	35
2.6	Exemplo Qt::connect	36
2.7	Exemplo Qt::disconnect	36
2.8	Exemplo emitindo <i>Signal</i>	36
2.9	Exemplo de caso teste com Mock Objects utilizando JMock	38
5.1	Assinatura da macro ABORT	60
5.2	Exemplo de uso da macro ABORT	60
5.3	Assinatura da macro ASSERT	60
5.4	Exemplo de uso da macro ASSERT	61
5.5	Algumas funções para validação de ponteiros	62
5.6	Assinatura da macro ASSERT_VALID	62
5.7	Exemplo de uso da macro ASSERT_VALID	64
5.8	Utilizando o <i>log</i> para desabilitar assertivas - Parte I	65
5.9	Utilizando o <i>log</i> para desabilitar assertivas - Parte II	65
5.10	Utilizando o <i>log</i> para desabilitar assertivas - Parte III	66