

1 Introdução

A tecnologia de agentes de software é considerada como adequada para lidar com a heterogeneidade, adaptabilidade e a característica aberta de alguns sistemas distribuídos (Hannoun, Boissier et al. 2000; Dignum and Dignum 2001; DeLoach and Matson 2004). Em sistemas distribuídos abertos, os agentes estão imersos em um ambiente altamente imprevisível, podem ser não-cooperativos e, freqüentemente, os outros agentes que compõem o sistema não são conhecidos a priori. Pode-se dizer que sistemas abertos são concorrentes, uma vez que são compostos por diferentes subsistemas operando geograficamente distribuídos e utilizam-se de diferentes fontes de informações; assíncronos; possuem controle descentralizado, no sentido de que o fluxo de execução do sistema é regido por decisões locais dos subsistemas; e podem conter informação inconsistente (Hewitt 1986). O comportamento global deste tipo de sistema emerge da interação entre os agentes e, portanto, não pode ser totalmente previsto. Embora este tipo de comportamento seja aceitável ou mesmo desejável, em algumas aplicações, isto pode levar a falhas no sistema. Isto é especialmente verdade em domínios tais como comércio eletrônico (Guttman, Moukas et al. 1998; Ripper, Fontoura et al. 2000), cadeia de fornecimento (Michael, Alberto et al. 2006) e aplicações médicas (Lapinsky, Weshler et al. 2004).

Muitos pesquisadores utilizam a idéia de instituições eletrônicas para alcançar níveis maiores de previsibilidade e confiabilidade nas organizações de agentes. Desta forma, a natureza aberta dos sistemas multi-agentes pode ser melhor gerenciada. Instituições eletrônicas são uma metáfora baseada nas instituições que existem na sociedade humana. Estas instituições são estabelecidas para regular as interações entre as partes que estão realizando alguma atividade. Uma vez que as interações são especificadas pelas instituições eletrônicas, é preciso que haja uma infra-estrutura de software que garanta que os agentes que fazem parte da instituição interajam de acordo com as especificações. Desta forma, é possível alcançar níveis maiores de previsibilidade e, conseqüentemente,

diminuir a probabilidade de falhas no sistema. Neste contexto, vários modelos de instituições eletrônicas foram propostos (Hannoun, Boissier et al. 2000; Minsky and Ungureanu 2000; Dignum 2002; Esteva 2003; Esteva, Rosell et al. 2004).

O termo organização de agentes também é bastante utilizado na literatura. Uma organização de agentes pode ser definida como um padrão de cooperação emergente ou pré-definida entre os agentes de um sistema. Este padrão pode ser definido pelo projetista do software ou pelos próprios agentes (Boissier, Hübner et al. 2006).

Neste trabalho, utiliza-se o termo governança para reunir o conjunto de abordagens que objetivam estabelecer e verificar alguma forma de estrutura, conjunto de normas ou convenções de comportamento para articular ou restringir as interações dos agentes. Desta forma, as instituições eletrônicas e alguns tipos de organizações são consideradas como uma abordagem de governança.

Por outro lado, artigos recentes enfatizam que software continua sofrendo pela falta de qualidade (Kan 2002; Noushin 2003; Mahmood, David et al. 2005). Sabe-se que cerca de 50% do software tornado disponível contém falhas não triviais (Boehm and Basili 2001). Ainda assim, sabe-se que, independentemente do rigor com que seja desenvolvido, todo software conterà falhas. Conseqüentemente, a ocorrência de falhas de uso, de hardware ou do próprio software é um fato com o qual se precisa conviver sem que, no entanto, estas falhas levem a um nível de danos inaceitável.

Este problema na produção de software ocorre mesmo quando se considera software com arquiteturas convencionais. A tendência disto é se agravar cada vez mais já que o software é desenvolvido por equipes geograficamente distribuídas e operam de forma distribuída (várias CPUs contendo partes heterogêneas do software operando sobre uma única aplicação, por exemplo em sistemas grid). Outra razão para que o problema se torne um desafio crescente é o fato dos sistemas de software tornarem-se cada vez mais complexos (volume e abrangência da funcionalidade e requisitos de qualidade mais restritivos), precisarem estar disponíveis por mais tempo (sistemas 24h / 7 dias por semana) e serem utilizados por pessoas sem treinamento.

Em virtude da crescente participação de software na sociedade, torna-se cada vez mais necessário assegurar que ele seja fidedigno. Desenvolver software

fidedigno é bem mais amplo do que incorporar tolerância a faltas¹ e assegurar corretude, entre outras propriedades. O entendimento da noção de fidedignidade² depende do serviço que o software oferece. Em alguns serviços como o comando e controle, uma falha pode ter conseqüências catastróficas. Já em outros contextos, como por exemplo a busca de informação na Web, um resultado incorreto (falsos positivos ou ausência de respostas relevantes) é tolerável desde que isto ocorra com uma freqüência baixa. Entretanto, não é tolerável que um sistema crítico interrompa a sua execução, a invasão de privacidade por programas maliciosos, ou ainda um aplicativo apresentar riscos à segurança de quem o usa. Ou seja, um software é dito fidedigno quando se pode confiar no mesmo através de verificações formais ou informais assumindo riscos de danos compatíveis com o serviço prestado pelo software (Avizienis, Laprie et al. 2004).

A hipótese apresentada neste trabalho é que a incorporação de mecanismos de fidedignidade em uma abordagem de governança baseada em leis de interação pode ser utilizada para a construção de sistemas multi-agentes abertos com um nível de fidedignidade adequado. Esta hipótese foi baseada na percepção de que as abordagens de governança atuais focam na especificação e verificação do comportamento esperado em um sistema. Entretanto, atributos de fidedignidade como tolerância a faltas não são levados em consideração. A incorporação de aspectos de fidedignidade em uma abordagem de governança tem como principal benefício a geração de uma tecnologia integrada que possui as vantagens de uma abordagem de governança e ao mesmo tempo lida com especificações de instrumentos para alcançar maiores níveis de fidedignidade.

Desta forma, a principal contribuição deste trabalho situa-se na linha de pesquisa de governança de sistemas multi-agentes abertos. A contribuição ocorre através da disponibilização de uma abordagem de governança com os objetivos de ser (i) flexível para dar suporte a evoluções na própria abordagem; (ii) permitir agregar e expressar os conceitos relacionados a governança encontrados na literatura atual; (iii) possuir um alto nível de abstração de seus elementos e ainda

¹ A presença de faltas pode levar a ocorrência de falhas no sistema. Uma definição mais precisa sobre estes termos será apresentada no capítulo 5.

² A palavra fidedignidade foi a tradução encontrada pelo Prof. Arndt Von Staa Staa, A. v. (2006). Engenharia de Software Fidedigno. Monografias em Ciência da Computação, n. 13/06. PUC-Rio. Rio de Janeiro, Pontifícia Universidade Católica do Rio de Janeiro. para o termo inglês *dependability*.

assim viabilizar o mapeamento da especificação para um mecanismo de implementação; e (iv) permitir a incorporação de aspectos de fidedignidade na especificação das leis. Uma vez desenvolvida a abordagem, mostra-se como a sua aplicação na construção de sistemas multi-agentes pode auxiliar a construção de sistemas que incorporam técnicas de fidedignidade.

1.1. Problema

O principal problema abordado neste trabalho é a falta de garantia que o resultado observável resultante da interação entre os agentes de um sistema multi-agente aberto ocorra de acordo com o especificado e que o projetista do sistema possua ferramentas para lidar com atributos de fidedignidade. Mais especificamente, existe uma carência para o estabelecimento de instrumentos de governança que permitam que sistemas multi-agentes abertos adotem atributos de fidedignidade, considerando as seguintes hipóteses:

- (i) os detalhes da arquitetura e a implementação dos agentes são inacessíveis;
- (ii) os agentes interagem através de troca de mensagens;
- (iii) é possível especificar quais são as interações e comportamentos esperados do sistema a priori.

1.2. Limitações de Abordagens Existentes

As abordagens atuais de governança que poderiam ser utilizadas para este propósito (i) dispõem de um conjunto restrito de entidades que permitem somente a especificação de leis em um nível de abstração muito baixo, contendo primitivas muito elementares; o que aumenta de forma significativa o esforço de desenvolvimento e manutenção (Minsky 2005); (ii) ou são de um nível de abstração tão alto que não permitem o mapeamento automático da especificação para um mecanismo de implementação (Dignum 2002); (iii) ou possuem um conjunto de elementos adequados, porém baseado em modelos conceituais que tornam fixa a forma de composição entre os seus elementos e não prevêem mecanismos para evoluir o próprio modelo conceitual, o que dificulta a adaptação da abordagem para evoluções tanto das leis quanto da abordagem em si (Esteve 2003; Hübner 2003). É importante destacar que a limitação relacionada ao item

(iii) não se dá pela inflexibilidade ou pela falta de expressividade dos elementos que compõem seus modelos conceituais, mas sim pela própria estrutura de composição acoplada entre esses elementos. Suponha, por exemplo, que se deseje incluir um elemento de tratamento de exceções em um destes modelos conceituais; isso não seria possível sem uma revisão estrutural em todos os elementos que seriam afetados pela entrada deste novo conceito. Finalmente, (iv) não foram encontradas abordagens que incorporem preocupações de fidedignidade.

1.3. Solução Proposta

Diante do problema especificado na Seção 1.1, nesta tese propõe-se uma abordagem de governança original para que leis de interação de um sistema multi-agente aberto possam ser especificadas, para que depois seja possível verificar se as interações estão seguindo a especificação e caso não estejam, ações corretivas possam ser tomadas. A abordagem proposta aborda as limitações identificadas na Seção 1.2 da seguinte forma:

Em relação ao item (i), parte dos elementos que compõem o meta-modelo são abstrações que fazem parte do estado da arte das pesquisas em governança (Paes, Carvalho et al. 2007). Outros elementos foram introduzidos conforme a necessidade dos experimentos realizados e relatados em (Carvalho, Paes et al. 2005; Carvalho, Paes et al. 2005; Paes 2005; Rodrigues, Carvalho et al. 2005; Carvalho, Almeida et al. 2006; Gatti, Carvalho et al. 2006; Gatti, Lucena et al. 2006; Gatti, Paes et al. 2006). Isto permitiu a criação de um meta-modelo composto por entidades de alto nível de abstração e ainda assim com semântica suficientemente bem-definida para que seja possível especificar leis baseadas nestas abstrações e principalmente verificar em tempo de execução de um sistema se as leis estão sendo cumpridas, o que resolve o problema do item (ii).

Em relação a (iii), o meta-modelo é baseado em um modelo de eventos que permite fraco acoplamento entre as entidades do próprio modelo conceitual, refletindo-se em flexibilidade para acomodar modificações e expressividade para compor os elementos do modelo. O mecanismo de eventos também é mapeado para o nível de implementação em um *middleware* baseado em componentes, no qual os componentes se comunicam principalmente através de eventos.

Finalmente, em relação ao item (iv), este trabalho ilustra como duas técnicas de fidedignidade (tolerância a faltas e *dependability explicit computing*) podem ser aplicadas utilizando a abordagem proposta.

1.4. Contribuições

São contribuições desse trabalho³:

- (i) um meta-modelo de alto nível baseado em eventos para o domínio de leis de interação;
- (ii) uma notação gráfica que permite representar os elementos do meta-modelo;
- (iii) uma linguagem de representação declarativa que permite a especificação do meta-modelo. Esta linguagem é uma evolução do XMLaw (Paes 2005; Paes, Carvalho et al. 2005) e utiliza-se de analogias para os conceitos de programação por convenção (do inglês *coding by convention*) de projetos como Rails (Thomas, Hansson et al. 2006) e Grails (Rocher 2006) para aumentar a produtividade e a simplicidade das especificações das leis.
- (iv) um middleware capaz de monitorar interações em um sistema multi-agente e de interpretar as especificações das leis para verificar se elas estão sendo efetivamente cumpridas;
- (v) Ilustração da adoção de estratégias e conceitos da área de fidedignidade de forma integrada a abordagem proposta de utilização de leis de interação em sistemas multi-agentes abertos. A incorporação de preocupações de fidedignidade em uma abordagem de governança é certamente a principal contribuição deste trabalho;
- (vi) Um estudo de caso realista que ilustra que a abordagem de leis pode ser aplicada para construir sistemas complexos mais confiáveis.

³ Os itens de (i) a (iv) são referidos como a abordagem proposta.

1.5. Organização da Tese

Capítulo 2 - O Meta-Modelo de Leis: XMLaw	Neste capítulo, apresenta-se o meta-modelo de leis utilizado pela abordagem. O meta-modelo descreve todos os elementos usados para especificar uma lei. É apresentada a semântica, o relacionamento e a forma de representação em uma linguagem declarativa de cada um dos elementos.
Capítulo 3 - Trabalhos Relacionados ao Meta-Modelo	Neste capítulo, apresentam-se as características do meta-modelo do XMLaw que o diferencia dos modelos atuais. Discute-se a importância de se possuir elementos com alto nível de abstração e da flexibilidade para acomodar mudanças no modelo e na especificação.
Capítulo 4 - Infra-estrutura de Implementação: M-Law	Neste capítulo, apresenta-se um middleware baseado em um modelo de componentes que dá suporte ao monitoramento da interação entre os agentes com objetivo de verificar se as especificações das leis estão sendo seguidas.
Capítulo 5 - Fidedignidade	Neste capítulo, é feito um levantamento bibliográfico que apresenta uma taxonomia que classifica os vários aspectos do conceito de fidedignidade e discute-se como a abordagem de governança pode contribuir para alcançá-los.
Capítulo 6 - Incorporação de Fidedignidade na Abordagem de Governança	Neste capítulo, apresenta-se como a fidedignidade pode ser incorporada na abordagem de governança proposta. Mostra-se como implementar as estratégias <i>Full Forward Recovery</i> e <i>Partial Forward Recovery</i> . Apresenta-se também o conceito de <i>dependability explicit computing</i> e uma implementação para ilustrar como realizar <i>dependability explicit</i>

	<i>computing</i> utilizando a abordagem proposta
Capítulo 7 - Estudo de Caso: Controle de Tráfego Aéreo	Neste capítulo, apresenta-se um estudo de caso que ilustra a aplicação de leis. Discute-se também, como as leis estão ajudando a alcançar um maior grau de fidedignidade para este exemplo.
Capítulo 8 - Conclusões	Neste capítulo, são apresentadas reflexões sobre o trabalho realizado e discussões sobre pontos em aberto e trabalhos futuros.