

# SIMPÓSIO DOUTORAL 2006

## IDENTIFICAÇÃO

Doutorando (nome, e-mail)

*Marco Antonio de Castro Barbosa*

Título da Tese

*A Refinement Calculus for Software Components and Architectures*

Orientadores (nome, universidade, departamento, e-mail)

*Luís Soares Barbosa – Universidade do Minho – Departamento de Informática*

[lsb@di.uminho.pt](mailto:lsb@di.uminho.pt)

Data Início (oficial, efectiva)

01/03/2003

Data Término (prevista, efectiva)

03/03/2006

## RESUMO

Área de Investigação e Desenvolvimento (I&D)

*Fundamentos da Informática*

Resumo

*A tese em causa tem como tema “A Refinement Calculus for Software Components and Architectures”. O processo de refinamento de uma especificação de software pode ser definido como uma transformação de um desenho “abstrato” para outro mais “concreto”, incorporando, portanto, uma noção de implementação por substituições progressivas que preservam, de algum modo, a semântica do sistema. O objetivo desta tese é a definição de leis que permitam realizar o refinamento de componentes e arquiteturas de software de forma sistemática.*

Objectivos Estratégicos

- Suporte para o raciocínio no desenvolvimento de programas baseados em componentes (num macro-nível)*
- Suporte para o desenho de arquiteturas de software de maneira sistemática.*

## CONTRIBUIÇÕES

Principais contribuições técnico-científicas

- Cálculo de refinamento para componentes e arquiteturas de software;*
- Fornecer uma visão baseada em canais para a coordenação de componentes de software;*
- Biblioteca (PUR Camila) para coordenação de componentes;*

- *Customização de componentes. Estender o cálculo de componentes de forma a lidar com a especificação do comportamento pretendido que não é intrínseco ao próprio componente mas ao seu papel (uso) em uma situação particular.*

## Publicações

M. A. Barbosa and L. S. Barbosa. Towards a relational model for component interconnection. In R. Lins and C. Braga, editors, Proc. 8th Brazilian Symposium on Programming Languages, pages 17–30, Niteroi, Brasil, May 2004.

M. A. Barbosa and L. S. Barbosa. A Relational Model for Component Interconnection. Journal of Universal Computer Science, 10(7):808–823, July 2004.

The basic motivation of component based development is to replace conventional programming by the composition of reusable off-the-shelf units, externally coordinated through a network of connecting devices, to achieve a common goal. This paper introduces a new relational model for software connectors and discusses some preliminary work on its implementation in Haskell. The proposed model adopts a coordination point of view in order to deal with components' temporal and spatial decoupling and, therefore, to provide support for looser levels of inter-component dependency and effective external control.

M. A. Barbosa and L. S. Barbosa. Specifying Software Connectors. In K. Araki and Z. Liu, editors, Proc. First International Colloquium on Theoretical Aspects of Computing (ICTAC'04), Guiyang, China, pages 53–68. Springer Lect. Notes Comp. Sci. (3407), 2004.

Orchestrating software components, often from independent suppliers, became a central concern in software construction. Actually, as relevant as components themselves, are the ways in which they can be put together to interact and cooperate in order to achieve some common goal. Such is the role of the so-called software connectors: external coordination devices which ensure the flow of data and enforce synchronization constraints within a component's network. This paper introduces a new model for software connectors, based on relations extended in time, which aims to provide support for light inter-component dependency and effective external control.

Marco Antonio Barbosa. A refinement calculus for software components and architectures. In ESEC/FSE-13: Proceedings of the 10th European software engineering conference held jointly with 13th ACM SIGSOFT international symposium on Foundations of software engineering, pages 377–380, New York, NY, USA, 2005. ACM Press.

The complexity and ubiquity achieved by software in the present world makes it imperative, more than ever, the availability of both technologies and sound methods to drive its development. Programming “*in-the-large*”, component-based programming and software architecture are popular expressions which embody this concern and correspond to driving forces in current software engineering. In such a context, this paper reports on the research, which constitutes the PhD project of the author, on a formal calculus for reasoning about and transforming component-based architectures.

## ENQUADRAMENTO

### Motivação

*A programação em grande escala, exige novas técnicas e métodos para o seu raciocínio e desenvolvimento. Seguindo estas necessidades podemos citar as Arquiteturas de Software. Entretanto, esta abordagem é, em sua maioria, baseada em forma diagramática, i.é., caixas e linhas, o que corresponde a um tratamento informal ou, algumas vezes, semi-formal. O objetivo desta tese é fornecer suporte matemático, permitindo a especificação*

de arquiteturas de maneira rigorosa, permitindo assim o raciocínio sistemático sobre elas.

#### Trabalhos Alternativos

- Como modelo/linguagem para coordenação de componentes, pode-se considerar a linguagem Reo, desenvolvida por Arbab e seu grupo no CWI, Amsterdão, como o principal modelo de referência para este trabalho.

#### Bibliografia Principal

R. Allen and D. Garlan. A forma basis for architectural connection. *ACM TOSEM*, 6(3):213-249, 1997

R. Backhouse. An exploration of the Bird-Meertens formalism. CS 8810, Groningen, University, 1988. Also presented the Int. Sum. School on Constructive Algorithmics, Holum, Ameland, 1989.

R. C. Barckhouse, P. Janson, J. Jeuring, and L. Meertens. Generic Programming: An introduction. In S. D. Swierstra, P. R. Henriques, and J. N. Oliveira, editors, *Third International Summer School on Advanced Functional Programming, Braga*, pages 28-115. Springer Lect. Notes Comp. Sci. (1608), September 1998.

L. S. Barbosa. Componentes as processses: An exercise in coalgebraic modeling. In S. F. Smith and C. L. Talcott, editors, *FMOOD'2000 – Formal Methods for Open Object-Oriented Distributed Systems*, pages 397-4170. Kluwer Academic Publishers, September 2000.

L. S. Barbosa. *Componentes as Coalgebras*. PhD thesis, DI, Universidade do Minho, 2001.

L. S. Barbosa Process calculi à la Bird-Meertens. In *CMCS'01 – Workshop on Coalgebraic Methods in Computer Science*, pages 47-66, Genova, April 2001. ENTCS, volume 44.4, Elsevier.

L. S. Barbosa and J. N. Oliveira. State-based components made generic. Submitted for publication, 2002.

R. Bird and O. Moor. *The Algebra of Programming*. Series in Computer Science. Prentice-Hall International, 1997.

J. Fiadeiro and A. Lopes. Semantics of architectural connectors. In *Proc. of TAPSOFT'97*, pages 505-519. Springer Lect. Notes Comp. Sci. (1214), 1997.

M.M. Fokkinga, *Law and Order in Algorithmics*. PhD thesis, University of Twente, Dept INF, Enschede, The Netherlands, 1992.

D. Garlan and M. Shaw. An Introduction to Software architecture. In V. Ambriola and G. Tortora, editors, *Advances in software Engineering annd Knowledge Engineering (volume I)*. World Scientific Publishing Co., 1993.

- M. C. Hennessy. *Algebraic Theory of Processes*. Series in the Foundations of Computing MIT Press, 1988.
- C. A. R. Hoare. *Communicating Sequential Processes*. Series in Computer Science. Prentice-Hall International, 1985.
- B. Jacobs and J. Rutten. A tutorial on (co)algebras and (co)induction. *EATCS Bulletin*, 62:222-159, 1997.
- Cliff B. Jones. *Systematic Software Development Using VDM*. Series in Computer Science. Prentice-Hall International, 1986.
- B. Meyer and C. Mingins. Component-based development: From buzz to spark. *IEEE Computer*, 32(7):35-37, 1999.
- A. J. R. G. Milner. *Communication and Concurrency*. Series in Computer Science. Prentice-Hall International, 1989.
- M. Moriconi and X. Qian. Correctness and composition of software architectures. In *Second Symposium on Foundations of Software Engineering*, pages 164-174. ACM Press, 1994.
- O. Nierstrasz and L. Dami. Computer-oriented software technology. in O. Nierstrasz and D. Tschritzis, editors, *Object-Oriented Software Composition*, pages 3-28. Prentice-Hall International, 1995.
- J. Rutten. Universal coalgebra: A theory of systems. *Theor. Comp. Sci.*, 249(1):3-80, 2000. (Revised version of CWI Techn. Rep. CS-R9652, 1996).
- J. Samentinger. *Software Engineering with Reusable Components*. Springer-Verlag 1997.
- J.-G. Schneider and O. Nierstrasz. Components, scripts, glue. In L. Barroca, J. Hall, and P. Hall, editors, *Software Architectures – Advances and Applications*, pages 13-25. Springer-Verlag, 1999.
- J. M. Spivey, *The Z Notation: A Reference Manual*. Series in Computer Science. Prentice-Hall International, 1989.
- C. Szyperski. *Component Software, Beyond Object-Oriented Programming*. Addison-Wesley, 1998.
- D. Turi and J. Rutten. On the foundation of final coalgebra semantics: non-well-founded sets, partial orders, metric spaces. *Mathematical Structures in Computer Science*, 8(5):481-540, 1998.
- P. Wadler and K. Weihe, Component-based programming under different paradigms. Technical report, Report on the Dagstuhl Seminar 99081, February 1999.

## DESENVOLVIMENTO

	1º Tri	2º Tri	3º Tri	4º Tri
1º A n o	- Estudos de background	- Estudos de background	- Desenvolvimento da Biblioteca Haskell; - especificação de modelo para conectores de software	- Desenvolvimento da Biblioteca Haskell; - especificação de modelo para conectores de software
2º A n o	- Especificação de modelo para conectores de software baseado em Relações	- Estudo de leis de refinamento e extensão do modelo relacional	- Extensão coalgébrica do modelo relacional	- investigação de leis subjacentes aos modelos desenvolvidos
3º A n o	- Investigação de noção mobilidade nos modelos desenvolvidos	- Investigação de noção mobilidade nos modelos desenvolvidos	- Biblioteca de coordenação na plataforma PRe Camila - Repositório de padrões arquiteturais	- Biblioteca de coordenação na plataforma PRe Camila - Repositório de padrões arquiteturais
4º A n o	- Consolidação das leis de refinamento; - Conclusão Biblioteca PReCamila - Conclusão Repositório de padrões	- Consolidação das leis de refinamento; - Conclusão Biblioteca PReCamila - Conclusão Repositório de padrões	Escrita/fechamento	Escrita/fechamento

## AVALIAÇÃO

### Análise Comparativa

*Como anteriormente citado, a linguagem Reo, constitui-se no principal modelo de comparação a este projeto. Entretanto, alguns dos aspectos a serem tratados nesta tese, como dinamismo e mobilidade, não são tratados até o momento em Reo, e de acordo com Arbab não há previsão de que será tratado a curto prazo.*

### Auto-avaliação do Trabalho Realizado

*- O trabalho realizado provê suporte para a concretização do objetivo central da tese, entretanto, o cerne do trabalho ainda não foi atingido pois demanda maior investigação no tópico mobilidade e na concretização das leis de refinamento que constituem o foco central da tese. A questão da mobilidade fora apontada na proposta de tese a nível secundário porém, adquiriu ao longo destes estudos uma importância maior. Desta forma, optou-se por dedicar um pouco mais do tempo inicialmente previsto à questão da mobilidade. Sabidamente, este é um tópico de tratamento difícil e por esta razão o cronograma original teve que ser reformulado. Outra alteração ao projeto inicial foi da integração da biblioteca a ser desenvolvida na linguagem Haskell à linguagem de especificação PRe Camila, por esta ser uma linguagem desenvolvida (e em reformulação) pelo grupo de LMF. Desta forma, a presente tese traria alguma contribuição a um projeto em comum do referido grupo.*

### Auto-avaliação da Documentação Produzida

*- As publicações aceitas até o momento corroboraram com os objetivos centrais da tese proposta. As publicações tiveram bons comentários tanto quando do seu processo de avaliação quanto durante suas apresentações nos eventos a qual se destinaram. Mesmo as publicações que foram rejeitadas*

*foram indicativo de que o rumo a ser tomado na concretização da tese está em bom caminho. Os comentários foram favoráveis às idéias apresentados porém os trabalhos ainda careciam de maior solificação e resultados mais conclusivos para poderem vir a ser publicações científicas.*

## **DIFICULDADES**

Dificuldades Técnico-Científicas

- *Imersão em nova área científica;*
- *Poucos trabalhos relacionados que permitissem uma avaliação mais precisa dos resultados a serem obtidos ou mesmo um confronto dos resultados para melhor validar a tese proposta (sobretudo a questão da mobilidade).*