

6 Conclusão

Este capítulo tem como objetivo fazer a conclusão da tese. Ele apresenta, na primeira seção, uma descrição condensada do Método ERI*c, depois faz uma avaliação da aplicação dos três estudos de caso desenvolvidos e, em seguida, discute o resultado de três experimentos aos quais o método foi submetido. A segunda seção faz um retrospecto das principais contribuições da tese e as compara a dois métodos GORE, que se relacionam ao nosso trabalho, são eles: GBRAM (Goal Based Requirements Analysis Method) e KAOS (Knowledge Acquisition in autOMated Specification). A última seção discorre sobre as limitações do método proposto, sobre trabalhos futuros que possam alavancar o uso do método e sobre os desdobramentos vislumbrados para o Método ERI*c.

6.1 Resumo e contextualização

Esta tese apresentou o método “**ERI*c** - Engenharia de Requisitos Intencional”. O método foi divulgado através de artigos em conferências e workshops da área de engenharia de software. As conferências de que participamos foram: CASCON 06, WER 06, WER 07, SCCC 07 e CAiSE’08 (aceito para publicação). Os artigos divulgaram o método através de estudos de casos e em mais de uma oportunidade salientamos a necessidade da experimentação de cada subparte proposta.

A recomendação da experimentação foi considerada durante o planejamento das atividades para a conclusão da tese. Por isso foram executados três experimentos para aprimorar as heurísticas do método, e para avaliar e testar a “capacidade de trabalho” do mesmo. O método foi avaliado através de “laboratórios” e ainda o será através de projetos finais de curso de alunos da graduação universitária.

6.1.1 O Método ERI*c

O Método ERI*c “começa do princípio”. Ele tem início com a elicitación das metas (concretas e flexíveis) que envolvem os atores de uma organização. Por isso, dizemos que ele é um método “goal oriented” para a engenharia de requisitos.

No Método ERI*c a intencionalidade é descoberta por uma idéia simples: “AÇÕES MUDAM ESTADOS E ESTADOS SÃO METAS”. As ações que permeiam uma organização são expressas pelo léxico da linguagem e podem ser elicitadas através do Léxico Ampliado da Linguagem (LAL) [Leite 93].

As metas, depois de elicitadas, refinadas e organizadas, são arranjadas de forma coesa em agrupamentos para o alcance de metas organizacionais de mais alto nível. As metas dos atores cooperam entre si para preencher uma Situação de Dependência Estratégica, uma SDsituation.

As metas (concretas e flexíveis) estão interligadas dentro de SDsituations através de dependências, correlações e contribuições e são modeladas em Painéis de Intencionalidade (Diagramas IP). Os Diagramas IP retratam metas e as suas inter-relações, permitindo uma percepção exclusivamente intencional da organização.

Com a intencionalidade mapeada, a funcionalidade (com tarefas, subtarefas e recursos que apoiarão as metas a serem alcançadas) poderá ser definida. O novo sistema de software fica representado através de modelos i* (modelos SD e modelos SR), organizados por Situações de Dependência Estratégica - SDsituations.

Esses modelos poderão então ser especificados pelo engenheiro de requisitos com a aplicação da reconhecida Técnica de Cenários.

Na última etapa do trabalho, a organização já modelada e especificada terá seus modelos SD e SR submetidos a uma verificação de qualidade. Eles serão analisados seguindo critérios de completeza, eficácia e eficiência através de perguntas pré-elaboradas pela técnica de diagnósticos de modelos i*.

6.1.2 Avaliação dos estudos de casos

Os três exemplos elaborados nesta tese forneceram contribuições distintas e relevantes para o desenvolvimento, para a confirmação e para o aprimoramento dos procedimentos e das heurísticas recomendadas pelo método. As melhores contribuições aconteceram nas etapas do processo de elicitação da intencionalidade.

O “EC System” (“The Expert Committee System”) [Deloach 01], sistema de apoio para a organização de uma conferência, foi nosso primeiro estudo de caso. Constatamos na primeira modelagem, em 2005, que sem uma sistematização o processo de construir modelos i* era “penoso”. Esse estudo de caso forneceu a primeira versão do método, o qual tinha vantagens em relação aos procedimentos anteriores.

Para o caso do Controle do Caixa do Restaurante fizemos uma tentativa muito gratificante, utilizamos um léxico preparado vários anos antes por outro autor. Com este caso aprimoramos o método de elicitación de metas (concretas e flexíveis).

O estudo de caso da Companhia de Seguros – A Seguradora Imperial confirmou com mais uma evidência que os procedimentos do Método ERi*c funcionavam. Esse estudo de caso foi importante porque nos baseamos em fontes de informação distintas e que não foram elaboradas por nós para criarmos o Léxico Ampliado da Aplicação (LAL). As etapas e atividades foram executadas praticamente da mesma forma de um sistema real. O estudo de caso da Seguradora Imperial foi o maior dos três exercícios. Uma diferenciação deste para os outros casos foi que ele nos trouxe alternativas para a modelagem dos requisitos do negócio, problema que não tínhamos enfrentado nos dois estudos de caso anteriores.

6.1.3 O resultado dos experimentos

As experimentações contribuíram, primeiramente, para mostrar que o Método ERi*c possuía falhas na definição e na orientação da maioria das atividades. Diversas competências foram revistas e redefinidas para corrigir as falhas. Exemplos singelos foram também acrescentados ao material durante o treinamento do Método ERi*c.

O terceiro experimento teve o objetivo um pouco diferente em relação aos anteriores. Os alunos usaram a estratégia chamada de Diagnósticos i* para encontrar, em modelos SD e SR, itens a serem corrigidos e / ou melhorados. Resultado: com a aplicação do procedimento de perguntas da estratégia de Diagnósticos i*, mais da metade dos alunos conseguiu identificar 70% dos problemas latentes no exercício.

O exercício da experimentação foi significativamente vantajoso para o Método ERi*c. Queremos destacar que o método foi usado por “engenheiros de requisitos” em laboratórios de sala de aula, com resultados que nos encorajam a investir mais no aprimoramento do mesmo. Os alunos, mesmo que sob orientação para solução de dúvidas, partiram da descrição do sistema em linguagem natural, construíram o LAL e chegaram a Modelos i* com mais facilidade e com melhores resultados se compararmos ao primeiro exercício de aplicação do Framework i*.

A experimentação concluiu que o Método ERi*c contribui com 23% mais competências úteis para a Engenharia de Requisitos de Sistemas Multi-Agentes em adição às competências já fornecidas pelo Framework i*. Acreditamos que agora, com

as correções e mais exemplos, outra experimentação apresentaria resultados ainda melhores sobre a adequação das competências.

6.2 Declaração de contribuição e comparação com outras propostas

6.2.1 As contribuições do Método ERI*c

O objetivo da seção é apresentar as contribuições do método Engenharia de Requisitos Intencional - ERi*c à Engenharia de Requisitos de Sistemas Multi-Agentes. Elas são percebidas nos modelos, nas heurísticas e no processo. A Figura 3.1 mostra um esquema simplificado do método dividido em seis etapas.

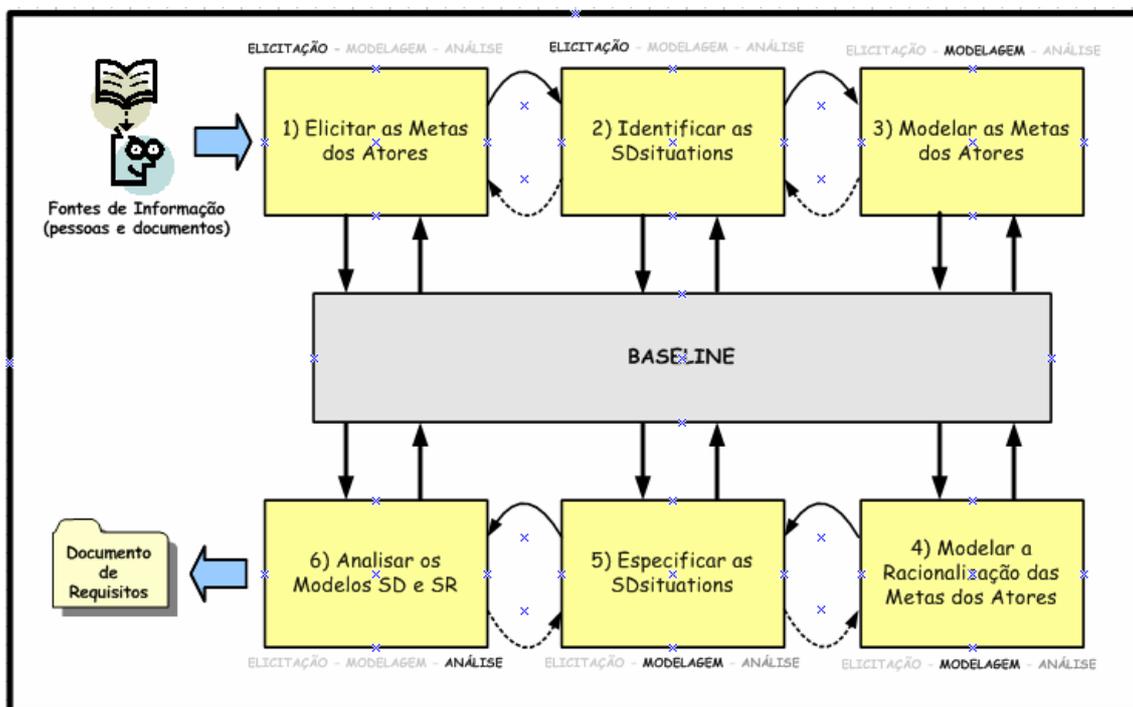


Figura 3.1 – Visão geral do Método ERI*c

Na primeira etapa, o método traz a contribuição de uma técnica para elicitar as metas “goals” dos atores de uma organização. A técnica denominada “Agent Goals from Lexicon” - AGFL [Oliveira 07], ou Metas dos Agentes Vindas do Léxico, faz a elicitação das metas a partir da linguagem do domínio da aplicação, do LAL [Leite 93].

Na segunda etapa, a contribuição é a idéia de “Strategic Dependency Situations” - SDSituations ou Situações de Dependência Estratégica, definida por Oliveira e Cysneiros [Oliveira 06b]. As SDSituations trabalham para que, no processo de

modelagem do Framework i^* , a variável tempo e as ligações tipo pontes (ou “bridges”) sejam consideradas através de uma abordagem situacional.

Na etapa três, a contribuição do método é a representação de metas concretas e metas flexíveis no diagrama “Intentionality Panel” - IP Diagram [Oliveira 07], ou Painel de Intencionalidade. O diagrama traz benefícios relativos à representação das metas e suas relações em um diagrama exclusivo.

Na etapa quatro, há duas contribuições: o artifício do detalhamento de metas concretas e de metas flexíveis em “mini” modelos SR, chamados de SRconstructs [Oliveira 07], para a simplificação de Modelos SR, e a orientação ou guia, para a derivação de Modelos SR a partir do Modelo SD [Oliveira 08].

A quinta e penúltima etapa contribui com heurísticas para a especificação das SDSituations através da Técnica de Cenários [Leite 00]. As heurísticas sugeridas em [Oliveira 06a] desejam preparar a especificação para a arquitetura de Sistemas Multi-Agentes e, por isso, as heurísticas priorizam as “propriedades de agência” [Wooldridge 02], [Silva 04], [Garcia 04].

A sexta e última etapa contribui com a técnica de diagnósticos, denominada “ i^* Diagnoses” [Oliveira 08] ou Diagnósticos de Modelos i^* , para verificar Modelos SD e Modelos SR.

6.2.2 Os Principais Métodos GORE

A Engenharia de Requisitos Orientada a Metas (GORE - Goal-Oriented Requirements Engineering) recomenda que os requisitos de sistemas de software devam, inicialmente, enfatizar o “por que” e o “como”, ao invés de se fixarem em questões de “o que” precisa ser implementado. Métodos tradicionais de análise e desenho de software focalizam a funcionalidade do software a ser desenvolvido e as interações com os usuários. Em vez de perguntar o que o sistema necessita fazer, métodos orientados a metas perguntam por que certa funcionalidade é necessária e como ela pode ser implementada. Em razão disso, métodos orientados a metas trabalham com o “rationale” para chegar à funcionalidade do sistema enquanto também investigam diferentes alternativas e critérios para a seleção dentre elas.

Esta seção compara o Método ERI^{*c} aos métodos GBRAM (Goal Based Requirements Analysis Method) e KAOS (Knowledge Acquisition in autOdated

Specification), no ponto que consideramos mais relevante – a aplicação e a representação da idéia de metas.

6.2.2.1 O que se entende por meta nos principais métodos GORE

a) Metas GBRAM [Anton 97]

Metas são alvos a serem alcançados. Elas fornecem uma “moldura” para o sistema desejado. São objetivos de alto nível do negócio, organização ou sistema. Elas expressam o “rationale” do sistema proposto e guiam decisões em vários níveis da organização. Dois tipos primários de metas são trabalhados pelo método GBRAM: as metas a serem alcançadas e as metas a serem mantidas.

b) Metas KAOS [Dardenne et al. 93]

Uma meta é um objetivo não-operacional a ser alcançado pelo sistema. Não-operacional significa, segundo Dardenne, que o objetivo não é formulado em termos de objetos e ações disponíveis para algum agente do sistema. O método KAOS diferencia metas particulares e metas do sistema. As metas particulares são as metas dos agentes individuais e as metas do sistema são aquelas do sistema como um todo.

c) Metas ERi*c [Oliveira et al. 07]

Uma meta é uma condição ou estado desejados no mundo que um ator quer alcançar [Yu 95]. O Método ERi*c diferencia metas próprias de metas estratégicas. As metas próprias são as metas dos atores individuais e as metas estratégicas são aquelas que os atores alcançam devido às dependências estratégicas de colaboração. As metas próprias e as metas estratégicas são especializadas em metas concretas e metas flexíveis. As metas flexíveis, que representam os requisitos não-funcionais (RNFs), não possuem critério de satisfação bem definido; diferentemente das metas concretas, que possuem o critério bem definido.

d) Refinamento e abstração de metas [van Lamsweerde 2000]

Refinamento de metas é o processo no qual as metas são “partidas” em submetas. A abstração de metas trabalha no sentido oposto fornecendo metas mais abstratas. As metas podem ser refinadas em submetas perguntando-se como essas metas podem ser alcançadas, enquanto metas mais abstratas são encontradas perguntando-se por que certa meta é alcançada.

e) A origem das metas [Anton 1997, van Lamsweerde et al. 1995]

As principais fontes para identificar metas são cenários, casos de uso, transcrições de entrevistas, documentos de missão, documentos de política e metas corporativas.

e) A origem das metas [Oliveira et al. 07]

As metas, concretas e flexíveis, são elicitadas do LAL - Léxico Ampliado da Aplicação, que é obtido através de documentos e pessoas que pertencem ao UdI – Universo de Informações.

6.2.2.2 GBRAM (Goal Based Requirements Analysis Method)

O Método GBRAM define uma abordagem para refinar as metas atribuindo-as a agentes. A elicitação das metas parte de documentos de missão da organização, documentos de política e transcrições de entrevistas.

6.2.2.3 KAOS (Knowledge Acquisition in autOmedated Specification)

O Método KAOS é composto por um framework formal, assistido por lógica temporal e técnicas de refinamento de IA, em que todos os termos, tais como meta e estado, são definidos rigorosamente. O principal objetivo do KAOS é a prova formal que os requisitos casam com as metas definidas para o sistema almejado. O KAOS também define uma taxonomia de duas dimensões para as metas [Dardenne et al. 1993]. Outra característica é que o KAOS trabalha com a noção de restrição e obstáculo no modelo de metas adotadas por ele.

6.2.2.4 ERi*c (Engenharia de Requisitos Intencional)

O Método ERi*c define como critério uma estratégia situacional para fazer a abstração de metas, para adicionar a noção de tempo nos modelos i* e para controlar a complexidade de modelos i*. O ERi*c define um diagrama específico para trabalhar com as metas e transfere para a etapa de especificação o detalhamento da representação de restrições e obstáculos. Esse método prescreve uma etapa para a verificação dos modelos que executa um processo de perguntas para o diagnóstico dos modelos.

O diagrama da Figura 6.1 foi inspirado em diagramas do Método KAOS [Dardenne 91] [Heaven 04], apresentando os quatro tipos do LAL. Está representado que atores (sujeitos) desejam metas (estados) e são responsáveis por ações (verbos). As metas são inter-relacionadas a metas e requerem ações. As ações, por sua vez,

algumas atividades de estágios diferentes). Os estágios são: a modelagem das atividades humanas, a modelagem das metas com a utilização de modelos i^* , a modelagem de casos de uso e a gerência de requisitos.

Outro trabalho com alguma similaridade é de Cruz Neto et al. [Cruz Neto 08] intitulado “Balanceando entre a Riqueza do Campo e a Praticidade do Design de Software” que trata do mapeamento entre diagramas da Teoria da Atividade para modelos organizacionais baseados em i^* . Nesse trabalho os autores aplicam as idéias de análise etnográficas baseadas na Teoria da Atividade para chegar aos modelos i^* .

6.3 Limitações, Trabalhos Futuros e Desdobramentos

6.3.1 Limitações do Método ERI*c

Como fator limitador, identificamos a dependência ou necessidade do método em relação a ferramentas de software para facilitar a integração entre as fases do método. As ferramentas mencionadas na próxima seção são muito importantes para que o método esteja habilitado a enfrentar projetos de desenvolvimento de sistemas complexos de grande porte (muitos objetivos e clientes) e com funcionalidade crítica para o negócio, além de requisitos de desempenho do tempo de desenvolvimento.

6.3.2 Trabalhos Futuros

Em trabalhos futuros, esperamos medir se existe algum “overhead” a respeito da aplicação do método. Temos consciência de que a avaliação dessa natureza requer cuidados especiais para que os resultados sejam perfeitamente idôneos e confiáveis, porque medir “overhead” significa comparar quantitativamente resultados entre duas ou mais abordagens.

Em nossos planos está uma experimentação para avaliar qual é a recompensa da aplicação da estratégia de Diagnósticos i^* em projetos reais de desenvolvimento. Pensamos em empregar mais de uma equipe independente para avaliar o resultado dos diagnósticos, do mesmo modo que foi aplicado na técnica de avaliação “N-Fold inspection” [Martín 90].

O desenvolvimento de ferramentas de modelagem integradas é necessário para viabilizar a eficiência do método proposto. Em geral, os elementos usados em uma etapa e atividade (como atores, metas, tarefas e recursos) aparecem em mais de um

diagrama (no LAL, em SDSituations, no diagrama IP, nos modelos i^* e no texto de cenários, como também em diagnósticos).

O desenvolvimento de ferramentas de apoio ao método também é importante para a documentação do rastro do “rationale” das decisões tomadas pelo engenheiro de requisitos e clientes, como também para rastrear os próprios requisitos. A rastreabilidade de requisitos é condição relevante para uma eficaz Gerência de Requisitos [Breitman 98] [Toranzo 02].

Os trabalhos futuros deverão se concentrar nas ferramentas que o método necessita:

- AGFL – Agent Goals from Lexicon
Objetivo: fazer a “ponte” entre o LAL e a elicitación das metas.
- Diagramas de SDSituations e diagramas IP
Objetivo: fazer a diagramação de SDSituations com as metas estratégicas para a avaliação da propagação de problemas.
- Diagramador de SRconstructs e de Diagnósticos i^*
Objetivo: fazer a diagramação de SRconstructs e preparar os templates de diagnósticos a partir de SDSituations e SRconstructs.

Além do desenvolvimento de ferramentas específicas para se trabalhar com o Método ERi^*c , um trabalho que trará benefícios com pouco investimento é integrar os diagramas do Método ERi^*c e a OME, ou Open OME, através de um mecanismo de importação de modelos usando o padrão XML.

6.3.3 Desdobramentos

Um importante potencial de desdobramento é justamente a integração do Método ERi^*c com outro método de desenho e implementação de sistemas multi-agentes. Um exemplo que cabe à análise de viabilidade, porque esse processo já foi feito manualmente em [Oliveira 06], é o framework de orientação a aspectos chamado ASPECTT [Garcia 04a].