



**Adriano Medeiros dos Santos**

**Suporte a Componentes Compostos Para o  
Middleware SCS**

**Dissertação de Mestrado**

Dissertação apresentada ao Programa de Pós-graduação em Informática do Departamento de Informática do Centro Técnico Científico da PUC–Rio como requisito parcial para obtenção do grau de Mestre em Informática.

Orientador: Prof. Renato Fontoura de Gusmão Cerqueira

Rio de Janeiro  
Abril de 2012



**Adriano Medeiros dos Santos**

## **Suporte a Componentes Compostos Para o Middleware SCS**

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

**Prof. Renato Fontoura de Gusmão Cerqueira**

Orientador

Departamento de Informática — PUC-Rio

**Prof. Alessandro Fabricio Garcia**

Departamento de Informática — PUC-Rio

**Prof. Antônio Tadeu Azevedo Gomes**

LNCC

**Prof. José Eugenio Leal**

Coordenador Setorial do Centro Técnico Científico — PUC-Rio

Rio de Janeiro, 03 de Abril de 2012

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

### **Adriano Medeiros dos Santos**

Graduou-se em Bacharelado em Ciência da Computação pela UESC. É pesquisador do laboratório Tecnologia em Computação Gráfica (Tecgraf) da PUC-Rio desde Agosto de 2009, onde trabalha no desenvolvimento do *middleware* SCS que apoia diversos projetos em parceria com a Petrobras S/A.

#### Ficha Catalográfica

Santos, Adriano Medeiros

Suporte a componentes compostos para o middleware scs / Adriano Medeiros dos Santos; orientador: Renato Fontoura de Gusmão Cerqueira. — Rio de Janeiro : PUC–Rio, Departamento de Informática, 2012.

v., 99 f: il. ; 29,7 cm

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

Inclui referências bibliográficas.

1. Informática – Tese. 2. Modelos de Componentes de Software. 3. Programação Orientada a Componentes. 4. Componentes Compostos. 5. Middleware. I. Cerqueira, Renato. II. Pontifícia Universidade Católica do Rio de Janeiro. Departamento de Informática. III. Título.

CDD: 004

## Agradecimentos

Ao meu pai Josembergue Campos dos Santos, meu exemplo de como ser uma pessoa dedicada ao trabalho e sempre me ajudar em decisões como por exemplo investir neste mestrado.

À minha mãe Joíse Cristina Medeiros dos Santos, pelo carinho e amor que me deram força para este trabalho.

À minha namorada Tiara, pelo amor e compreensão da necessidade do namoro à distância.

Ao meu orientador Renato Cerqueira, que acreditou em meu trabalho e me inspirou a alcançar meus objetivos.

À PUC-Rio e Tecgraf, pelos infraestruturas, sem os quais este trabalho não poderia ter sido realizado.

Ao Amadeu pelas explicações e por me mostrar que sempre é bom pensar bem antes de discutir uma idéia.

Aos meus amigos, que fizeram com que esse mestrado no Rio de Janeiro longe da família fosse bem animado.

## Resumo

Santos, Adriano Medeiros; Cerqueira, Renato. **Suporte a Componentes Compostos Para o Middleware SCS**. Rio de Janeiro, 2012. 99p. Dissertação de Mestrado — Departamento de Informática, Pontifícia Universidade Católica do Rio de Janeiro.

Diversos modelos de componentes de software apresentam o conceito de componentes compostos, como é o caso do Fractal, OpenCOM, Koala, Kobra e SaveCCM. Um componente composto encapsula outros componentes e é responsável por oferecer mecanismos de configuração e introspecção estrutural, bem como externalizar serviços e dependências internas. A idéia de componentes compostos é reconhecida como uma boa técnica para abstrair estruturas complexas e maximizar o reuso. Entretanto, não existe um estudo mais aprofundado sobre tal mecanismo que aborde questões sobre a configuração inicial dos subcomponentes, o mapeamento de serviços e dependências de subcomponentes através de interfaces do componente composto, formas de navegação, compartilhamento de subcomponentes, ciclo de vida e formas de introspecção estrutural. Neste trabalho foi realizado um levantamento sobre como os modelos Fractal e OpenCOM tratam o conceito de componentes compostos e é proposto um modelo que oferece este suporte. Como estudo experimental foi implementada uma nova versão do SCS com suporte a componentes compostos, o SCS-Composite. Também foi feita uma avaliação da expressividade do SCS-Composite através da adaptação de uma aplicação pré-existente baseada no SCS.

## Palavras-chave

Modelos de Componentes de Software. Programação Orientada a Componentes. Componentes Compostos. Middleware.

## Abstract

Santos, Adriano Medeiros; Cerqueira, Renato (Advisor).  
**Composite Component Support for the SCS Middleware.**  
Rio de Janeiro, 2012. 99p. MSc Dissertation — Departamento de  
Informática, Pontifícia Universidade Católica do Rio de Janeiro.

Several software component models support composite components, such as Fractal, OpenCOM, Koala, Kobra and SaveCCM. A composite component encapsulates components and provides introspection and configuration mechanisms. Composite components are recognized as a good practice because abstract complex structures and increase reuse. However, there is not a comprehensive study about composite component mechanisms addressing issues like initial composite configuration, mapping of subcomponents' services and dependencies through composite component's interfaces, introspection, component sharing, and composite's life cycle. In this work we perform a study about how the Fractal and OpenCOM apply the composite component concept and we propose a new model for composite components. As an experimental study we implemented a new version of the SCS component system with support for composite components, the SCS-Composite. We also present an evaluation of SCS-Composite through the adaptation of a SCS based application.

## Keywords

Software Component Models. Component-oriented programming.  
Composite Component. Middleware.

## Sumário

1	Introdução	12
1.1	Objetivos e Contribuições	15
1.2	Estrutura do Documento	16
2	Trabalhos Relacionados	18
2.1	Fractal	19
2.2	OpenCOM	22
2.3	Características de Componentes Compostos	26
2.3.1	Suporte ao Desenvolvimento Incremental	26
2.3.2	Mapeamento de Serviços dos Subcomponentes como Serviços do Componente Composto	27
2.3.3	Mapeamento de dependências de subcomponentes como dependências do componente composto	29
2.3.4	Conectores para <i>Binding</i> Vertical	31
2.3.5	Relacionamento Serviço/Componente	32
2.3.6	Compartilhamento de subcomponentes	34
2.3.7	Navegação entre Subcomponente e Componente Composto	36
2.3.8	Mecanismos de Reconfiguração	36
2.4	Considerações Finais	37
3	Modelo SCS com Suporte a Componentes Compostos	39
3.1	Modelo de Componentes SCS	39
3.1.1	Componente	39
3.1.2	Mecanismo de <i>Binding</i>	40
3.2	Modelo de Componentes SCS- <i>Composite</i>	43
3.2.1	Componente Composto	43
3.2.2	Hierarquia de Componentes	44
3.2.3	Externalização de Facetas e Receptáculos	44
3.2.4	Regras de Conexão	46
3.3	Diretrizes do Modelo	47
3.3.1	Suporte ao Desenvolvimento Incremental	47
3.3.2	Mapeamento de Serviços dos Subcomponentes como serviços do componente composto e Conectores Exógenos	48
3.3.3	Relacionamentos Serviço/Componente e Subcomponente/Componente Composto	50
3.3.4	Mapeamento de Dependências de Subcomponentes como Dependências do Componente Composto	50
3.3.5	Compartilhamento de subcomponentes	51
3.3.6	Mecanismos de Reconfiguração	52
3.4	Considerações Finais	52
4	<i>Middleware SCS-Composite</i>	55
4.1	<i>Middleware SCS</i>	55
4.2	<i>Middleware SCS-Composite</i>	57

4.2.1	Estrutura do Componente	57
4.2.2	Mecanismo de <i>binding</i>	60
4.3	Exemplo de Uso	62
4.3.1	Mapeamento de Facetas	62
4.3.2	Mapeamento de Receptáculos	63
4.4	Considerações Finais	65
5	Exemplo de Uso	<b>67</b>
5.1	Modelagem do CAS	67
5.2	Implementação com <i>Middleware SCS</i>	68
5.3	Implementação com <i>Middleware SCS-Composite</i>	71
5.3.1	Gerência sobre Componentes	71
5.3.2	Mapeamento de Serviços Internos	72
5.3.3	<i>Binding</i> Vertical com Aridade 1-n	72
5.3.4	Gerência de Configuração da Aplicação	73
5.4	Comparação do CAS: SCS Vs. SCS- <i>Composite</i>	74
6	Conclusão	<b>77</b>
7	Referências Bibliográficas	<b>80</b>
A	<i>Middleware SCS 2.0</i>	<b>83</b>
B	Exemplos de Uso	<b>86</b>
B.1	Mapeamento de Facetas	86
B.2	Mapeamento de Receptáculos	90
C	Interfaces do CAS	<b>96</b>

## Lista de figuras

1.1	Sistema baseado em componentes. Em 1 é ilustrado o <i>binding</i> entre dois componentes; e em 2, é ilustrado <i>binding</i> entre um componente e a plataforma operacional.	13
1.2	Componente primitivo e composto.	14
1.3	Representação dos dois tipos de <i>bindings</i> verticais	14
2.1	Estrutura de um componente no modelo Fractal.	19
2.2	Arquitetura do modelo de componentes OpenCOM (Michael Clarke, 2007).	23
2.3	Um <i>Component Framework</i> em OpenCOM.	24
2.4	Mapeamentos de serviços dos subcomponentes através de interfaces do componente composto.	27
2.5	Exemplo de um cenário do CAS demonstrando a necessidade de disponibilizar serviços internos por interfaces externas diferentes.	28
2.6	Mapeamentos de dependências dos subcomponentes através de interfaces externas do componente composto.	30
2.7	Arquitetura de uma interface gráfica com compartilhamento de subcomponentes.	34
2.8	Cenário onde o compartilhamento de subcomponentes nos leva a uma situação de ambiguidade quanto as dependências do componente compartilhado.	35
3.1	Representação de um componente SCS.	41
3.2	Exemplo de conexão entre facetas e receptáculos de componentes SCS.	42
3.3	Exemplo de uma composição formada pelos componentes A, B e C.	42
3.4	Os dois tipos de receptáculos presentes no <i>SCS-Composite</i>	46
3.5	Exemplos de conexões entre componentes primitivos e compostos.	48
3.6	Mapeamento de uma faceta de um subcomponente para facetas externas diferentes.	48
3.7	Funcionamento de um conector no processo de externalização de facetas.	49
3.8	Representação de externalização de um receptáculo de um subcomponente.	51
4.1	Os dois tipos de receptáculos presentes no <i>SCS-Composite</i>	61
4.2	Arquitetura do exemplo de uso do <i>SCS-Composite</i> Local	62
4.3	Arquitetura do exemplo de uso do <i>SCS-Composite</i> Distribuído.	64
5.1	Modelagem realizada para arquitetura atual do CAS.	69
5.2	Implementação do CAS com o SCS sem suporte a componentes compostos.	69
5.3	Exemplo do uso de conectores no CAS.	73

## Lista de tabelas

- 2.1 Comparação entre os modelos Fractal e OpenCOM sobre características de componentes compostos em conjunto com as decisões realizadas para o modelo SCS. 38

*A mente que se abre a uma nova idéia jamais  
voltará ao seu tamanho original.”*

**Albert Einstein (1879–1955)**