

Marcelo de Almeida Maia

Especificação Formal da Interação de Componentes de Sistemas Computacionais

Especificação Formal da Interação de Componentes de Sistemas Computacionais

Tese 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

27 de agosto de 1999



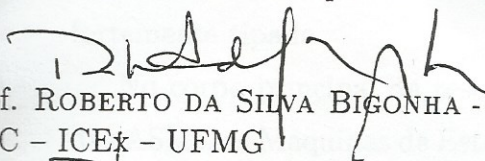
UFMG - ICEx
DEPARTAMENTO DE CIÊNCIA DA
COMPUTAÇÃO

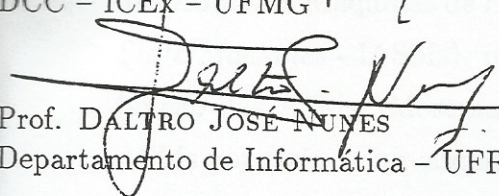
FOLHA DE APROVAÇÃO

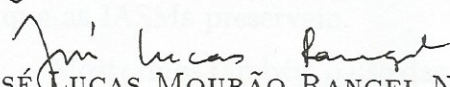
Especificação Formal da Interação de Componentes de Sistemas Computacionais

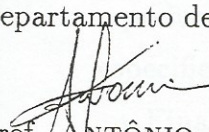
MARCELO DE ALMEIDA MAIA

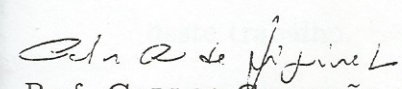
Tese defendida e aprovada pela banca examinadora constituída pelos Senhores:

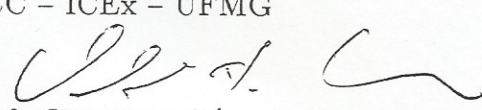

Prof. ROBERTO DA SILVA BIGONHA - Orientador
DCC - ICEx - UFMG


Prof. DALTRO JOSÉ NUNES
Departamento de Informática - UFRGS


Prof. JOSÉ LUCAS MOURÃO RANGEL NETTO
Departamento de Informática - PUC/Rio


Prof. ANTÔNIO ALFREDO FERREIRA LOUREIRO
DCC - ICEx - UFMG


Prof. CARLOS CAMARÃO DE FIGUEIREDO
DCC - ICEx - UFMG


Prof. OSVALDO SÉRGIO FARHAT DE CARVALHO
DCC - ICEx - UFMG

Belo Horizonte, 27 de agosto de 1999.

Resumo

Esta tese trata da especificação formal de aspectos da interação entre componentes de computação de um sistema. Utilizamos como arcabouço para o desenvolvimento do nosso trabalho as Máquinas de Estado Abstratas (ASMs) [Gurevich, 1995]. Fazemos uma revisão das ASMs, mostrando um breve histórico, uma introdução informal com exemplos, tendências de pesquisa e também a definição original do modelo.

Apresentamos uma nova definição formal para as ASMs introduzindo um sistema de tipos para a linguagem e demonstrando que a linguagem se torna fortemente tipada.

No corpo principal da tese, introduzimos um novo conceito no contexto das ASMs, as Máquinas de Estado Abstratas Interativas (*Interactive Abstract State Machines* - IASMs), para em seguida mostrar sua definição formal em função das ASMs. Mostramos que as IASMs podem ser reduzidas para as ASMs por meio de um esquema de tradução e demonstramos as propriedades que as IASMs preservam.

Mostramos também o uso das IASMs em três estudos de casos para validar o poder de especificação da linguagem e do mecanismo de raciocínio sobre as especificações.

Finalmente, avaliamos o contexto de atuação das IASMs fazendo uma comparação com outras abordagens relacionadas e mostrando as conclusões deste trabalho.

Sumário

Lista de Figuras	ii
1 Introdução	1
1.1 Motivação	2
1.2 O Uso de Métodos Formais	4
1.3 Problemas com os Métodos Formais	6
1.4 Plano da tese	8
2 Máquinas de Estado Abstratas	10
2.1 Introdução Informal	12
2.2 Tendências Atuais de Pesquisa	13
2.3 A Definição Original de ASM	13
2.3.1 O Estado Abstrato	14
2.3.2 Conceitos Preliminares	15
2.3.3 A Máquina Abstrata	17
2.4 Conclusão	24
3 Uma Definição de Máquinas de Estados Abstratas Tipadas	25
3.1 Uma Definição de ASMs Multi-agentes Tipadas	26
3.2 O Sistema de Tipos	32
3.2.1 Regras do Sistema de Tipos	33
3.2.2 Verificação de Tipos	38
3.2.3 Consistência Interna	39
3.3 Conclusão	41
4 Máquinas de Estados Abstratas Interativas	42
4.1 Regras de Entrada e Saída	49
4.2 Regras de Configuração da Topologia	50
4.3 Outras Regras de Interação	60
4.4 Regras de Computação Interna	60
4.5 Verificação Contextual de Interação	60
4.5.1 Regras de Verificação Contextual	61
4.6 Conclusão	64

5	Aplicações	66
5.1	Arquiteturas e Componentes de Software	66
5.1.1	Arquitetura e Componentes como Máquinas Interativas	68
5.2	Protocolo do Bit Alternante	73
5.2.1	Propriedades da Especificação do Protocolo AB	79
5.3	Sistemas Orientados por Conexão vs. Objetos Móveis	83
5.3.1	O Problema do Comitê de Programa	84
5.3.2	Propriedades da Especificação do Sistema de Comitê de Programa .	90
5.4	Conclusão	96
6	Trabalhos Relacionados	98
6.1	O π -Cálculo	98
6.2	O Modelo de <i>Actors</i>	100
6.3	O Cálculo de Ambientes	102
6.4	Modelos de Coordenação	103
6.5	Conclusão	104
7	Conclusões	106
	Referências Bibliográficas	109

Lista de Figuras

2.1	Gráfico exemplo	15
2.2	Diagrama de Fluxo ASM dos Dados Filósofos	23
3.1	Uma máquina para somar os membros de uma lista qualquer	38
3.2	Um trecho de verificação de soma	39
4.1	Diagrama de Fluxo ASM	45
4.2	<i>“Engenheiros de software se esforçam para ser engenheiros de verdade. Engenheiros de verdade usam matemática apropriada ao problema. Então, engenheiros de software deveriam usar matemática apropriada ao problema. Dado que, por definição, métodos formais representam a matemática de programas, então engenheiros de software devem utilizar métodos formais apropriados ao problema”</i> . Adaptado de [Holloway, 1997].	45
4.3	Regra de Desconstrução $R_{desconstrução}$	50
4.4	Regra de Desconstrução $R_{desconstrução}$	50
4.5	Julgamentos	61
4.6	Ambiente	62
4.7	Regras para Definição	64
4.8	Função Env: define o ambiente	64
4.9	Regras para Verificação de Intenção	65
4.10	Exemplo de especificação	66
4.11	Adicionando as Regras	66
5.1	Interação síncrona para chamada de serviço	69
5.2	Interação com lexemas	71
5.3	Comunicação entre dois clientes via um canal de serviço	71
5.4	Aproximação de Componentes	72
5.5	Unidades e o Fluxo de mensagens no Processo AB	73
5.6	Especificação Principal do Processo AB	74
5.7	Unidade Sender	75
5.8	Unidade Receiver	76

Lista de Figuras

2.1	Grafo exemplo	15
2.2	Especificação ASM dos <i>Dinning Philophers</i>	23
3.1	Uma máquina para somar os inteiros de uma lista qualquer	38
3.2	Um trecho da verificação de tipos	39
4.1	Estrutura de uma especificação IASM	45
4.2	Regra para instanciação estática	48
4.3	Sintaxe para Regras de Interação	49
4.4	Regra de Saída para $R_{out} \ t:c \rightarrow u$	51
4.5	Regra de Entrada R_{in} para $f(tt) \leftarrow u.c$	51
4.6	Regra de Conexão R_{conn1}	54
4.7	Regra de Conexão R_{conn2}	54
4.8	Regra de Conexão R_{conn3}	55
4.9	Regra de Desconexão $R_{disconn1}$	55
4.10	Regra de Desconexão $R_{disconn2}$	55
4.11	Julgamentos	61
4.12	Ambiente	62
4.13	Regras para Declarações	64
4.14	Função <i>Env</i> : define o ambiente	64
4.15	Regras para Verificação da Interação	65
4.16	Exemplo de especificação	66
4.17	Aplicando as Regras	64
5.1	Interação síncrona para chamada de serviço	69
5.2	Interação com terceiros	71
5.3	Comunicação entre dois clientes via um canal servidor	71
5.4	Agregação de Componentes	72
5.5	Unidades e o fluxo de mensagens no Protocolo AB.	73
5.6	Especificação Principal do Protocolo AB	74
5.7	Unidade <i>Sender</i>	75
5.8	Unidade <i>Receiver</i>	76

5.9	Unidade <i>Channel</i>	77
5.10	Unidade <i>Loose</i>	78
5.11	Unidade <i>ClientSender</i>	78
5.12	Unidade <i>ClientReceiver</i>	78
5.13	Especificação do Problema do Comitê de Programa	84
5.14	Unidade <i>Author</i>	86
5.15	Unidade <i>Submission</i>	87
5.16	Unidade <i>Chair</i>	88
5.17	Unidade <i>Reviewer</i> (Interação com <i>Chair</i>)	90
5.18	Unidade <i>Reviewer</i> (Interação com <i>Reviewer</i>)	91
5.19	Especificação Principal do Sistema de Comitê de Programa	92