

# 1 Introdução

## 1.1 Motivação

Pesquisadores de Engenharia de Software têm apresentado, nos últimos anos, muitos trabalhos sobre o desenvolvimento de Sistemas Multi-Agentes [DeLoach et al. 01], [Yu 01b], [Castro et al. 02], [Cysneiros & Yu 03a], [Silva & Lucena 04], [Choren & Lucena 04], [Garcia et al. 04]. Esse grande número de pesquisas se deve ao fato de o desenvolvimento de Sistemas Multi-Agentes (SMA) ser considerado uma abordagem promissora. Sistemas Multi-Agentes trabalham fundamentalmente com agentes, os quais são sistemas de software que representam os atores do contexto de uma organização. Yu define: “*Agente é um ator com manifestações físicas concretas, como um ser humano*” [Yu 95]. Agentes possuem propriedades, eles são: autônomos, pró-ativos e sociais; colaboram uns com os outros, interagem entre si e se comunicam; têm capacidade de aprender, de se adaptar e devem estar aptos a mudar de ambiente [Wooldridge 02]. O desenvolvimento de Sistemas Multi-Agentes, “Multi-Agent Systems” – MAS, tem como objetivo vencer o desafio de lidar com a alta complexidade de sistemas de software (numerosas metas de diferentes agentes) em uma tecnologia da Internet que é naturalmente “WEB networked”.

De modo a solucionar a demanda por sistemas que trabalham para atingir diversas e numerosas metas (“goals”), a modelagem de Sistemas Multi-Agentes necessita usar conceitos e metáforas que espelhem a maneira da qual o mundo organizacional precisa ser compreendido. Desse modo, a Engenharia de Requisitos (ER) [Kotonya 98] deve suprir o desenvolvimento de SMA com métodos, técnicas e ferramentas para lidar com o alto nível de abstração envolvido a respeito de como o software deve se comportar e quais requisitos devem ser considerados [Oliveira 06a].

Sabe-se que, para se aproximar mais do sucesso, sistemas de software devem atender as metas dos atores da organização. As metas nas quais as decisões e ações dos atores se baseiam refletem a intencionalidade dos atores. O conceito de intencionalidade, que significa motivações e interesses dos atores da organização, foi definido por Yu [Yu 95] e é amplamente utilizado em trabalhos de pesquisa como Jureta et al. [Jureta 06]. Importantes pesquisas promovem a intencionalidade, que pode ser

observada em diversas abordagens que usam metas como núcleo para métodos, modelos e ferramentas: KAOS [Lamsweerde 01], i\* Framework [Yu 95], GBRAM [Anton 97], Tropos [Castro 02] e [Bresciani 04], e NFR Framework [Chung 00] são exemplos. Lamsweerde classifica esses métodos como pertencentes à Engenharia de Requisitos Orientada a Metas - “GORE (Goal-Oriented Requirements Engineering)”. A orientação a metas desempenha um papel importante no processo de Engenharia de Requisitos [Dardenne 93], especialmente “softgoals” [Lamsweerde 98]. Um sem-número de pesquisadores considera GORE como um dos melhores caminhos para produzir software de qualidade [Kavakli 05].

Kavakli e Loucopoulos, em [Kavakli 03], mostram que o conceito de metas tem sido aplicado na Engenharia de Requisitos com diferentes finalidades e que um grande problema com o uso de metas ocorre na maneira que elas são comunicadas (na forma). No nosso entendimento, metas cobrem duas dimensões, simultâneas e inseparáveis: a forma (ou representação) e o conteúdo (ou significado). Metas são abstratas por natureza [Kavakli 05] e por isso pode ser difícil para muitas pessoas entender o significado e o poder da abordagem GORE.

Utilizando uma perspectiva bem abstrata, nossa pesquisa deseja contribuir para a resposta às perguntas: Por que e para que trabalhar com intencionalidade? Como seria possível descobrir e modelar a intencionalidade dos atores de uma organização que terá a gestão apoiada por um sistema de software?

## 1.2 Objetivo da Pesquisa

Agentes são, por natureza, orientados a metas. Por esse motivo o paradigma de Sistemas Multi-Agentes tem forte dependência com a intencionalidade. Dar ênfase à intencionalidade para atender o paradigma de Sistemas Multi-Agentes é o desafio para a Engenharia de Requisitos. O engenheiro de requisitos precisa entender, pensar e atuar para melhor explorar os conceitos do paradigma da orientação a agentes [Oliveira 06a] e, indo mais além, segundo Leite [Leite 95] o processo de requisitos requer as atividades de elicitar, modelar e analisar os requisitos e, por isso, para o sucesso da captura da intencionalidade dos atores, o engenheiro de requisitos necessita (a) elicitar a intencionalidade através de metas (“goals”), compromissos (“commitments”), crenças (“beliefs”) e habilidades (“abilities”) dos atores do contexto organizacional, (b) retratar

a intencionalidade através da modelagem dos requisitos e (c) analisar os modelos de requisitos produzidos.

Uma importante lacuna na abordagem GORE reside no fato de que aqueles métodos não tratam precisamente da elicitação da intencionalidade. Todos, sem exceção, preocupam-se com a modelagem [Oliveira 08a]. O método GBRAM [Anton 97], por exemplo, usa a estratégia de obter as metas em documentos que definem a missão da organização e depois essas metas são decompostas pelo engenheiro de requisitos. O Framework KAOS [Lamsweerde 01] recomenda que metas sejam obtidas de documentos que utilizam palavras-chave intencionais tais como: “objetivo”, “finalidade”, “intenção”, “preocupação”, “na direção de”, etc. O Framework i\* [Yu 95] (“i-estrela”), e conseqüentemente TROPOS [Castro 02], que usa conceitos do Framework i\*, lidam apenas com a modelagem da intencionalidade. Em todos esses métodos, a elicitação fica exclusivamente a cargo da competência do engenheiro de requisitos.

Adotamos as traduções: **meta concreta** para “goal” e **meta flexível** para “softgoal” porque essas palavras têm a representação adequada para o objetivo do nosso trabalho. A palavra meta, para a área de Administração, possui a definição de ser um objetivo quantificado [Drucker 06]. Moura em [Moura 06] define meta como: “Ponto ou objetivo a ser atingido em determinada medida e prazo. Enquanto o objetivo apenas explicita o propósito, intenção ou fim que se deseja alcançar, a meta quantifica e define um prazo. Uma meta é um objetivo quantificado a ser atingido dentro de um prazo especificado”, por isso, para o método **Eri\*c**, escolhemos meta concreta como a tradução de “goal”. Para a Engenharia de Requisitos, o uso da palavra “goal” significa que o método é “goal oriented”, ou seja, trabalha na captura dos requisitos do sistema priorizando a elicitação do porquê (“why”) antes de trabalhar e definir, através dos modelos, o que (“what”) o software deverá fazer.

Este trabalho defende a idéia de que para ser bem-sucedido o processo de requisitos de SMA deve ter a intencionalidade como viga-mestre. Por isso, esta tese contribui com um método que: fornece uma estratégia simples e “bottom-up”, baseada no Léxico Ampliado da Linguagem - LAL [Leite 93], para a elicitação das metas (“goals”) para sistemas multi-agentes; e promove dois construtores canônicos: o “SRconstruct” e a “SDsituation”, oriundos do Modelo SR e do Modelo SD do Framework i\* [Yu 95]. O SRconstruct [Oliveira 08b] representa e trata local e exclusivamente a intencionalidade própria de cada agente e a SDsituation [Oliveira 06b]

particiona os modelos  $i^*$  com base na seqüência temporal das situações de dependência estratégica (Strategic Dependency Situations - SDSituations). A estratégia usada no método, que resulta no particionamento dos Modelos SD e SR, guiado por SDSituations [Oliveira 07], visa contribuir para a redução do problema da “escalabilidade”, enfatizado por Pastor [Pastor 06] e Estrada [Estrada 06], que freqüentemente aparece nos modelos  $i^*$ .

Para o método “Engenharia de Requisitos Intencional” - **ERi\*c**, foi escolhido o Léxico Ampliado da Linguagem (LAL) [Leite 93] como ferramenta para impulsionar a elicitación da intencionalidade, porque o LAL promove a captura das boas sugestões para se chegar à elicitación das metas. O LAL elicita os elementos do vocabulário da aplicação e os classifica como: sujeito, aquele que executa uma ação; objeto, aquele que sofre a ação; verbo, que representa a própria ação; e estado, que é uma situação.

Escolhemos o Framework  $i^*$  [Yu 95] para a modelagem dos requisitos devido aos seguintes motivos. Primeiro porque o Framework  $i^*$  já é centrado na intencionalidade, e segundo porque é (bem) reconhecido: o  $i^*$  já deu origem a outros dois métodos GORE (GRL <<http://www.cs.toronto.edu/km/GRL/>> e TROPOS [Castro 02] e [Bresciani 04]). Além disso, o  $i^*$  lida com a intencionalidade de forma mais completa, através da idéia da dependência estratégica entre os atores, e também é um dos frameworks mais referenciados e discutidos da atualidade.

Completando, para o método foi escolhida a Técnica de Cenários [Leite 00] para a especificação dos requisitos porque essa não tem viés de implementação, não é centrada na operacionalização dos requisitos, ou melhor, considera a organização de um modo abrangente e independente do uso de computadores ou da mecanização, além de ser também reconhecida por diversas pesquisas como [Carroll 94], [Hsia 94], [Zorman 95], [Kazman 96], [Breitman 98] [Leite 00], [Carroll 00].

A contribuição fundamental desse trabalho pode ser resumida em três partes: um processo bem delineado de construção, com uma técnica para a elicitación de metas e uma estratégia de modelagem que possibilita a modularização dos modelos SD e SR do Framework  $i^*$ .

### 1.3 Organização da Tese

A tese está organizada da seguinte forma: no capítulo 2 descrevemos as técnicas que o método proposto utiliza. Na seção 2.1, o Léxico Ampliado da Linguagem, a

Técnica de Cenários e o “i\* Modeling Framework”. Na seção 2.2, primeiro apresentamos a conceituação de “ação concreta e de ação flexível” e depois descrevemos os dois construtores canônicos auxiliares do método: o construto SDSituation – Situação de Dependência Estratégica e o construto SRconstruct – Construto de Razão Estratégica, o Painel da Intencionalidade – “IP Diagram” e, concluindo a seção, apresentamos a técnica de análise de modelos i\*: o “i\* Diagnoses” – Diagnósticos i\*. No capítulo 3 demonstramos o método proposto e mencionamos as contribuições trazidas por cada uma das etapas. No capítulo 4 expomos três exemplos, dos quais dois foram publicados em artigos. Os exemplos são os estudos de caso: “Controle de Caixa do Restaurante”, o “Expert Committee” [Oliveira 06a], [Oliveira 08a] e a “Seguradora Imperial” [Oliveira 07], [Oliveira 08b]. No capítulo 5 abordamos o processo de experimentação aplicado com a ajuda de alunos da graduação da PUC-Rio e da UERJ. Finalmente, no capítulo 6 concluímos focalizando também nos possíveis trabalhos após a conclusão da tese.