

CRISTIANE APARECIDA LANA

**TÉCNICA BASEADA EM SINÔNIMOS PARA AUXILIAR NA
ELABORAÇÃO DE MODELOS iSTAR**

**Dissertação apresentada à Universidade
Federal de Viçosa, como parte das
exigências do Programa de Pós-
Graduação em Ciência da Computação,
para obtenção do título de Magister
Scientiae.**

**VIÇOSA
MINAS GERAIS - BRASIL
2014**

**Ficha catalográfica preparada pela Biblioteca Central da
Universidade Federal de Viçosa - Câmpus Viçosa**

T

L243t
2014 Lana, Cristiane Aparecida, 1979-
Técnica baseada em sinônimos para auxiliar na
elaboração de modelos istar / Cristiane Aparecida Lana. -
Viçosa, MG, 2014.
xix, 128f. : il. (algumas color.) ; 29 cm.

Inclui apêndices.
Orientador : José Luis Braga.
Dissertação (mestrado) - Universidade Federal de
Viçosa.
Referências bibliográficas: f.90-98.

1. Engenharia de software. 2. Software -
Desenvolvimento. 3. Computação. 4. Istar. I. Universidade
Federal de Viçosa. Departamento de Informática. Programa
de Pós-graduação em Ciência da Computação. II. Título.

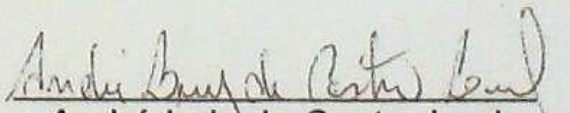
CDD 22. ed. 005.1

CRISTIANE APARECIDA LANA

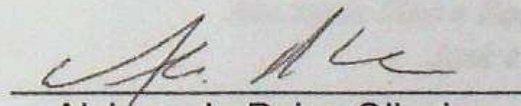
**TÉCNICA BASEADA EM SINÔNIMOS PARA AUXILIAR NA
ELABORAÇÃO DE MODELOS ISTAR**

Dissertação apresentada à Universidade Federal de Viçosa, como parte das exigências do Programa de Pós-Graduação em Ciência da Computação, para obtenção do título de *Magister Scientiae*.

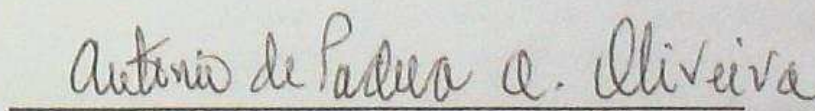
APROVADA: 17 de setembro de 2014.



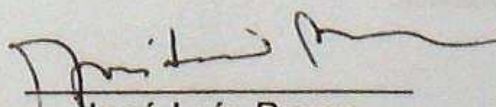
André Luiz de Castro Leal



Alcione de Paiva Oliveira



Antônio de Pádua Albuquerque Oliveira
(Coorientador)



José Luís Braga
Orientador

Dedico essa dissertação a minha mãe
Maria do Carmo

Aos meus Tios e Padrinhos
José e Miraci

Ao meu orientador
José Luis Braga

Ao meu co-orientador
Antônio de Pádua Albuquerque Oliveira

E aos demais parentes e amigos.

“[...] nossas lembranças permanecem coletivas, e elas nos são lembradas pelos outros, mesmo que se trate de acontecimentos nos quais só nós estivemos envolvidos, e com objetivos que só nós vimos. É porque, em realidade, nunca estamos sós [...]” (Wasserman, 1996).

“Aqueles que esperam no Senhor renovam as suas forças. Voam alto como águias. Correm e não ficam *exaustos, andam e não se cansam*”. (Isaias 40:31)

AGRADECIMENTOS

Nos reinventamos a cada tentativa de buscar respostas às nossas aflições, sejam elas diárias ou de ‘pesquisador’. Para aqueles que compartilham conosco desse momento, parece uma tarefa interminável e enigmática que só se torna realizável graças as muitas pessoas que participam, direta ou indiretamente, mesmo sem saber realmente o que e para que nos envolvemos. E sem dúvida, é a essas pessoas que eu quero deixar meus singelos e sinceros agradecimentos.

Início agradecendo a **Deus**, pelo infinito amor e cuidado, por me amparar em todos os momentos de minha vida, por ser a rocha que me sustenta e por me permitir vencer mais essa etapa em minha vida.

A minha amada e querida mãe, **Maria do Carmo**, que me ensinou a ousar, questionar e, acima de tudo ser persistente e curiosa. Muito persistente e curiosa.

Obrigada, mãe, por fazer tudo isso possível, pelos dias corridos, pelas noites não dormidas, pelos finais de semanas, madrugadas e feriados trabalhados, por se esforçar para me dar o que há de melhor, por ser meu chão, minha força e a determinação de minha vida.

Ao meu Pai **Manoel Vieira**, que de aluna me adotou como filha. Obrigada por cuidar de mim com o mesmo amor e carinho que cuida de seus filhos de sangue. Por ter me amparado em momentos difíceis e por ter estado presente em diversos momentos de vitórias. O senhor é um grande exemplo de vida para mim.

Aos meus tios e padrinhos **Miraci** e **José**, pais por adoção. Amo vocês, obrigada por fazer com que este período fosse mais tranquilo, e por sempre estarem do meu lado.

A todos os meus familiares, obrigada por permanecerem ao meu lado.

Aos meus amigos que sempre estiveram ao meu lado e em especial a **Ana Christina** que foi muito mais que minha psicóloga, se tornou minha amiga/mãe/conselheira. Muitas vezes Ana, você acreditou mais em mim que eu mesma. Obrigada por não me deixar desistir.

A **Vand** e ao **Anderson**, obrigada pelas horas de leitura, pelas noites não dormidas e pelas dificuldades vencidas, se não fosse o desafio que vocês me deram na graduação, com certeza todas as demais atividades acadêmicas se tornariam mais difíceis e eu não conheceria a satisfação que tenho em trabalhar com pesquisa. Obrigada meus eternos” livros de cabeceira”.

Joás e família, obrigada por me acolherem com tanto carinho em sua casa e a você Joás, pelos ensinamentos. Eles foram fundamentais para eu poder começar a engatinhar e chegar até aqui. Minha eterna gratidão.

Ao meu pequeno, **Giorgio**, anjo que caiu do céu. Que desde o primeiro semestre me ajudou e trabalhou comigo até o fim. Pequeno, você é simplesmente demais. Gosto muito.

Ao **Marcão**, meu desenhista particular, obrigada por todo cuidado, atenção e por todo auxílio.

A **Luciana Rocha, Aparecida, Renne e Mariana e ao David Ladeira**, obrigada por estarem sempre presentes nos momentos mais importantes da minha vida, pelas orações e que nossa amizade seja eterna.

Ao **José Nilton e a Juliana Solar**, vocês me ensinaram o que é superação e que para chegar onde queremos só precisamos confiar em Deus e ter força de vontade. Obrigada meus anjos, amigos, irmãos, vocês são muito especiais pra mim.

A **Teresa Carlos**, anjo que caiu do céu e com palavras singelas e doces alegrou os meus dias e me deu força para continuar. Muito obrigada Dona Teresa.

Aos **amigos do Scheilla**, obrigada por me receberem e me ampararem na maior batalha da minha vida, muitos foram e muitos serão os ensinamentos adquiridos junto de vocês, eles com certeza me fazem ver a vida com uma leveza que antes não me era possível. Sem esse auxílio, tudo isso não teria sido possível. A minha eterna gratidão.

A todos os **Alunos do Mestrado do DPI** e, em especial, aos meus colegas de turma, meu agradecimento. Passamos muitos momentos juntos, viveu-se amores, encontros e desencontros, brigamos muitas vezes, mas aprendemos que unidos somos mais fortes, e por isto caminhamos assim. De vocês, meus amigos, levarei lindas lembranças, e mesmo que a vida nos afaste, estarão sempre em meu coração.

Ao meu **Grupo de Pesquisa** e em especial ao **Lucas**, foi muito bom poder compartilhar com vocês todos esses momentos. As reuniões, as conversas, as discussões, as pizzas, as trocas de informações e os conselhos. Se tudo foi mais simples é porque vocês também estavam por perto.

A **Comissão Coordenadora do Programa de Pós-Graduação em Ciência da Computação**, obrigada pela oportunidade ímpar de aprendizado ao lado de vocês.

Aos **Funcionários e estagiários do Departamento de Informática**, muito obrigado, por me receberem com tanto carinho. Em especial ao **Altino Alves de Souza Filho**, secretário da Pós-graduação, obrigada pelas conversas, conselhos e por sempre me auxiliar nas burocracias e atividades do mestrado.

Aos **Mestres do Departamento de Informática**, a minha eterna gratidão. Vocês que de forma tão concreta e ativa, contribuíram para minha formação. A todos, o meu muito obrigada, por todo o incentivo. Tenho certeza que levarei todos os ensinamentos não só acadêmicos, mas de vida, por todos os lugares que eu for.

Ao **André Gustavo** e a **Luciana Brugiolo**, dois anjos, lindos anjos que fizeram com que esta etapa de minha vida fosse mais leve, mais prazerosa, mais doce, bem mais doce...rsrsrs e simplesmente divertida. Obrigada André e Lú pela amizade, pela atenção, pelo cuidado e principalmente pelos ensinamentos. A Lú, obrigada também pelos colos e por todos os “segredos” e risadas. A vocês, minha eterna gratidão.

Ao professor **Alcione**, pela disposição em me auxiliar nas inúmeras dúvidas durante a execução deste trabalho. Meu muito obrigado.

Agradeço ao meu orientador, amigo e professor **José Luis Braga**. Já estamos trabalhando juntos há algum tempo e a liberdade facilita o diálogo e a interação. Obrigada Pai zé, por acreditar e confiar em mim, pelo apoio, atenção, cuidado e compreensão e por todos os conselhos, ensinamentos e pelas sábias palavras ao longo desses anos de trabalho. Obrigada por ser muito mais que orientador e pai, por ser meu amigo e por sempre me mostrar a direção correta que eu deveria seguir, mesmo que indiretamente. Agradeço, também, ao meu co-orientador **Antônio de Pádua Albuquerque Oliveira**, pelos conselhos, conversas e contribuições dadas a este trabalho.

Agradeço ao financiamento fornecido pela **CAPES**, que possibilitou que eu me dedicasse exclusivamente à pesquisa e à escrita dessa dissertação.

Agradeço a **Universidade Federal de Viçosa** e ao **Departamento de Informática** por abrirem as portas para que eu pudesse realizar mais um sonho. Proporcionaram-me mais que a busca de conhecimento técnico e científico, mas uma **LIÇÃO DE VIDA**.

Ninguém vence sozinho...À TODOS que, de algum modo, contribuíram direta e indiretamente para conclusão desse trabalho, o meu **MUITO OBRIGADA!!!**

BIOGRAFIA

CRISTIANE APARECIDA LANA, filha de Maria do Carmo Silva e Antônio Lana Filho, brasileira nascida em 11 de março de 1979 na cidade de Belo Horizonte, no estado de Minas Gerais.

No ano de 2005, ingressou na graduação em Sistemas de Informação da Faculdade de Viçosa – FDV, na cidade de Viçosa, Minas Gerais, concluindo no ano de 2008.

Em 2010 deu início a pós-graduação Lato Sensu em Gestão Estratégica de Pessoas na Faculdade de Ciências Biológicas e da Saúde – UNIVIÇOSA, na cidade de Viçosa, concluindo em 2012.

Ainda no ano de 2012, foi aprovada na seleção do programa de pós-graduação do Departamento de Informática – DPI, onde em vinte seis de novembro de 2012, iniciou o mestrado em Ciência da Computação na Universidade Federal de Viçosa – UFV, defendendo sua dissertação em Setembro de 2014.

Atualmente é professora da Faculdade de Ciências Gerenciais de Manhuaçu e atua como orientadora de trabalhos de Graduação na Faculdade de Viçosa e pós-graduação Lato Sensu pela FACISA - UNIVIÇOSA. Tem experiência na área de Sistemas de Informação com ênfase em Engenharia de Software, Engenharia de Requisitos e Representação do Conhecimento, Sistemas de Apoio à Decisão e Gestão de Pessoas.

SUMÁRIO

LISTA DE FIGURAS	xi
LISTA DE TABELAS	xiii
LISTA DE GRÁFICOS	xiv
LISTA DE ABREVIATURA E SIGLAS	xv
RESUMO	xvi
ABSTRACT	xviii
1 INTRODUÇÃO	1
1.1 O problema e sua importância.....	3
1.2 Objetivos	5
1.2.1 Objetivo Geral.....	5
1.2.1.1 Objetivos Específicos.....	5
1.3 Organização da dissertação	6
2 REFERENCIAL TEÓRICO.....	8
2.1 A Elicitação de Requisitos	8
2.2 As Metas Flexíveis.....	11
2.3 O Framework I*	13
2.3.1 Modelo SD	14
2.3.2 Modelo SR	16
2.4 O Léxico Ampliado de Linguagem – LAL.....	18
2.5 O Wordnet – A lexical database for english	20
2.6 Sistema de Produção Integrado da Linguagem C - CLIPS	21
2.7 Trabalhos Correlatos	22
3 PROCEDIMENTOS METODOLÓGICOS	24
3.1 Aspectos Metodológicos: Caracterização da pesquisa.....	24
3.2 Processo de escolha das ferramentas.....	25
3.3 Escolha do dicionário de sinônimos.....	27
3.4 Fluxo de trabalho utilizado no desenvolvimento da pesquisa.....	29
4 IMPLEMENTAÇÃO DA TEKBS PARA ELICITAÇÃO DE METAS FLEXÍVEIS	32
4.1 Trabalhos considerados base da TEKBS	32
4.2 Método Eri*c – Engenharia de Requisitos Intencional.....	33
4.2.1 Elicitar Metas de Atores.....	36
4.2.1.1 Definir AGFL – Metas dos Agentes vindas do Léxico.....	37
4.3 Processo de Construção da TEKBS	39
4.4 Elicitar Metas Flexíveis com apoio de uma ferramenta de Inteligência Artificial	42
4.4.1 Elicitar Metas Flexíveis.....	43

4.4.2	Selecionar e Armazenar Metas Flexíveis.....	45
4.4.3	Relacionar Metas Flexíveis com Sinônimos.....	48
4.4.4	Refinar Metas Flexíveis.....	51
5	EXEMPLO DE APLICAÇÃO DA TÉCNICA TEKBS.....	54
5.1	Processo de validação da TEBKS.....	54
5.2	Expert Committee System.....	56
5.3	Execução da TEKBS utilizando o EC.....	56
5.3.1	Selecionar e Armazenar Metas Flexíveis.....	57
5.3.2	Relacionar Metas Flexíveis com Sinônimos.....	60
5.3.3	Refinamento de Metas Flexíveis.....	64
5.3.4	Armazenar Metas Flexíveis.....	68
5.4	Análise de acoplamento das Metas Flexíveis Sinônimas.....	69
5.4.1	Processo de seleção de sinônimos.....	70
5.4.2	Análise de contexto das Metas Flexíveis sinônimas.....	71
5.4.3	Comparação da TEKBS com o processo Manual realizado por (Oliveira, 2008).....	74
5.4.4	Comparação do processo realizado pela TEKBS utilizando os Dicionários de sinônimos Macmillan e Thesaurus.....	76
5.5	Matriz de Influência das Metas Flexíveis.....	80
5.6	Generalização do processo de TEKBS.....	82
6	CONSIDERAÇÕES FINAIS.....	86
6.1	Principais contribuições.....	88
6.2	Trabalhos futuros.....	88
7	REFERÊNCIAS BIBLIOGRÁFICAS.....	90
8	APÊNDICE A. BASE DE REGRAS.....	99
9	APÊNDICE B: BASE DE FATOS.....	113
10	APÊNDICE C: SOFTGOALFILE.....	119
11	APÊNDICE D: RESULTADO DA BUSCA DE SINÔNIMOS.....	120
12	APÊNDICE E: BASE WORDNET COM SINÔNIMOS VÁLIDOS.....	124
13	APÊNDICE F: BASE MACMILLAN COM SINÔNIMOS VÁLIDOS ...	125
14	APÊNDICE G: BASE THESAURUS COM SINÔNIMOS VÁLIDOS	127

LISTA DE FIGURAS

Figura 2.1: Processo de Requisitos.	9
Figura 2.2: Tipos de dependência entre atores no Modelo SD	15
Figura 2.3: Relacionamentos intencionais internos aos atores.....	16
Figura 2.4: Exemplo do Modelo SR (Parte do Modelo)	18
Figura 2.5: Os elementos do LAL representados em um diagrama de classes.	19
Figura 3.1: Fluxo das etapas de desenvolvimento da pesquisa.....	31
Figura 4.1: Visão geral do Método Eri*c.....	34
Figura 4.2: Detalhamento da 1ª etapa do Método ERI*c.....	36
Figura 4.3: Detalhamento da 1ª etapa do Método ERI*c automatizada por Cunha (2014).....	37
Figura 4.4: Ações flexíveis de símbolos tipo sujeito e tipo objeto.	38
Figura 4.5: Visão Geral da TEKBS inserida no Método ERI*c.	41
Figura 4.6: Exemplo de um símbolo do LAL tipo estado.....	42
Figura 4.7: Transcrição do símbolo conflito da Fig. 2 para o Template 3.C.	43
Figura 4.8: Estrutura de parametrização da MF.....	43
Figura 4.9: Estrutura do módulo principal	44
Figura 4.10: Estrutura responsável por acionar o “Module Extracting”	45
Figura 4.11: Estrutura responsável por finalizar a TEKBS.....	45
Figura 4.12: Regra responsável por retornar ao menu principal.....	45
Figura 4.13: Regra responsável por guardar as Metas Flexíveis na basesoftgoal.....	46
Figura 4.14: Exibir softgoals raizes	47
Figura 4.15: Salva softgoals encontradas na base de fatos em BasesoftgoalFile.clp.	47
Figura 4.16: Salva sinônimos encontradas na base de fatos em "NewSynonymousFile.clp"	48
Figura 4.17: Base de Sinônimos.	49
Figura 4.18: Tela do programa JAVAS que acessa o Wordnet e retorna sinônimos das Metas Flexíveis	50
Figura 4.19: Consulta individual de sinônimo de Meta Flexível	50
Figura 4.20: Regra responsável por excluir duplicidade.....	52
Figura 4.21: Função responsável por deletar uma única Meta Flexível.	52
Figura 4.22: Função responsável por inserir sinônimos de Metas Flexíveis.	53
Figura 5.1: Utilização dos comandos (reset) e (run).....	58
Figura 5.2: Execução do Module Extracting e Print Softgoals.....	59
Figura 5.3: Fluxo da atividade Seleccionar e Armazenar Metas Flexíveis.....	60
Figura 5.4: Levantamento de sinônimos utilizando o JAVAS.....	61
Figura 5.5: Inserção da BS na TEKBS.	62
Figura 5.6: Metas Flexíveis sinônimas impressas na tela do CLIPS.	63
Figura 5.7: Fluxo da atividade Relacionar Metas Flexíveis com Sinônimos.....	63
Figura 5.8: Exclusão de MF Sinônimas fora do contexto EC.....	65
Figura 5.9: Memória de trabalho antes e após a utilização do "Module Single Delinting".....	66
Figura 5.10: Inserção de MF sinônimas.....	67
Figura 5.11: Fluxo da atividade Refinar Metas Flexíveis.....	68
Figura 5.12: Fluxo da atividade Armazenar Metas Flexíveis.	69
Figura 5.13: Processo de Análise e Seleção de Sinônimos.....	80

Figura 5.14: Matriz de Influências de Metas Flexíveis Raiz no contexto do EC.....	81
Figura 5.15: Processo da TEKBS generalizado	84
Figura 5.16: Processo generalizado da TEKBS detalhado.....	85

LISTA DE TABELAS

Tabela 3.1 Comparação de Ferramentas	26
Tabela 3.2: Comparação dos dicionários de sinônimos.	28
Tabela 5.1: Tabela comparativa dos dicionários de sinônimos de Língua Inglesa. ...	76
Tabela 5.2: Comparação de MF elicitadas pela TEKBS versus as elicitadas por (Oliveira, 2008) e por (Cunha, 2014).....	78

LISTA DE GRÁFICOS

Gráfico 5.1: Metas Flexíveis selecionadas por grupos de análises.	72
Gráfico 5.2: MFs descartadas e MFs válidas.	72
Gráfico 5.3: Comparação entre as MF Válidas e Base de Dados de (Cunha, 2014). ..	73
Gráfico 5.4: Comparação de MF elicitadas por (Oliveira, 2008) x TEKBS.....	75
Gráfico 5.5: Metas Flexíveis selecionadas por grupos de análise e por dicionários..	77

LISTA DE ABREVIATURA E SIGLAS

AGFL	Metas dos Agentes Vindas do Léxico
API	Application Programming Interface
BS	Base de Sinônimos
C&L	Cenários e Léxicos
CLIPS	Sistema de Produção Integrado da Linguagem C
EC	Expert Committee
ERi*c	Método de Engenharia de Requisitos Intencionais
GORE	Engenharia de Requisitos Orientados a Meta
HTML	Hyper Test Markup Language
i*	IStar, iEstrela ou Intencionalidade Distribuída
IP	Painel de Intencionalidade
LAL	Léxico Ampliado de Linguagem
MC	Meta Concreta
MF	Meta Flexível
MI	Matriz de Influência
MR	Matriz de Rastreabilidade
PUC-RIO	Pontifícia Universidade Católica do Rio de Janeiro
RNFs	Requisitos Não-Funcionais
SD	Dependência Estratégica
SDsituation	Situação de Dependência Estratégica
SI	Sistemas de Informação
SR	Razão Estratégica
SRS	Especificação de Requisitos de Software
TEKBS	Técnica de Elicitação de Metas Flexíveis Baseadas em Conhecimento
UdI	Universo de Informação
UERJ	Universidade do Estado do Rio de Janeiro

RESUMO

LANA, Cristiane Aparecida, M.Sc., Universidade Federal de Viçosa, setembro de 2014. **Técnica baseada em sinônimos para auxiliar na elaboração de Modelos iStar**. Orientador: José Luis Braga. Coorientador: Antônio de Pádua Albuquerque Oliveira.

Elicitar Metas Flexíveis ou “softgoals”, que são critérios de qualidade segundo a linguagem para modelagem de requisitos i-star, em diferentes níveis de abstração, não é uma tarefa trivial, porém a elicitação é extremamente necessária para a identificação dos requisitos não funcionais do sistema. Como os processos de elicitação não são totalmente delineados ou são quase que integralmente Ad-Hoc, a utilização de uma aplicação baseada em conhecimento permitiu soluções que fizeram uso da inteligência artificial facilitando o processo de elicitação. A técnica TEKBS visou aprimorar uma subatividade do Método ERI*c que auxiliou os engenheiros de requisitos no processo de elicitação de Metas Flexíveis. Para implementação da TEKBS, foram identificadas no Léxico Ampliado de Linguagem (LAL) do domínio de uma aplicação “ações flexíveis” que foram combinadas com os sinônimos das mesmas extraídos do tesouro Wordnet da língua inglesa, gerando uma lista de metas flexíveis candidatas. Estas compuseram a base de fatos relacionados com o contexto do problema, que foi utilizada por uma máquina de inferência da linguagem CLIPS, para a elicitação não somente das Metas Flexíveis explícitas, mas também daquelas que pudessem estar implícitas no contexto do problema. Além disso, foi elaborada uma matriz de influência das Metas Flexíveis que utilizou em sua estrutura as “ações flexíveis” e a partir delas a verificação de contribuições. Os resultados alcançados por meio da aplicação da técnica TEKBS sugerem que a utilização da inteligência artificial juntamente com a automatização parcial do Método ERI*c adotada neste trabalho auxiliam no processo de elicitação de Metas Flexíveis, facilitando o processo de aprendizagem, expansão, separação, controle e flexibilização do processo de modelagem. Além de proporcionar ao engenheiro de requisitos mais abrangência, possibilidade de requisitos mais bem identificados e maior facilidade na verificação das influências mútuas existentes entre as Metas Flexíveis. Isso se torna

possível porque a elicitação utilizando a TEKBS fornece ao engenheiro de requisitos um número maior de Metas Flexíveis que auxiliam no atendimento de mais necessidades dos envolvidos e ainda no refinamento das necessidades elicítadas.

ABSTRACT

LANA, Cristiane Aparecida, M.Sc., Universidade Federal de Viçosa, September, 2014. **Thesaurus-based technique to assist in the elaboration of Models iStar.** Adviser: José Luis Braga. Co-Adviser: Antônio de Pádua Albuquerque Oliveira.

Eliciting Flexible Goals or "soft goals", which are quality criteria according to language for modelling i-star requirements at different levels of abstraction, is not a trivial task, but the elicitation is extremely necessary for the identification of requirements for a non-functional system. As the processes of elicitation are not fully delineated or are almost entirely ad-hoc, using an application based on knowledge enabled solutions that made use of artificial intelligence facilitating the elicitation process. The TEKBS technique aimed at improving a sub activity, the Eri *c Method, that helped engineers with the requirements in the process of elicitation of Flexible Goals. To implement the TEKBS there were identified in the Expanded Definition Language (LAL) domain, application "soft actions" that were combined with the synonyms of the same extracted from the Wordnet thesaurus of the English language, generating a list of flexible candidate targets. These formed the basis of facts related to the context of the problem, which was used by the inference machine CLIPS language, not only for the elicitation of explicit flexible goals, but also those that could be implied in the context of the problem. In addition, we created a matrix of influence of goals that used flexibility in its structure "flexible actions" and from them to check contributions. The results achieved through the application of the TEKBS technique suggest that the use of artificial intelligence along with the partial automation of the Eri *c method adopted in this study to assist in the elicitation of the Flexible goals process, facilitating the process of learning, growth, separation, control and flexibility of the modelling process. Besides providing the most comprehensive engineering requirements, it also provides the possibility for better requirements identified and easier verification of existing mutual influences between Flexible Goals. This is possible because the elicitation using TEKBS provides the engineering requirements of a greater number of targets that support flexibility in

meeting the needs of those involved and still have the refinement of elicited requirements.

Capítulo 1

1 INTRODUÇÃO

A utilização da tecnologia de software em aplicações comerciais e industriais tem crescido consideravelmente e nos últimos anos, diversos são os mecanismos e as atividades ao redor do mundo que dependem de sistemas de software para seu funcionamento. Em média 40% da população mundial deixariam de realizar suas atividades caso sistemas de uso global deixassem de funcionar (Reed, 2000).

A tecnologia exerce um papel crescente na economia e as empresas têm crescido mais fortemente dependentes da entrega bem sucedida de Sistemas de Informação - SI (Jun, Qiuzhen e Qingguo, 2011). Entretanto, no processo de desenvolvimento de SI falhas em projetos são comuns. Estudos como de (Capers, 1996) (Charette, 2005) (Jun, Qiuzhen e Qingguo, 2011) e (Sommerville, 2011) ressaltam que os requisitos incorretos, incompletos ou conflitantes são considerados as grandes causas de insucesso dos projetos de software. De forma geral, os insucessos estão associados à falta de compreensão do negócio pelo analista de sistema, ao mal entendimento da finalidade do sistema, e às falhas de comunicação entre os analistas envolvidos (Xavier, Alencar, Castro e Pimentel, 2010).

Para (Cysneiros, 2001) os fatores de insucesso estão mais intimamente associados às Metas Flexíveis – MFs – que a grande maioria das vezes são tratadas de forma secundária e informal do ponto de vista de elicitação. Assim, os requisitos obtidos podem se tornar conflitantes, estarem incorretos ou incompletos, sendo difícil sua consideração durante o processo de desenvolvimento de software.

As MFs, segundo a conceituação extraída do iStar (i*) (Yu, 1995) (Serrano, 2011) são qualidades desejáveis que não possuem critérios claros para sua satisfação, ficando dependente do ponto de vista dos interessados (Serrano, 2011). Por despreverem aspectos intencionais, as MFs devem ser cuidadosamente tratadas desde o início do projeto de software (Chung, Nixon, Yu e Mylopoulos, 2000) (Cysneiros, 2001) e sua rastreabilidade deve ser estendida por todo o ciclo de vida do

produto/serviço. Para (Brooks Jr., 1987) (Davis, 1993) e (Ernst, Borgida, Jureta e Mylopoulos, 2014) as MFs são consideradas as mais caras de obter e as mais difíceis de corrigir.

Para amenizar as limitações da elicitação das MFs, (Oliveira, 2008) criou o Método de Engenharia de Requisitos Intencional - Intentional RE Method - ERi*c, que é uma contribuição à Engenharia de Requisitos Orientada a Metas (GORE) (Oliveira, Antônio Padua Albuquerque, Leite, J.C.S. e Cysneiros, Luiz Marcio, 2008) e que tem como característica contribuir para melhorar a qualidade final dos requisitos. O ERi*c se baseia no conceito de intencionalidade como ela é aplicada no Framework i*. A intencionalidade deve refletir as motivações e os interesses dos atores de uma organização, sendo representada nos modelos por meio das metas, que podem ser concretas ou flexíveis (Oliveira, Antônio Padua Albuquerque, Leite, J.C.S. e Cysneiros, Luiz Marcio, 2008). Modelos intencionais são mais representativos que os demais modelos (Oliveira, Antônio Padua Albuquerque, Leite, J.C.S. e Cysneiros, Luiz Marcio, 2008), porém existe a dificuldade de extrair informações do domínio de aplicação.

Com base na literatura (Brooks Jr., 1987) (Mylopoulos, Chung e Nixon, 1992) (Chung e Nixon, 1995) (Rawlins e Chung, 1999) (Chung et al., 2000) (Charette, 2005) (Serrano, 2011) (Baía e Braga, 2013) (Ernst et al., 2014) nota-se que a elicitação de MF ainda é pouco abordada ou abordada de forma parcial. Sendo assim, pode-se dizer que existe uma lacuna no que tange à existência de propostas que apresentem técnicas para auxiliar o engenheiro de requisitos na tarefa de elicitar MF. A utilização de uma técnica que empregue a inteligência artificial pode auxiliar o engenheiro de requisitos a utilizar soluções que facilitem o processo de elicitação e a tomada de decisão.

A pesquisa descrita nesta dissertação propõe, projeta e implementa uma Técnica de Elicitação de Metas Flexíveis Baseada em Conhecimento – TEKBS para auxiliar os engenheiros de requisitos no processo de elicitação de MFs. A TEKBS faz uma extensão do Método ERi*c e está focada no processo descrito em sua primeira etapa que retrata um processo manual de elicitação de metas (Oliveira, Antônio Padua Albuquerque, Leite, J.C.S. e Cysneiros, Luiz Marcio, 2008) e que foi semi-automatizada por (Cunha, 2014).

A técnica é elaborada para ser utilizada ao longo do processo de elicitação e do ciclo de desenvolvimento do software, porém sua utilização mais efetiva verifica-

se nos passos iniciais da elicitação, onde é de extrema necessidade entender de fato quais as reais necessidades dos envolvidos. Evitando produzir algo que não condiz com a necessidade do cliente.

A TEKBS procura auxiliar o processo de elicitação de MFs possibilitando a extração de metas não declaradas explicitamente, via a utilização de sinônimos, com objetivo de contribuir para a elicitação de um número maior de MFs e com a minimização do número de omissões, permitindo a elaboração de um documento de requisitos mais completo e mais próximo da realidade do problema.

Como forma de facilitar o entendimento das influências entre as MFs raízes dentro do escopo, utilizou-se o conceito de Matriz de Rastreabilidade – MR e criou-se uma prova de conceito para uma Matriz de Influência - MI entre Metas Flexíveis.

A MR segundo (Sayão e Leite, 2005) (Beatty e Wiegers, 2013) e (Ernst et al., 2014) é uma das abordagens mais utilizadas para a obtenção de referências cruzadas permitindo a identificação dos relacionamentos e influências de cada requisito elicitado, além de ser essencial no gerenciamento dos requisitos e para a produção de software de qualidade.

A MI de MFs poderá auxiliar os engenheiros de requisitos no processo de tomadas de decisões antevendo principalmente os impactos que possam existir causados pelas relações e modificações das MFs.

1.1 O problema e sua importância

A Elicitação é considerada fundamental para o sucesso do desenvolvimento de sistema. A área de elicitação de requisitos é relativamente madura (Sutcliffe e Sawyer, 2013). É a fase onde os engenheiros de requisitos trabalham com os stakeholders a fim de explorar o contexto do problema (Pohl, 1997).

A fase de elicitação é considerada um dos maiores desafios no processo de desenvolvimento de software. Isso ocorre porque a falta de levantamento de requisitos, ou o levantamento feito de forma errada ou incompleta causará falhas em todo o projeto (Boehm, 2001; Pa e Zin, 2011).

Cysneiros em (Cysneiros, 2001) reassalta que as falhas ou insucessos dos projetos de software estão mais intimamente relacionados com o processo de elicitação de Metas Flexíveis – MFs, que muitas das vezes são consideradas

secundárias e informalmente do ponto de vista de elicitação. Porém, por definirem atributos de qualidade juntamente a funcionalidade a sua não elicitação, elicitação incorreta ou incompleta pode causar conflitos dificultando a consideração desses requisitos durante o ciclo de vida do software.

Normalmente, no processo de elicitação os requisitos são organizados por funcionalidades, porém, no caso das MFs, que são baseadas em intenções, elas podem estar explícitas e identificadas nas descrições, ou podem estar implícitas no contexto, escondidos por detrás de sinônimos ou frases que representam intenções. Esta é a parte mais difícil da elicitação dos requisitos, pois se identificada é preciso saber qual o impacto de cada MF dentro do projeto ou de parte do projeto e se não identificadas a tempo, podem levar a conflitos importantes, capazes de afetar a qualidade do produto final e ocasionar a insatisfação dos envolvidos (Cysneiros, 2001) (Ernst et al., 2014).

O trabalho de (Rajagopal, Lee, Ahlswede, Chiang e Karolak, 2005), identifica três classes de problemas que podem auxiliar no entendimento dos desafios na elicitação de requisitos, como problemas de escopo, de entendimento e compreensão das necessidades e volatilidade dos requisitos. Para (Farinha e Silva, 2013) a primeira razão se baseia na comunicação complexa e passível de erros entre os stakeholders e os engenheiros. A segunda, na dificuldade que as partes interessadas possuem de serem claras sobre suas necessidades e por fim, o não entendimento do contexto do negócio pelos engenheiros de requisitos.

Devido às dificuldades encontradas no processo de elicitação, muitos erros são cometidos, prejudicando o modelo de requisitos (Campos, 2009). Uma boa especificação de requisitos é determinante para a qualidade do produto. Neste caso, se um engenheiro de requisitos, analista ou desenvolvedor não compartilham o mesmo entendimento sobre os requisitos (funcionais ou não-funcionais) e Metas Flexíveis, o resultado do processo não irá satisfazer as necessidades dos stakeholders (Alshazly, Elfatary e Abougabal, 2014).

Na busca de minimizar os problemas enfrentados no processo de elicitação surgem vários grupos de pesquisa no exterior e no Brasil (Tropos Group, SERG¹, REQENG², Requirements Engineering Research Group³, PUC-RIO (ER), UFPE

¹ Software Engineering Research Group - Suécia

² Requirements Engineering Research Group- Austrália

³ University of Zurich

(LER), entre outros) que investigam formas de melhorar o processo de elicitação de requisitos. Desses esforços, métodos (Oliveira, Leite e Cysneiros, 2008), frameworks (Yu, 1995) e ferramentas (Darimont, Delor, Massonet e van Lamsweerde, 1998; Oliveira et al., 2013; Chawla, Srivastava e Malhotra, 2014) foram criadas. No entanto, a grande parte dos esforços estão voltados para a elicitação de requisitos funcionais ou requisitos não-funcionais, sendo a MF pouco abordada ou abordada de forma parcial. Nesse sentido possuir um mecanismo que facilite a elicitação de MFs, assim como de verificação da influência entre a metas, pode auxiliar o engenheiro de requisitos tanto no processo de elicitação, criando modelos mais adequados as necessidades dos stakeholders como no processo de tomada de decisão.

1.2 Objetivos

1.2.1 Objetivo Geral

A pesquisa foi iniciada com objetivo de projetar e implementar uma técnica que auxiliasse no processo de verificação das influências das Metas Flexíveis e de seus sinônimos na elaboração do modelo iStar. Contudo, com os avanços da pesquisa, percebeu-se que era necessário inicialmente, criar um mecanismo que facilitasse o processo de elicitação das MFs utilizando sinônimos e a partir da elicitação buscar formas de analisar as influências. Tal alteração teve como base estudos como de (Brooks Jr., 1987) (Chung et al., 2000) (Cysneiros, 2001) (Oliveira, Antônio Padua Albuquerque, Leite, J.C.S. e Cysneiros, Luiz Marcio, 2008) (Serrano, 2011) (Ernst et al., 2014) que relatam as dificuldades existentes para elicitar MFs.

Partindo desse pressuposto de dificuldades, o trabalho passa a considerar como objetivo geral: projetar e implementar uma técnica que auxilie a elicitação de Metas Flexíveis e a verificação das influências mútuas entre as Metas Flexíveis na elaboração do modelo iStar.

1.2.1.1 Objetivos Específicos

Iniciamos este trabalho com quatro objetivos específicos. Porém, com o avanço da pesquisa e o ajustamento do objetivo geral notou-se a necessidade de explicar e/ou justificar três deles, quais sejam o 1, 2 e 3.

O objetivo específico 1: Identificar o acoplamento entre as Metas Flexíveis e seus sinônimos, que possam impactar as decisões dos engenheiros de requisitos e na elaboração do modelo iStar. Foi criado para obter uma forma de assegurar se um sinônimo encontrado pela técnica é de fato relevante ou é de fato sinônimo no contexto da aplicação que está sendo desenvolvida, evitando assim, a associação pela técnica de um sinônimo a MF que seja desalinhada com o domínio do problema. A solução encontrada está na intervenção do engenheiro de requisitos, que vai validar cada sinônimo. Os sinônimos só serão considerados após passarem pela avaliação do engenheiro de requisitos. Um procedimento parecido acontece para exclusão dos sinônimos de MFs. Os sinônimos só serão excluídos caso o engenheiro de requisitos entenda que eles não estão alinhados com o escopo em estudo.

O objetivo específico 2: Elaborar um grafo de colaborações e/ou de influências das Metas Flexíveis e seus sinônimos, alimentando-o com pesos do ambiente do problema, para que as simulações sejam o mais próximo do real quanto possível; foi modificado em sua forma de implementar, ao invés de trabalharmos com uma rede (grafo) de colaboração, optamos por utilizar parte do conceito da MR que permite conhecer as influências entre as Metas Flexíveis. Em função da modificação na forma de implementar o objetivo específico 2, o objetivo específico 3 - Elaborar tabela de notação dos pesos das influências – não precisa ser implementado, uma vez que, seria complementar do objetivo específico 2 como considerado inicialmente.

Os demais objetivos específicos não sofreram nenhuma modificação e são listados a seguir:

4. Testar a técnica por meio de casos de estudo.

1.3 Organização da dissertação

Nesta subseção é feita uma descrição por capítulos da organização desta dissertação.

A dissertação, além da introdução possui mais cinco capítulos. O Capítulo 2 apresenta o referencial teórico, resultado da revisão bibliográfica realizada. Neste capítulo são discutidas as tecnologias, os métodos e os conceitos necessários para o entendimento e desenvolvimento desta pesquisa. Também são apresentados os trabalhos correlatos, que de alguma forma buscam melhorar o processo de elicitação de MFs ou trabalhos que aplicam algumas das técnicas ou abordagens aqui utilizadas.

No Capítulo 3 é feita a descrição dos procedimentos metodológicos, abordando o processo de escolha das técnicas e do dicionário de sinônimos, além do fluxo seguindo por este trabalho.

No Capítulo 4 são apresentados os esforços em relação a implementação da TEKBS, os trabalhos considerados base para a técnica, a descrição do Método ERi*c e da atividade Elicitar Metas de Atores do Método ERi*c.

No Capítulo 5 é apresentada a aplicação da TEKBS utilizando o Expert Committee System - EC, a análise e discussão dos resultados, além do processo de validação da técnica.

No Capítulo 6 são apresentadas as considerações finais, bem como as principais contribuições disponibilizadas por esta pesquisa. Também são apresentadas sugestões de trabalhos futuros que poderão contribuir para melhoria do trabalho desenvolvido.

E, ao final do documento é apresentado um apêndice contendo o código fonte da implementação da TEKBS, a Base de Dados da i*GET, Cunha (2014), o softgoalFile, contendo as MFs raízes extraídas da Base de Dados da i*Get, a Base de Sinônimos, a Base Macmillan com as MFs válidas, a Base Thesaurus com as MFs válidas e a Base do Wordnet com as MFs válidas.

Capítulo 2

2 REFERENCIAL TEÓRICO

Neste capítulo, são apresentados os conceitos, abordagens e ferramentas que serviram como base para a criação da TEKBS, além dos trabalhos correlacionados. São abordados o processo de elicitação de requisitos; o conceito de MF e de RNF; a técnica de engenharia de requisitos do Léxico Ampliado da Linguagem, a ferramenta Cenário e Léxico que permite uma melhor organização das necessidades dos envolvidos; o processo de extração de sinônimos na Ferramenta online do Wordnet; o Framework i* e a utilização da intencionalidade. Por fim, conceituamos o Sistema de Produção Integrado da Linguagem – CLIPS e sua aplicação.

2.1 A Elicitação de Requisitos

Requisitos de um sistema são considerados descrições das funcionalidades do sistema e suas restrições operacionais. Esses requisitos podem ser considerados uma declaração abstrata, de alto nível de abstração, das funções ou restrições existentes no sistema, além de refletirem as necessidades dos clientes na elaboração de um sistema que auxilie na resolução de algum problema (Sommerville, 2011).

Para melhor entendimento dos requisitos eles foram classificados de duas formas: funcionais – RF - que representam as funcionalidades do sistema, ou não funcionais – RNF - que mostram as propriedades e as restrições existentes no sistema. Um requisito funcional são declarações de alto nível acerca das funcionalidades do sistema (o que ele deve fazer). Por exemplo, para o sistema EC RF poderiam ser: cadastrar revisor, Cadastrar autor, e RNF o tempo de resposta, usabilidade, confiabilidade do sistema.

O Processo de Requisitos – PR - contempla segundo (Leite, 1988) a elicitação, a modelagem e a análise de requisitos. Para (Pacheco e Garcia, 2012) (Sommerville, 2011) (Carrizo, Dieste e Juristo, 2014) o PR tem como fases levantar,

analisar, documentar e verificar as funcionalidades e restrições do software. Para esta pesquisa descreveremos o PR como descrito em (Leite, 1988).

O PR surge da necessidade de melhorar as especificações das funcionalidades dos softwares, uma vez que requisitos deficientes são a maior causa de falhas nos projetos de software (Hofmann e Lehner, 2001) (Sutcliffe e Sawyer, 2013). O PR consiste em um processo sistemático de desenvolvimento de requisitos por meio de um processo iterativo de análise do problema, documentação das observações resultantes e verificação acerca da precisão do atendimento (Christel e Kang, 1992), sendo considerada a parte mais importante no desenvolvimento do ciclo de vida dos projetos de desenvolvimento de software (Shams-Ul-Arif, Khan e Gahyyur, 2009) (Chakraborty, Baowaly, Arefin e Bahar, 2012). Porém, se a ER não for feita de forma eficiente e eficaz, erros irão surgir e se não forem detectados o mais cedo possível, os custos tendem a se elevar cada vez mais o que poderá ocasionar a perda de prazos, o não cumprimento do orçamento, e em alguns casos, a insatisfação do cliente. A importância da PR no desenvolvimento de sistemas de software tem sido estabelecida e reconhecida tanto por pesquisadores como por profissionais de Tecnologia da Informação (Zowghi e Coulin, 2005).

Como exibe a Figura 2.1, frequentemente as fases de elicitação, modelagem e análise ocorrem de forma simultânea e incremental em um processo evolutivo ao longo do ciclo de vida do software (Leite, 1988) (Silva, 2006). Neste trabalho abordaremos a atividade de elicitação.

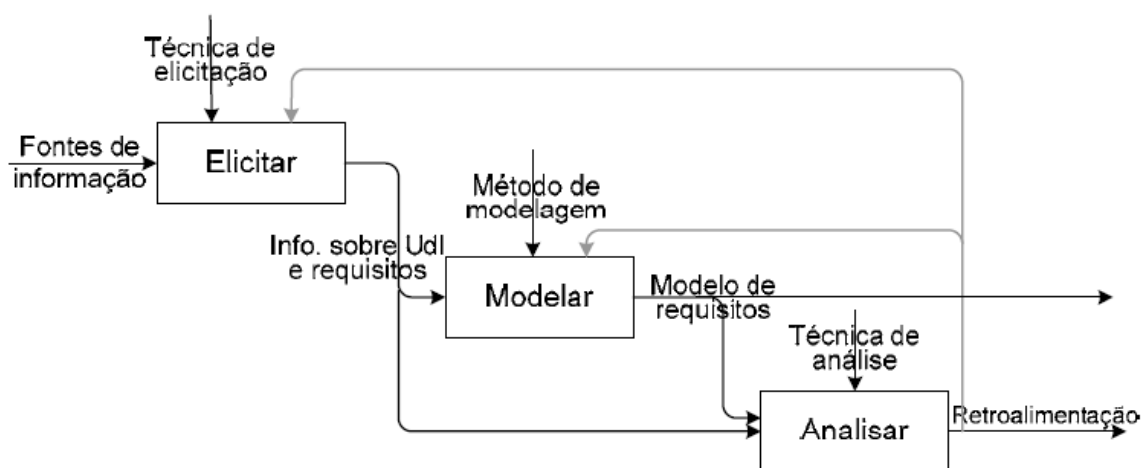


Figura 2.1: Processo de Requisitos.

Fonte: Leite (1988)

A Elicitação é considerada fundamental para o sucesso do desenvolvimento de sistema. A área de elicitação de requisitos é relativamente madura (Sutcliffe e Sawyer, 2013). É a fase onde os engenheiros de requisitos trabalham com os interessados pelo sistema a fim de explorar o contexto do problema (Pohl, 1997). Neste processo o engenheiro de requisitos atua como um investigador.

Para auxiliar o processo de investigação, o engenheiro de requisitos pode utilizar algumas técnicas que facilitam a coleta das informações a ser inseridas no sistema, incluindo as informações de Udi – Universo da Informação – que o restringem (Silva, 2006). Algumas técnicas mais conhecidas e utilizadas envolvem a análise de documentos e sistemas existente, entrevistas, observação, questionários, reuniões, protótipos, além de induzir os envolvidos a falar sobre suas expectativas relacionadas ao produto (Campos, 2009). Contudo, problemas na elicitação de requisitos são um dos maiores desafios no processo de desenvolvimento de software. Isso ocorre porque a falta de levantamento de requisitos causará falhas em todo o projeto (Oliveira, A. P. A., Leite, J. C. S. e Cysneiros, L.M., 2008) (Pa e Zin, 2011).

O trabalho de (Rajagopal et al., 2005), identifica três classes de problemas que podem auxiliar no entendimento dos desafios na elicitação e requisitos:

- **Problema de escopo:** As necessidades das partes interessadas nem sempre são bem definidas, de modo que informações importantes podem não ser expressas e informações desnecessárias podem ser disponibilizadas para o engenheiro de requisitos.
- **Problema de entendimento/compreensão:** Os usuários em muitos casos possuem uma compreensão incompleta do domínio do problema, dificultando a comunicação entre as partes; informações “evidentes” podem ser omitidas; usuários diferentes podem ter necessidades conflitantes ou percepções diferentes; requisitos são muitas vezes expressos com ambiguidade.
- **Problemas de volatilidade:** requisitos evoluem ao longo do tempo, seja pela necessidade de mudança ou por causa da mudança de percepção das partes interessadas.

Um resumo das razões que tornam crítica a elicitação de requisitos é feito por (Farinha e Silva, 2013) com uma modificação para a última classe definida por (Rajagopal et al., 2005). A primeira razão se baseia na comunicação complexa e passível de erros entre as partes interessadas e os engenheiros. A segunda, na dificuldade que as partes interessadas possuem de serem claras sobre suas necessidades e por fim, o não entendimento do contexto do negócio pelos engenheiros de requisitos.

Devido às dificuldades encontradas no processo de elicitação, muitos erros são cometidos, prejudicando a especificação de requisitos (Campos, 2009). Uma boa especificação de requisitos é determinante para a qualidade do produto. Neste caso se um engenheiro de requisitos, analista ou desenvolvedor não compartilham o mesmo entendimento sobre os requisitos, o resultado do processo não irá satisfazer as necessidades das partes interessadas (Alshazly, Elfatry e Abougabal, 2014).

Uma elicitação de requisitos bem elaborada contribui para a qualidade do documento de requisitos. Para o (IEEE, 1998) um bom documento de requisitos precisa ser correto, não ambíguo, consistente, superior pela importância e/ou estabilidade, verificável, modificável e rastreável. Assim, a elicitação de requisitos passa a ser essencial, determinando o sucesso dos projetos de software e a qualidade dos sistemas entregues aos clientes (Nuseibeh e Easterbrook, 2000).

2.2 As Metas Flexíveis

Uma MF é considerada uma qualidade, inicialmente abstrata, tal como segurança, desempenho, transparência e confiabilidade (Chung et al., 2000). Diversas são as pesquisas que consideram MF como sinônimo de RNFs (Chung et al., 2000) (Cysneiros, Yu e Leite, 2003) (Vinay, Aithal e Sudhakara, 2014). Porém, a presente pesquisa adotou o termo em português “Meta Flexível” como sinônimo de softgoal, aqui, considerada, como em (Mylopoulos, Chung e Nixon, 1992) um conceito mais abstrato para representar a intencionalidade dos atores com relação à qualidade.

Metas Flexíveis (softgoals) foram introduzidas em (Mylopoulos, Chung e Nixon, 1992) como um conceito primitivo para modelagem de RNFs. Por isso, (Chung et al., 2000) e (Oliveira, 2008) definem o que vem a ser MF e como ela

difere de um RNF. Para eles, MF é uma espécie de meta que é razoavelmente satisfeita para denotar a falta de precisão na percepção de satisfação e os RNFs, estritamente falando, são frases usadas para designar uma qualidade ou a restrição que o software deve ter. Metas pertencem à intencionalidade dos atores, enquanto RFs e RNFs estão relacionados ao software. Em síntese, MFs representam necessidades que são parte da intencionalidade, RNFs são características de qualidade implementadas por funções de software.

Para (Jureta, Faulkner e Schobbens, 2006) e (Serrano, 2011) uma MF não pode ser claramente atingida. Por exemplo, não é possível dizer que a MF “segurança[conta]” foi totalmente satisfeita, pois depende da interpretação individual de cada cliente. (Serrano, 2011) diz que os RNFs são critérios de qualidade que são esperados pelos envolvidos, são subjetivos e não facilmente mensuráveis por meio de métricas, porém precisam ser considerados no desenvolvimento do software.

Segundo (Rawlins e Chung, 1999) e (Serrano, 2011), uma análise baseada em MF pode ser um complemento útil para abordagens baseadas nos requisitos funcionais. A não elicitação de uma MF ou sua elicitação tardia pode ser tão crítica para o sucesso de um sistema quanto a não elicitação de um requisito funcional. Por exemplo, um sistema de reservas de passagens em companhias aéreas pode fornecer toda a funcionalidade necessária, mas se a MF de tempo de resposta não for cumprida ele poderá fracassar. O mesmo pode ser observado para sistemas controladores de vôos, sistemas médicos, sistemas de gestão de produção, entre outros.

Para (Chung et al., 2000) e (Rawlins e Chung, 1999) MFs são expressas usando uma notação específica (tipo [tópico]) e deve refletir as informações necessárias para seu entendimento pelo engenheiro de software. Dessa forma ela possui um tipo, que deve ser um atributo de qualidade e possui um tópico, que determina em que característica do sistema se espera aquela qualidade (tipo).

2.3 O Framework I*

O Framework i* (iStar), é uma técnica de referência na área de Engenharia de Requisitos Orientada a Metas (Goal Oriented Requirement Engineering – GORE), cujo o foco é a intencionalidade.

O Framework i* foi introduzido por (Yu, 1995). Sua intenção foi elaborar uma técnica de modelagem organizacional que fosse mais expressiva para descrição de requisitos que envolvessem muitos atores e que pudesse ser empregada nas primeiras fases do ciclo de vida do software. A maioria das técnicas existentes concentra-se em aspectos comportamentais do processo. Uma grande lacuna em diversas técnicas é a capacidade de descrever questões motivacionais do processo, como “o porquê - whys” de se ter escolhido determinada opção de modelagem.

A característica de descrição das razões envolvidas no processo de elicitação é de extrema importância na aquisição de conhecimento sobre o domínio. A grande maioria dos modelos convencionais não consegue expressar tais razões, por estarem estritamente ligadas à descrição de entidades, de atividades e seus relacionamentos.

No Framework i*, as entidades são caracterizadas por atores que se relacionam via uma relação de dependência, a fim de alcançar seus objetivos. Atores podem ser humanos, hardware ou software ou suas combinações. Eles são autônomos não tendo comportamentos totalmente controláveis, nem perfeitamente conhecidos. Para manter a relação de dependência, tais atores podem consumir ou produzir recursos com propósito de ter suas metas atingidas (Yu 2009).

O framework propõe dois tipos de diagramas para a modelagem, cada um corresponde a um nível diferente de abstração: Modelo de Dependência Estratégica – Modelo SD e o Modelo de Razão Estratégica – Modelo SR que são descritos nos subitens 2.3.1 e 2.3.2.

O i* é utilizado para modelagem em diversos contextos como pode ser visto em (Colomer, López, Cares e Franch, 2011), (Cysneiros, 2013) e (Monsalve e Leite, 2013) e, em função dessa vasta utilização na comunidade acadêmica e de acordo com (Franch, 2012), ele pode ser considerado um framework com boa maturidade em função do número elevado de publicações em diversos eventos, a criação de um evento específico – International i* Workshop, a sua disseminação na comunidade de engenharia de requisitos, a criação de um livro e ao processo de padronização.

2.3.1 Modelo SD

No Modelo SD somente são expressos relacionamentos de dependência estratégica, sendo o modelo baseado em um conjunto de nós e arestas onde os nós representam os atores e as arestas definem a dependência entre eles. Cada dependência chamada de *dependum* representa um relacionamento de cooperação entre os atores, onde um ator conhecido como *dependor* depende de outro chamado de *dependee*, a fim de obter ou alcançar algum objetivo. O objetivo a ser alcançado é um elemento intencional que pode ser um recurso, uma tarefa, Meta Concreta - MC ou MF (Yu, 1995). Uma MF representa um atributo de qualidade ou meta que normalmente não possui um consenso de como a atividade pode ser executada, ficando na dependência da interpretação de cada envolvido (Cares, 2012) enquanto as MC, as tarefas e os recursos estão relacionados com as funcionalidades do sistema (Lucena, Silva, Santos, Alencar e Castro, 2009).

As relações de dependências podem apresentar vulnerabilidade quando o *dependee* não cumpre a sua parte, sendo assim é possível definir a importância – força - de cada ator envolvido utilizando três categorias: *open*, *committed* e *critical* (Cares, 2012). A Figura 2.2 exibe os quatros tipos dependências existentes no Modelo SD.

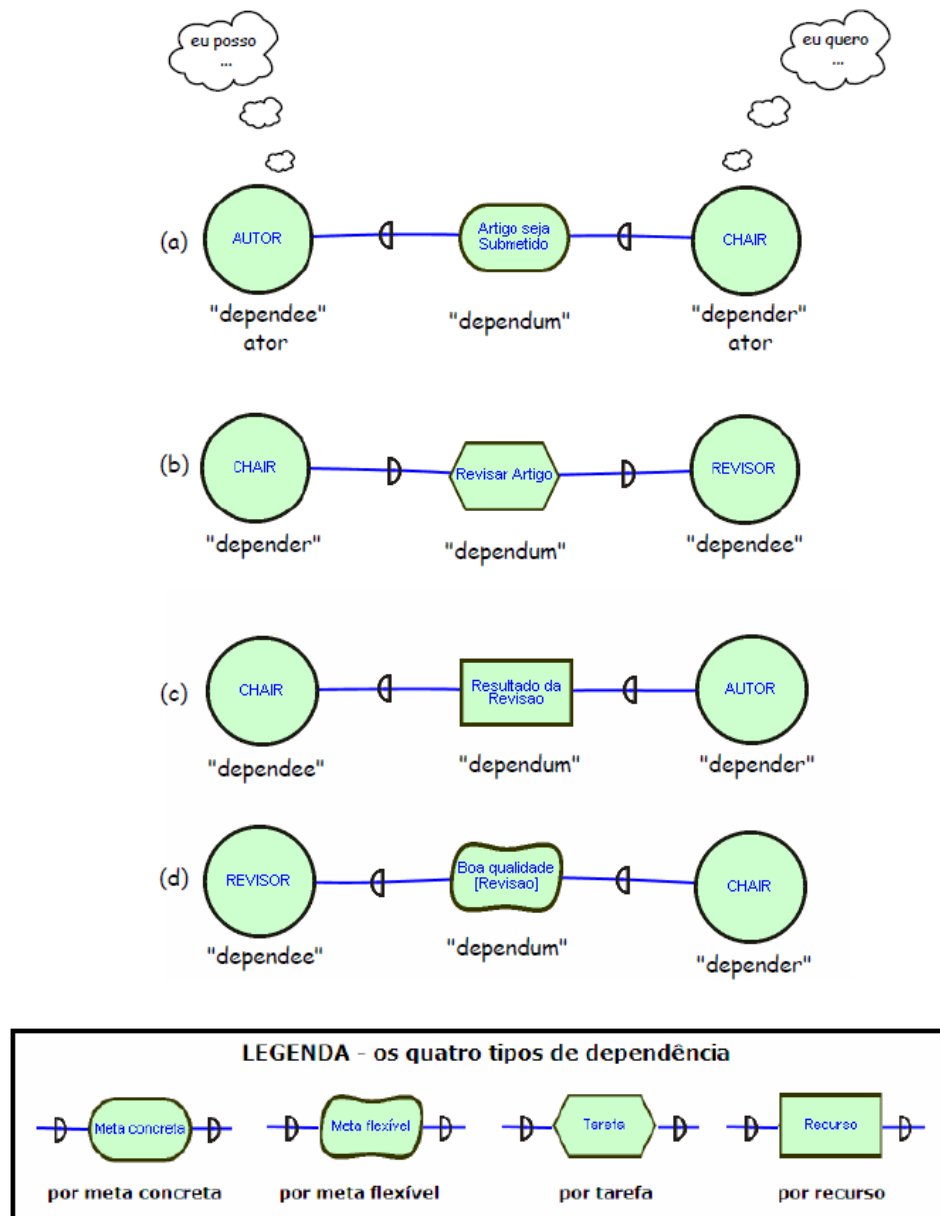


Figura 2.2: Tipos de dependência entre atores no Modelo SD

Fonte: (Napolitano, 2009)

Para (Yu, 1995) (Oliveira, 2008) (Napolitano, 2009) os tipos de dependências apresentados na Figura 2.2 refletem o grau de liberdade que existe no relacionamento.

Na Figura 2.2 parte (a) ilustra a dependência por meta entre dois atores: um ator “quer” alguma coisa que outro ator “pode” suprir. Assim, o CHAIR da conferência (depender) depende do AUTOR (dependee) para que “Artigo seja submetido”. A Figura 2.2 parte (b) faz a representação de uma dependência por tarefa. O ator CHAIR (depender) depende de o ator REVISOR (dependee) executar a

tarefa “Revisar Artigo”. A Figura 2.2 parte (c) mostra a representação de dependência por recurso. O AUTOR (dependor) depende do CHAIR (dependee) fornecer o “Resultado da Revisão”. Concluindo, a Figura 2.2 parte (d) ilustra a representação de uma dependência por MF. O ator CHAIR (dependor) depende do ator REVISOR (dependee) executar a revisão com boa qualidade (Oliveira, 2008).

Na Figura 2.2 a “letra D” mostra a direção de dependência. Sendo representados os quatro tipos de dependências estratégicas: (a) por meta, (b) por tarefa, (c) por recurso e (d) por meta flexível.

2.3.2 Modelo SR

O Modelo SR descreve a intencionalidade interna de cada ator em função do elemento do processo, descreve as razões que estão implícitas e as que estão associadas a um sistema por meio da modelagem de seus atores, das MCs ou MFs que precisam atingir e como elas serão atingidas, por metas ou recursos (Silva, Borba e Castro, 2011).

Ele fornece um nível detalhado para modelar relações intencionais internas dos atores. Elementos intencionais, metas, tarefas, recursos e metas flexíveis, aparecem em Modelos SR não somente como dependências externas, mas também como elementos internos organizados em sua maioria em estruturas hierárquicas com relacionamentos meio-fim, decomposição de tarefas e relacionamento de contribuição (Yu, Liu e Li, 2001), como exibe a Figura 2.3.

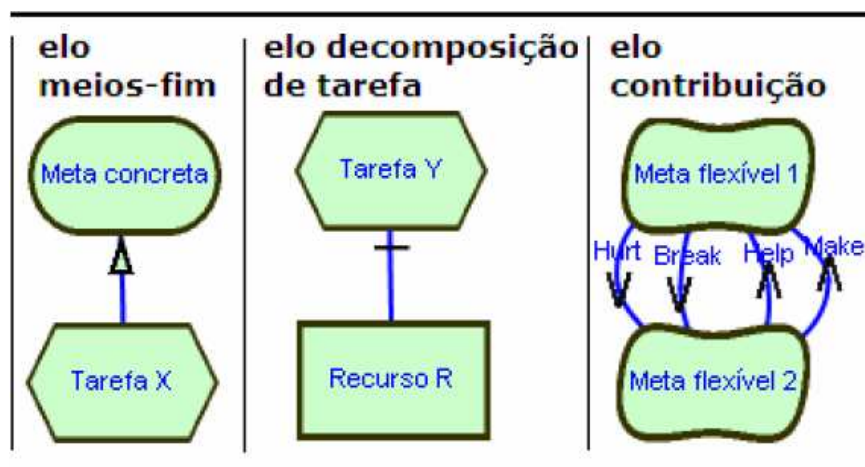


Figura 2.3: Relacionamentos intencionais internos aos atores.
Fonte:(Oliveira, 2008)

O relacionamento Meios-fim utiliza o link meios-fim (means-end) para estabelecer uma relação onde pode existir um ou mais elementos intencionais que são meios que auxiliam para a realização ou alcance de um fim⁴. O “fim” pode ser uma meta a ser alcançada, tarefa a ser desempenhada, um recurso a ser produzido ou meta-flexível a ser satisfeita “a contento”, enquanto que os meios tendem a ser uma tarefa (Napolitano, 2009). Na decomposição de tarefas o link Task-decomposition indica a decomposição de uma tarefa em diferentes elementos intencionais. Existe uma relação de “E⁵” quando a tarefa é decomposta em mais de um elemento intencional. E por fim, o relacionamento de contribuição que está voltado para as MFs. Ele exhibe o tipo de contribuição entre duas ou mais MFs, podendo a contribuição ser, segundo (Chung e Nixon, 1995), positiva (“+”), negativa (“-”) e neutra (“?”) e para (Chung et al., 2000) a escala deixa de ser de 3 tipos e passa a possuir 5 tipos: make (“++”), help (“+”), unknown (“?”), hurt (“-”) e break (“--”). Neste estudo adotaremos a descrita por (Chung e Nixon, 1995).

A Figura 2.4 retirada de (Oliveira, 2008) exemplifica um Modelo SR. Como não é possível a representação total do modelo em uma folha comum, utiliza-se parte de um modelo com alguns atores e as respectivas razões. Definida por Oliveira como uma Situação de Dependência Estratégica – SDsituation.

No diagrama da Figura 2.4, parte do “rationale” de dois atores é exibida. Nele é possível perceber a presença de dependências estratégicas e que existem diversos exemplos dos elementos de relacionamento: elo meios-fim, decomposição de tarefa e contribuição entre MFs. Na análise o ator “Autor” tem por meta que seu “Artigo seja publicado” e para alcançar essa meta (fim) o meio é “Submeter artigo”. Essa tarefa tem em sua decomposição duas Metas Flexíveis chamadas de “Comodidade [submissão]” e “Segurança[submissão]” e três metas concretas sendo elas: “Artigo seja enviado”, “Artigo seja aceito” e “CameraReady seja entregue”. Como a segurança reduz a comodidade da realização da tarefa foi inserida a contribuição negativa “hurt”. Para que a meta “Artigo seja publicado” seja atingida é preciso cumprir todas das demais tarefas, MCs e MFs envolvido o recurso “Artigo”.

⁴ Relação lógica do ou...ou. Onde um ou outro satisfaz a sentença.

⁵ Relação lógica do E, onde para satisfazer à sentença todas as tarefas precisam ser cumpridas.

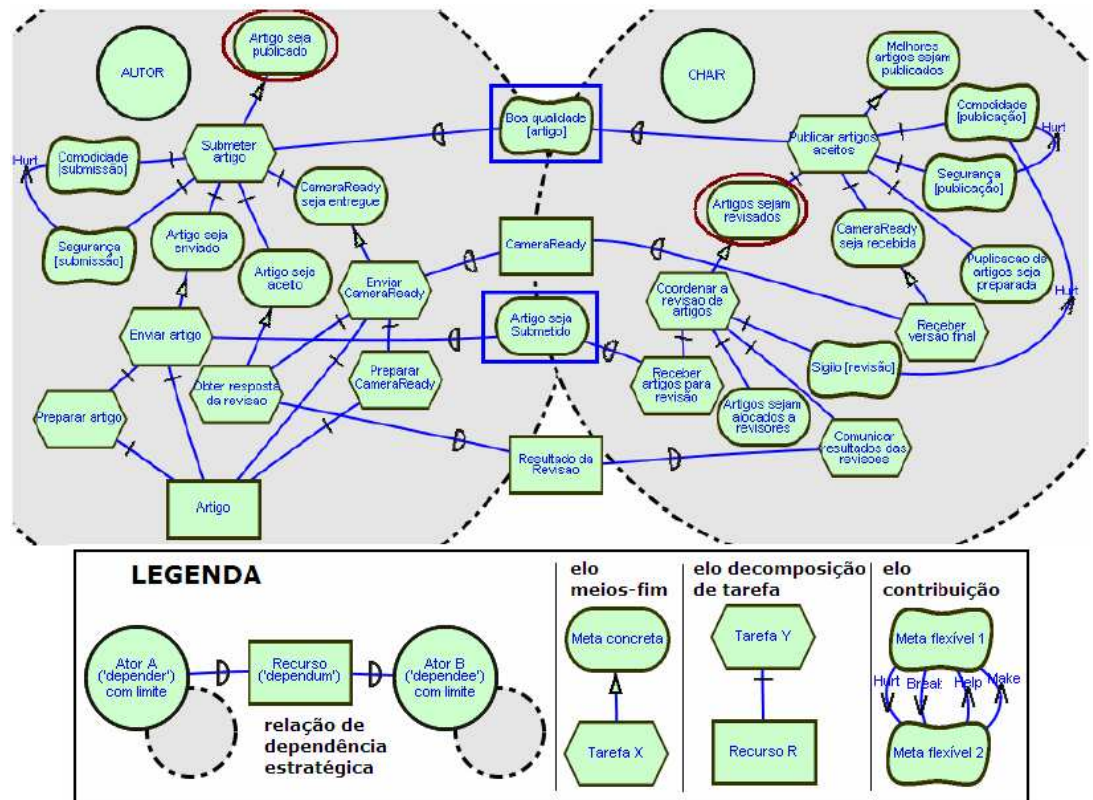


Figura 2.4: Exemplo do Modelo SR (Parte do Modelo)
 Fonte: (Oliveira, 2008)

2.4 O Léxico Ampliado de Linguagem – LAL

O LAL, ver Figura 2.5, é um glossário que permite descrever a linguagem de domínio de uma aplicação utilizando linguagem natural. Ele é composto por símbolos que são identificados por nomes, representados por duas descrições, uma chamada de noção e a outra de impacto ou resposta comportamental. A noção explica o significado literal do símbolo que são significados equivalentes encontrados nos dicionários, e o impacto descreve os efeitos do uso ou ocorrência do símbolo no domínio sob estudo, isto é, informação extra sobre efeitos comportamentais do símbolo no contexto do Universo de Informação (UdI) que é composto por todas as fontes de informação e pessoas relacionadas ao sistema.

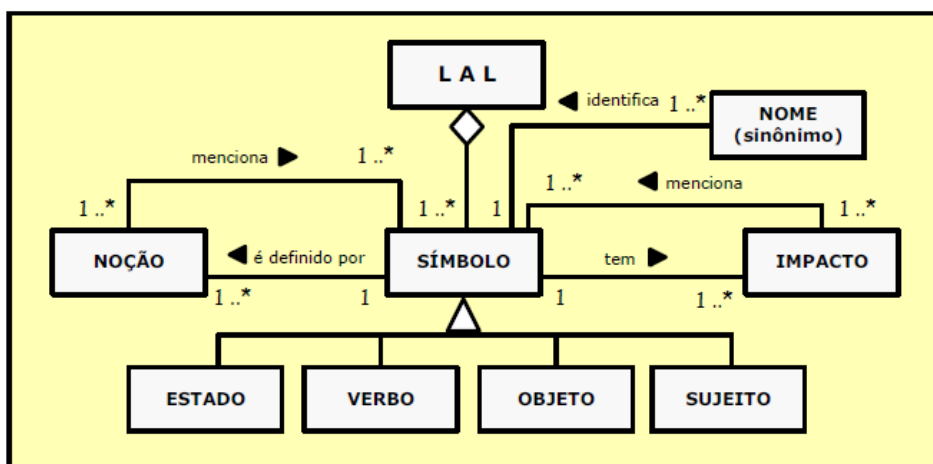


Figura 2.5: Os elementos do LAL representados em um diagrama de classes.

Fonte: (Oliveira, 2008)

Com o intuito de evitar representações prolongadas, duas regras básicas devem ser seguidas pelo engenheiro de requisitos na construção de um LAL. Uma é baseada nos princípios de circularidade e a outro no princípio do vocabulário mínimo. A circularidade descreve a ideia de possuir termos em função dos já existentes, sendo recomendado maximizar o uso de outros termos do léxico em outras descrições. O vocabulário mínimo reduz o uso de palavras fora do léxico, isto é, quanto menos termos fora do contexto da aplicação forem usados nas descrições melhor.

Para (Oliveira, 2008) os dicionários, de forma geral, representam somente a denotação dos termos e o LAL por meio dos impactos, representa também os relacionamentos comportamentais onde estão presentes os símbolos. Por isso, a representação no LAL se torna mais rica e com mais detalhes do UdI.

Para a construção do LAL é preciso seguir seis passos: 1) identificar as fontes de informações do UdI; 2) identificar a lista de símbolos relevantes no UdI; 3) classificar gramaticalmente os símbolos do léxico utilizando as seguintes heurísticas: se o símbolo pratica uma ação, ele é classificado como sujeito; se sofre a ação, é classificado como objeto; se é uma situação em um dado momento, é classificado como estado; se representa uma ação, ele é classificado como verbo (Oliveira, Braga, Lana e Cunha, 2013); 4) descrever os símbolos por meio da noção e do impacto garantindo os princípios do LAL; 5) verificar o LAL por meio da inspeção; e 6)

validar o LAL com os atores do UdI. Seguindo esses passos é possível ter uma descrição clara e de simples entendimento.

O LAL é uma ferramenta muito útil para pessoas sem habilidades técnicas, contudo, pessoas com tais habilidades podem obter maior benefício com sua utilização. A utilidade do LAL vem de três importantes características: é fácil aprender, é fácil de usar e possui boa expressividade com a validação de domínios complexos (Antonelli, Rossi, Leite e Oliveros, 2013).

Para facilitar o entendimento dos envolvidos e do engenheiro de requisitos, o domínio em linguagem natural é estruturado no LAL por meio de uma ferramenta que gerencia Cenários e Léxicos - C&L⁶. A C&L é uma ferramenta colaborativa desenvolvida pelo Grupo de Engenharia de Requisitos da PUC-RIO. A ferramenta utiliza a representação que ao mesmo tempo em que facilita a compreensão utilizando a linguagem natural, força a organização da informação por meio de uma estrutura bem definida (Felicíssimo, Leite, Breitman e Silva, 2004)

Segundo (Leite e Franco, 1993) (Oliveira et al., 2013) dentro dessa estrutura o LAL implementa uma ideia muito simples, “entender a linguagem utilizada pela aplicação sem se preocupar com a compreensão do problema envolvido”. O foco principal é capturar o vocabulário do contexto organizacional.

Devido a essas características, o LAL tem sido aplicado nas primeiras fases do desenvolvimento de software. Sua adoção facilita o entendimento dos envolvidos, seja no processo de definição do problema ou na necessidade de verificação e validação do usuário. O LAL melhora o entendimento da equipe técnica com relação à área de domínio, facilita a compreensão das alterações sofridas pelos requisitos ao longo do projeto, permitindo ao engenheiro de requisitos a identificação dos impactos de mudanças por meio da rastreabilidade (Felicíssimo et al., 2004).

A geração do léxico ocorre mediante a aplicação de técnicas de elicitación como pode ser visto em (Goguen e Linde, 1993).

2.5 O Wordnet – A lexical database for english

O WordNet é um dos tesouros de língua inglesa mais populares disponíveis na comunidade acadêmica. Ele é um dicionário legível por máquina desenvolvido na

⁶ Disponível em <http://pes.inf.puc-rio.br/cel/>.

Princeton University por Georgio Miller e seus colegas do grupo de Ciências Cognitivas (Miller, 1995) (Fellbaum, 1998) (Matsuoka e Lepage, 2011) (Kaur, 2014). Seu léxico é dividido em categorias distintas sendo elas substantivos, adjetivos, verbos e advérbios que suprem a necessidade de estabelecer uma semântica precisa para diferentes termos ou conceitos.

Para cada uma das categorias, existe um conjunto específico de arquivos que está estruturado de forma a facilitar a consulta de dados. Para os verbos, por exemplo, existe um arquivo de índices onde é feita a busca do termo desejado. Após a localização são extraídos deste arquivo os synsets ou grupos de sinônimos que estão ligados ao termo. Com essa informação o programa do WordNet navega por outros arquivos contendo synsets e relações (Torres e Braga, 2002) (Miller, 1995).

O Wordnet está disponível em duas versões, uma para ser utilizada em desktop e a outra na Web. Neste trabalho optamos por extrair os sinônimos da versão Web que vai permitir manter a base de sinônimos atualizada e crescente. Para a extração dos sinônimos na versão online utilizamos o Jsoup (Hedley, 2009), uma biblioteca JAVA que permite trabalhar com Hiper Text Markup Language - HTML em tempo real. O Jsoup fornece uma API – Application Programming Interface – para extrair e manipular dados, sendo possível a leitura da página HTML e a criação do arquivo na extensão do CLIPS o “.clp” que será carregado na TEKBS permitindo a aplicação das regras e a extração das informações necessárias para compor a Base de Sinônimos.

2.6 Sistema de Produção Integrado da Linguagem C - CLIPS

Para a elicitación das MFs foram utilizadas técnicas de inteligência artificial associados a sinonímia. O ambiente de desenvolvimento escolhido foi o CLIPS Language Integration Production System que foi utilizado para representar a base de conhecimentos sobre o problema, emprestando a máquina de inferência para auxiliar o engenheiro de requisitos na elicitación (Giarratano, 1993).

A Linguagem CLIPS foi criada anterior a 1984 pelo Centro Espacial Johnson da NASA, atualmente em sua versão 6.3 o CLIPS é mantido como software de domínio público pelos seus principais autores que não trabalham mais na NASA, por essa característica existe voluntários que auxiliam na manutenção e avanço da linguagem (Riley, 2013).

O CLIPS é baseado em fatos - memória de trabalho ou base de fatos – que neste trabalho são representados pelas MFs e seus sinônimos e em regras conhecidas como base de conhecimento. O CLIPS apesar de fornecer recursos básicos para construção de interfaces simples e primitivas é considerada a escolha mais popular dos engenheiros do conhecimento (Durkin Jack e Durkin John, 1998). Esta popularidade está atrelada ao grande número de pesquisas exploratórias em áreas nas quais há pouco conhecimento acumulado e sistematizado. Algumas vantagens e características da utilização de sistemas especialistas baseados em regras podem ser encontradas em (Negnevitsky, 2005) (Riley, 2013).

Uma Base de Fatos contém “fatos problemas” que podem ser inseridos ou ser inferidos por disparo de alguma regra. Porém para que os fatos sejam inseridos é necessário estruturar todas as informações que serão processadas pelas regras, nesse caso as informações do LAL foram estruturadas seguindo o mesmo padrão de (Cunha, 2014). A estruturação tem por objetivo formatar as informações do LAL para que seja possível a investigação das descrições e a extração das informações.

2.7 Trabalhos Correlatos

- a) A abordagem de (Sen e Jain, 2007) propõe um método baseado em técnica ágil utilizando refinamento de metas para elicitar MFs dentro do GORE. A proposta do método envolve alta participação dos interessados. As MFs são elicidadas pelas partes interessadas das metas de alto nível por meio do processo de refinamento de metas envolvendo agentes. As MFs elicidadas pelo método de (Sen e Jain, 2007) são compiladas por meio do programa Activity Card Compiler. O método é utilizado para elicitar MFs que são utilizadas no desenvolvimento de um aplicativo de notícias baseado na web. A Experimentação realizada pelos autores mostra que após as inúmeras iterações várias MFs são elicidadas criando um conjunto de metas completo, preciso e que pode ser utilizado para a criação do documento de Especificação dos Requisitos de Software, ou Software Requirements Specification - SRS com melhor qualidade. As estratégias de (Sen e Jain, 2007) e a apresentada neste trabalho possuem objetivos semelhantes, pois ambos os processos, apesar de

serem completamente diferentes, buscam facilitar o processo de elicitación de MFs.

- b) Em (Castañeda, Ballejos e Caliusco, 2012) foi utilizada uma abordagem baseada em tecnologia da Web semântica para melhorar a qualidade de uma SRS. Essa abordagem tem como base a ontologia e o LAL que foram utilizados para criar a OntoSRS e o OntoLEL que têm como função preservar a consistência, a exatidão, a rastreabilidade, a organização e a não ambiguidade de um SRS produzido pela dificuldade no processo de elicitación de metas vindas das diferentes interpretações do domínio. Em (Castañeda, Ballejos e Caliusco, 2012) as autoras procuram contribuir com um processo de especificação orientado para a obtenção de um SRS de qualidade. A abordagem apresentada em (Castañeda, Ballejos e Caliusco, 2012) difere da abordagem aqui descrita pelo fato de não elicitar MFs e sim alguns RNFs. Por outro lado, as duas abordagens utilizam o LAL para facilitar o entendimento da linguagem natural e os sinônimos para preservar a não ambiguidade.
- c) No trabalho desenvolvido por (Baía e Braga, 2013) buscou-se verificar a presença de atributos de transparência em modelos de requisitos de software representados com o Framework i^* . Para a análise foram utilizados os sinônimos dos atributos de transparência, extraídos do Wordnet, juntamente com as regras de inferência do CLIPS. Essa abordagem permitiu a identificação de novos atributos de transparência que estão relacionados com os conceitos do i^* e a identificação dos atributos que não possuem vínculo com o conceito. A abordagem baseada em sistema de inferência utilizada por (Baía e Braga, 2013) permitiu a obtenção de um sistema flexível e expansível capaz de verificar especificações de requisitos e sugerir a inclusão de novos atributos em especificações analisadas pelo sistema. O trabalho de (Baía e Braga, 2013) possui interseção com o presente trabalho. Ambos utilizam o CLIPS e a sinonímia, porém neste trabalho, o CLIPS é utilizado para auxiliar na elicitación de MFs por meio da utilização de sinônimos. Facilitando a elicitación e permitindo testar, por meio de intervenção do engenheiro de requisitos, se um sinônimo encontrado pela técnica é de fato relevante no contexto do domínio da aplicação, evitando assim a associação de um sinônimo fora do contexto do problema a alguma MF já elicitada.

Capítulo 3

3 PROCEDIMENTOS METODOLÓGICOS

Neste Capítulo apresentaremos a abordagem metodológica utilizada no desenvolvimento desta pesquisa, incluindo o processo de escolha das ferramentas, do dicionário de sinônimos e o fluxo de trabalho utilizado no desenvolvimento da pesquisa.

3.1 Aspectos Metodológicos: Caracterização da pesquisa

Este trabalho utiliza a investigação exploratória planejada por meio de delineamentos da pesquisa qualitativa e da pesquisa aplicada ou tecnológica utilizando procedimentos que envolvem a pesquisa bibliográfica e a documental para criação da técnica TEKBS.

A investigação exploratória tem por objetivo buscar uma descoberta em área na qual há pouco conhecimento acumulado e sistematizado que conforme Gil e Wazlawick (Gil, 2010) e (Wazlawick, 2010) tem como propósito, proporcionar maior familiaridade com o problema, com vistas a torná-lo mais explícito ou construir hipóteses.

O delineamento da pesquisa qualitativa e da pesquisa aplicada (ou tecnológica) aconteceu em função da necessidade de extrair dados e informações visíveis ou latentes e, por meio dessas informações, gerar novos conhecimentos (Godoy, 1995) (Chizzotti, 2003) visando realizar descobertas que pudessem ser imediatamente aplicadas para produzir algum tipo de ganho ou melhoria (Wazlawick, 2010).

No que diz respeito aos procedimentos técnicos foi considerado um estudo não experimental, onde todos os dados e informações foram coletadas na literatura, não tendo a intervenção sistemática dos pesquisadores (Wazlawick, 2010).

A pesquisa bibliográfica foi de fundamental importância para o aprofundamento do construto em estudo “Elicitação de Metas Flexíveis”. Por meio da pesquisa documental e da bibliográfica foi possível entender quais as possíveis

necessidades a serem atendidas e como poderiam ser atendidas e quais os procedimentos, ferramentas e linguagem mais adequadas para a elaboração da TEBKS.

3.2 Processo de escolha das ferramentas

Para escolher as ferramentas o primeiro passo foi entender o processo de trabalho do Método ERi*c (Oliveira, 2008), uma vez que a TEKBS foi desenvolvida como sua extensão. Assim, a escolha das técnicas que utilizam intencionalidade como o LAL e o Framework i* foram feitas em função do Método ERi*c que já utilizava as duas técnicas em sua estrutura mantendo um alinhamento entre a base e a extensão.

Em seguida analisou-se a i*Get desenvolvida por (Cunha, 2014). Nela (Cunha, 2014) utiliza o CLIPS como ferramenta de implementação do sistema especialista e apesar da TEKBS utilizar dados fornecidos pela i*Get a escolha pela utilização do CLIPS não se apoiou somente no alinhamento da i*Get e da TEKBS, mas também pelo conjunto de funcionalidades oferecidas pelo CLIPS e que não eram visíveis de forma conjunta em nenhuma das demais ferramentas analisadas para construção de sistemas especialistas.

Juntamente com o CLIPS foram estudadas três outras ferramentas: a) Expert Sinta (Sinta, 1996) (Spirandelli, Santos, Rodrigues e Bandos, 2011); b) SPIRIT (Pieritz, 2003) (Spirit, 2014) e o c) JESS (Friedman, 2003) (Friedman-Hill, 2003) (Friedman-Hill, 2013). Nessa análise, buscamos por uma ferramenta que trabalhasse com representação do conhecimento e nessa primeira condição todas as ferramentas estudadas se encaixavam, contudo o CLIPS se destacou porque além de permitir programação em forma de regras ele permitia duas outras formas de programação a orientada a objetos e utilizando procedimentos do tipo de função e sub-rotinas.

Posteriormente, verificou-se a portabilidade, facilidade de integração e a extensibilidade. Esta análise foi importante, pois poder-se-ia trabalhar em plataformas diferentes e como seria necessário acesso externo para extração dos sinônimos, necessitaríamos de mecanismos que permitissem além do acesso à integração dos sinônimos a ferramenta utilizada. Em relação a portabilidade todas foram satisfatórias, porém na facilidade de integração e extensibilidade não

conseguiu localizar informações relacionadas a Expert Sinta, com as demais existem diversas bibliotecas que permitem a integração e a extensibilidade.

Outras quatro condições avaliadas foram: desenvolvimento interativo, verificação e validação, totalmente documentada, ser software livre com atualizações frequentes. Considerando o desenvolvimento interativo todas as ferramentas possuem de alguma forma interatividade sendo que o CLIPS além da interatividade com o usuário ele possui: (1) ambiente orientado por textos; (2) ferramentas capazes de verificar os códigos implementado e (3) sistema online de ajuda e editor próprio.

Na análise da verificação e validação todas possuem mecanismos que de alguma forma atendem a este item. Em relação a ser totalmente documentada somente o CLIPS, o SPIRIT e o JESS possuem essa informação explícita, contudo a grande maioria da documentação do SPIRIT está em alemão o que dificulta o acesso e o Expert Sinta é um sistema que foi descontinuado e com pouca documentação.

A última condição avaliada é ser software livre e possuir atualizações frequentes. Em referência a essa condição temos o CLIPS e o JESS com atualizações frequentes, o SPIRIT não foi possível localizar essa informação e o Expert Sinta como já mencionado foi descontinuado. Das ferramentas analisadas somente o JESS não é um software livre.

Diante deste contexto e da possibilidade de integração entre a i*Get (Cunha, 2014) e a TEKBS escolheu-se utilizar o CLIPS. A Figura 3.1 ilustra as comparações.

Tabela 3.1 Comparação de Ferramentas

Condições Analisadas	CLIPS	Expert Sinta	SPIRIT	JESS
Representação do conhecimento	X	X	X	X
Postabilidade	X	X	X	X
Facilidade de integração	X	-	X	X
Extensibilidade	X	-	X	X

Continua.

Continuação da Tabela 3.1.

Condições Analisadas	CLIPS	Expert Sinta	SPIRIT	JESS
Desenvolvimento Interativo	X	X	X	X
Verificação e Validação	X	X	X	X
Totalmente documentada	X	-	X	X
Software Livre	X	X	X	-
Atualizações Frequentes	X	-	-	X

3.3 Escolha do dicionário de sinônimos

O processo de escolha do dicionário de sinônimos também foi baseado em comparação. Neste caso analisou-se o Wordnet (Wordnet, 2014), o Macmillan com Thesaurus (Macmillan, 2014) e o próprio Thesaurus (Thesaurus, 2014). Nesta análise foram considerados as seguintes características de avaliação: o tipo de acesso, a facilidade de aplicação, licença de utilização, integração com outras linguagens, documentação e o processamento de linguagem natural.

Na característica tipo de acesso, foi verificada a forma de disponibilização dos dicionários, se web ou desktop. Para a facilidade de aplicação analisou-se o nível de dificuldade de utilização dos tipos de acesso. Para isso foi necessário avaliar o tipo de licença se era software livre ou pago e se era possível integrar o dicionário com outras linguagens.

Por fim, analisou a existência de documentação e se existia algum mecanismo de processamento de linguagem natural. Essa última avaliação foi feita por recebermos descrição dos domínios em linguagem natural e apesar deste trabalho não utilizar o processamento da linguagem natural por receber dados já estruturados, no processo de automatização completo do Método ERI*c, pode ser necessário sua utilização.

A Tabela 3.2 mostra a comparação efetuada entre os dicionários.

Tabela 3.2: Comparação dos dicionários de sinônimos.

Descrição	Dicionários		
	Wordnet	Macmillan	Thesaurus
Tipo de acesso	Desktop/web	Web	Web
Facilidade de aplicação	Sim	-	-
Licença de utilização	Livre	Livre com restrição	Livre com restrição
Integração com outras linguagens	Sim	Sim	-
Documentação	Sim	Sim	-
Processamento de linguagem natural	Sim	Não	Não
Significado do sinônimo no aplicativo	Não	-	-

Dos três dicionários analisados somente o Wordnet possui aplicativo web e desktop o que facilita a utilização para ambientes com alguma restrição de acesso à internet. Se for analisado somente o acesso web, todos os três dicionários são online, mas somente o Wordnet e o Macmillan possuem APIs que permite a integração do dicionário com outras plataformas. Na análise do Thesaurus não foi possível localizar muita informação, uma vez que é o único dicionário que para acessos mais detalhados é necessário pagar uma taxa de acesso. Porém, todas as páginas Webs podem ser consultadas livremente, sem custo.

Quando se analisou a facilidade de aplicação do Macmillan não foi possível obter muitas informações, pois o Macmillan necessita de código de acesso que é fornecido pelo suporte do dicionário. Inúmeras foram as tentativas de solicitação do código de acesso e não obtivemos sucesso.

Um ponto importante de todos os dicionários está no retorno dos sinônimos com seus significados o que direciona a busca dos sinônimos, contudo quando

utiliza-se a API, no caso do Wordnet, essa funcionalidade deixa de existir sendo retornado diversos sinônimos que podem não ter o sentido desejado.

Apesar de não fornecer o significado quando se utiliza a API optou-se por utilizar o Wordnet, pois além de ser o dicionário que possui maior facilidade de aplicação e documentação, ele é largamente utilizado no meio acadêmico (Richardson, Smeaton e Murphy, 1994) (Torres e Braga, 2002) (Kamps, Marx, Mokken e De Rijke, 2004) (Liu, Scheuermann, Li e Zhu, 2007) (Matsuoka e Lepage, 2011) (Baía e Braga, 2013) (Araújo, Lana, Paula, Meira e Vidigal, 2010) (Kaur, 2014) e é o único que processa linguagem natural.

3.4 Fluxo de trabalho utilizado no desenvolvimento da pesquisa

Buscando atingir os objetivos propostos da pesquisa, foram realizadas as seguintes atividades, ver Figura 3.1:

- 1. Revisão bibliográfica:** No início, foi realizada uma ampla revisão bibliográfica para aprofundar os conhecimentos acerca do contexto em que a pesquisa está inserida. Foram exploradas: as técnicas de elicitação de requisitos, o conceito de MF e sua diferença para um Requisito Não Funcional – RNF, as abordagens que utilizam intencionalidade, como o Framework i^* e o Léxico Ampliado de Linguagem – LAL. Além disso, fez-se um estudo do dicionário de língua inglesa – Wordnet (Fellbaum, 1998), utilizado na busca de sinônimos e do Sistema de Produção Integrado da Linguagem C – CLIPS (Riley, 2013), usado para implementação da técnica. Também foram feitas investigações em busca de trabalhos que tinham alguma relação direta ou indireta com a proposta aqui descrita. Diversos artigos científicos, teses e livros que abordam assuntos relacionados foram lidos e serão apresentados ao longo desta dissertação.
- 2. Procedimentos metodológicos:** Foram analisados diferentes procedimentos metodológicos com a finalidade de caracterizar a pesquisa. Após a caracterização realizou-se a descrição das escolhas das ferramentas utilizadas nesta pesquisa e por fim, analisadas as técnicas utilizadas em (Oliveira, 2008), que descreve o Método ERi^*c , e a i^*Get (Cunha, 2014) que serão descritos no Capítulo 4.

- 3. Proposta da Técnica:** Para facilitar o entendimento da construção da técnica proposta, foi feito um resumo dos trabalhos de (Oliveira, 2008) e de (Cunha, 2014) considerados bases deste trabalho e uma descrição do Método ERI*c, mais especificamente da primeira atividade chamada Elicitar Metas de Atores. Para a criação da TEKBS foram implementadas diversas regras e funções dentro do ambiente da ferramenta CLIPS, considerando os conceitos de intencionalidade do Framework i* e a abordagem utilizada no LAL, ambos baseados na primeira fase do Método ERI*c e no produto da implementação criada por (Cunha, 2014). A TEKBS apesar de ser descrita neste documento como um fluxo linear, foi implementada em módulos o que proporciona flexibilidade de uso. Para localizar os sinônimos na base do Wordnet, foi desenvolvido o aplicativo JAVAS utilizando a biblioteca Jsoup (Hedley, 2009) (Xu e Yan, 2011) (Bhalalusesa e Arshad, 2014) (Kokkalis, 2014). O aplicativo permite a extração dos sinônimos e sua utilização na TEKBS.
- 4. Avaliação da técnica proposta:** Foi realizado um exemplo de aplicação da técnica. O caso escolhido para a aplicação foi o Expert Committee System - EC (Deloach, Wood e Sparkman, 2001), um sistema que auxilia a organização de uma conferência. O objetivo foi verificar se a utilização da técnica facilitava, melhorava e contribuía com o processo de elicitação de MFs em comparação com os resultado já encontrado por (Oliveira, 2008) que fez o processo manual sem sinonímia e em (Cunha, 2014) que realizou o mesmo processo de Oliveira com auxílio de uma ferramenta. Também foram realizadas aplicações experimentais no laboratório de pesquisa do Departamento de Informática da Universidade Federal de Viçosa.

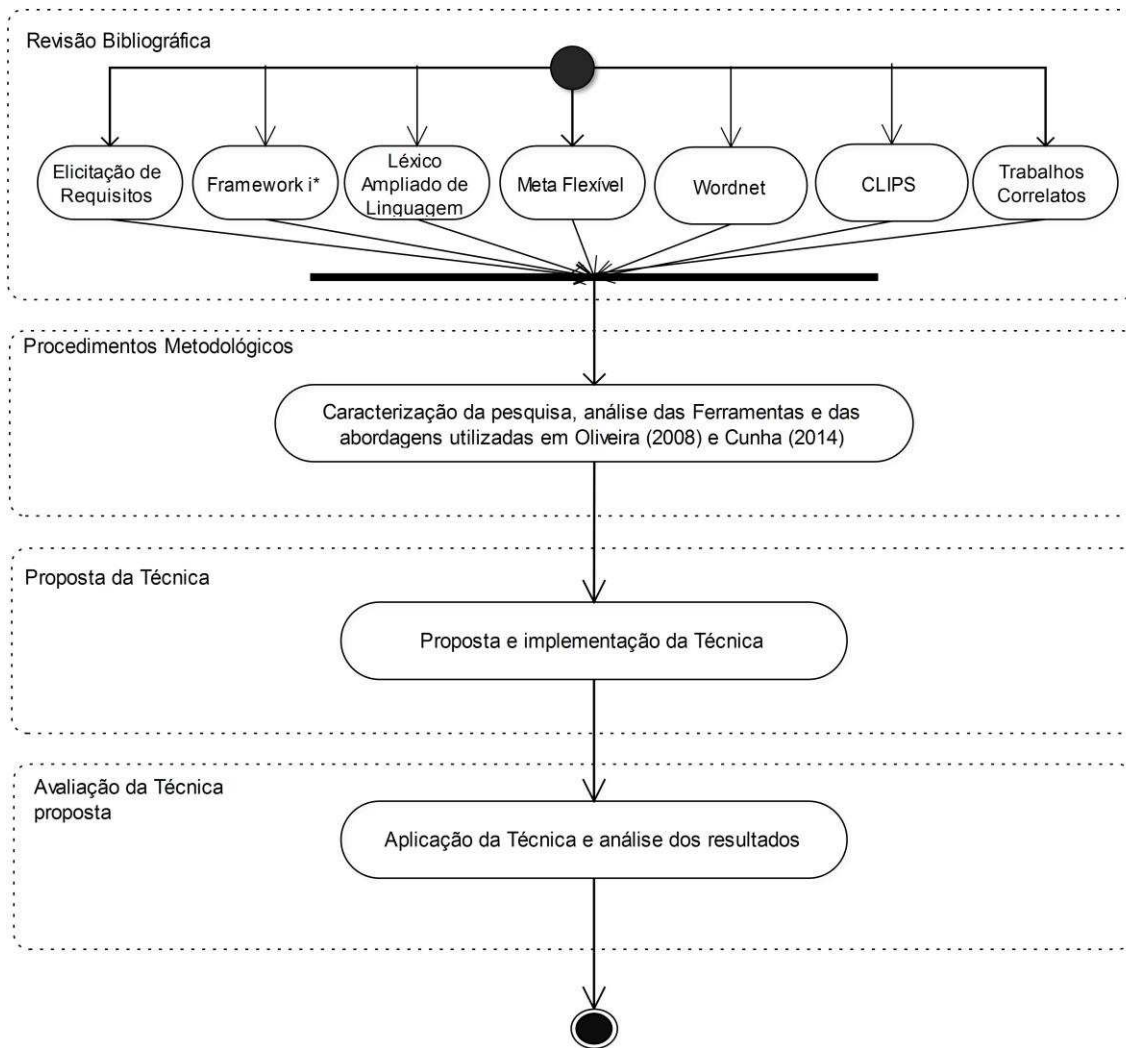


Figura 3.1: Fluxo das etapas de desenvolvimento da pesquisa.

Fonte: Adaptado de (Campos, 2009).

Capítulo 4

4 IMPLEMENTAÇÃO DA TEKBS PARA ELICITAÇÃO DE METAS FLEXÍVEIS

Neste Capítulo apresenta-se o resumo do Método ERi*c de (Oliveira, 2008) e do trabalho de (Cunha, 2014), juntamente com o processo de modelagem da TEKBS mostrando como ocorre o processo de elicitação de MFs com apoio de uma ferramenta de Inteligência Artificial, visando a atender ao objetivo geral que é “projetar e implementar uma técnica que auxilie a elicitação de Metas Flexíveis e a verificação das influências mútuas entre as Metas Flexíveis na elaboração do modelo iStar”.

4.1 Trabalhos considerados base da TEKBS

Este trabalho é uma extensão as Metas dos Agentes vindas do Léxico - AGFL pertencente a primeira etapa do Método ERi*c (Oliveira, Antônio Padua Albuquerque, Leite, J.C.S. e Cysneiros, Luiz Marcio, 2008) que foi semi-automatizado por (Cunha, 2014) criando a i*Get. No processo de semi-automatização de (Cunha, 2014) não foi inserida nenhuma modificação na essência do processo descrito em (Oliveira, Antônio Padua Albuquerque, Leite, J.C.S. e Cysneiros, Luiz Marcio, 2008).

A abordagem vista em (Oliveira, Antônio Padua Albuquerque, Leite, J.C.S. e Cysneiros, Luiz Marcio, 2008), Método ERi*c, se baseia em um processo bem delineado de construção, com uma técnica para elicitação de metas e atores envolvidos, além de prover uma estratégia de modelagem que possibilite a modularização dos modelos de Dependência Estratégica - SD e de Razão Estratégica - SR do Framework i*.

O trabalho de (Cunha, 2014) apresenta uma ferramenta. i*GET, que automatiza parte dos processos descritos em (Oliveira, Antônio Padua Albuquerque, Leite, J.C.S. e Cysneiros, Luiz Marcio, 2008), ver Figura 4.3. A ferramenta auxilia na

elicitação de metas, parametrizando a linguagem do domínio na busca por ocorrências de metas nos símbolos contidas no LAL. Na parametrização ocorre a transformação da linguagem em fatos usando o mecanismo de inferência da ferramenta CLIPS (Riley, 2013), formando a base de dados. A utilização do CLIPS facilita o processo de tomada de decisão do ER e auxilia a ferramenta a elicitar metas a serem usadas nos modelos construídos com a notação do Framework *i**. Com a ferramenta o autor (Cunha, 2014) espera facilitar e melhorar a extração de metas que comporão os modelos intencionais a partir da descrição dos símbolos disponíveis no LAL.

A base de fatos da *i*Get* (Cunha, 2014) contendo os atores, as metas, a descrição de comportamento do ator e o fator de certeza, que tem como função auxiliar na classificação das metas baseadas no histórico das classificações anteriores, é utilizada como entrada da TEKBS. Na subseção seguinte é feita uma descrição mais detalhada do Método *ERi*c* e da *i*Get*.

4.2 Método *ERi*c* – Engenharia de Requisitos Intencional.

O Método *ERi*c* foi desenvolvido por (Oliveira, 2008), ele contribui com a Engenharia de Requisitos de Sistemas Multi_Agentes e tem por base o conceito de intencionalidade utilizado pelo Framework *i**. Para (Monsalve e Leite, 2013) o Método *ERi*c* fornece algumas contribuições importantes, como o processo de elicitação de MCs e MFs, a técnica de AGFL para elicitar metas de atores, entre outras.

O *ERi*c* é composto por seis etapas, as quais cobrem as fases de elicitação, modelagem e a análise de processo de requisitos como pode ser visto na Figura 4.1.

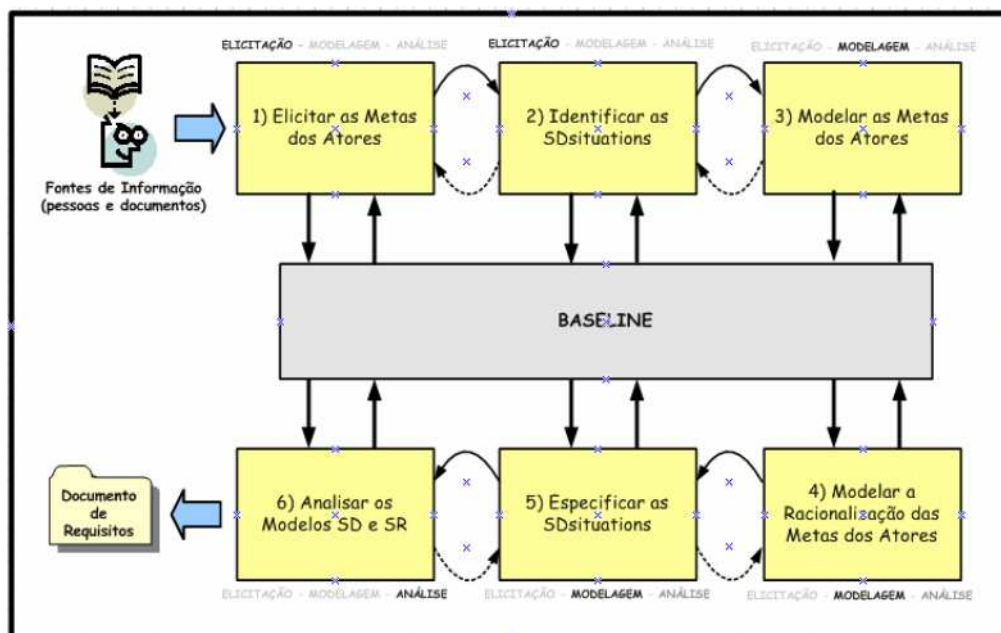


Figura 4.1: Visão geral do Método Eri*c.

Fonte: (Oliveira, 2008)

A primeira etapa do Método ERI*c é “Elicitar Metas dos Atores”, que incluem MCs e MFs é utilizada a técnica Metas dos Agentes Vindas do Léxico - AGFL que utiliza o LAL. Uma descrição mais detalhada da técnica AGFL pode ser encontrada na subseção 4.2.1.1.

A segunda etapa “Identificar as Strategic Dependency Situation – SDsituation, foi definida por Oliveira e Cysneiros (Oliveira, 2008) para auxiliar no processo de modelagem do Framework i* garantindo que a variável tempo e as ligações do tipo pontes, “bridges”, possam ser consideradas por meio da abordagem situacional.

Na terceira etapa “Modelar as Metas dos Atores” é possível representar as MCs e MFs no Painel de Intencionalidade (IP Diagram). Sua utilização pode beneficiar a representação em três pontos distintos: 1) representação exclusiva das MC e MF, e das relações entre elas por meio das dependências, correlações, contribuições e equivalências, utilizando um único diagrama considerando os “níveis de relevância” e o agrupamento das metas por SDsituation; 2) por não ser uma análise aprofundada, ele permite a avaliação do primeiro impacto do efeito da propagação do não-alcance de MC e o alcance parcial das MF; e 3) a facilidade da avaliação da complexidade do diagrama através do número ciclomático (McCabe,

1976) (Oliveira, Antônio Padua Albuquerque, Leite, J.C.S. e Cysneiros, Luiz Marcio, 2008) (Oliveira, 2008).

A quarta etapa “Modelar a Racionalização das Metas dos Atores” é composto pela modelagem da racionalização das MC e MF, criando “minimodelos” SR utilizando os Modelos SD com a finalidade de gerar dependências estratégicas entre os atores usando os mesmos critérios de (Yu, 1995).

Na quinta etapa “Especificar as SDsituation”, é feita a especificação das situações de dependência estratégica e por fim, na sexta e última etapa “Analisar o Modelo SD e SR”, é utilizando a técnica de análise denominada “i*Diagnoses”, ou diagnósticos do i*, para verificar modelos SD e SR (Oliveira, Antônio Padua Albuquerque, Leite, J.C.S. e Cysneiros, Luiz Marcio, 2008).

No ERi*c são encontrados indicativos da necessidade do método ser apoiado por ferramentas de software. A utilização das ferramentas tem como finalidade facilitar as atividades realizadas pelo engenheiro de requisito no processo e garantir, de modo automatizado, a rastreabilidade dos requisitos elicitados (Leite e Oliveira, 1995) (Oliveira, 2008).

Um exemplo da utilização do ERi*c pode ser visto em (Monsalve e Leite, 2013) que buscou mostrar a necessidade de considerar a interação entre os atores nas fases iniciais do modelo de jogos. Para isso, foi feita a modelagem do jogo SimULES no ERi*c, para que ao final do processo de elicitação, pudesse obter resultados com melhor entendimento. Outro exemplo pode ser visto em (Oliveira e Leite, 2014) que utiliza o Método ERi*c em uma situação real de uma grande organização usando o contexto de contratos de investimentos de Telecom. Os resultados da modelagem do contexto não só auxiliaram no entendimento do processo como também levaram a modificações trazendo melhorias para parte das atividades relacionadas aos contratos.

A ideia de Oliveira é que o Método ERi*c contribua para melhorar a qualidade final dos requisitos com modelos mais completos, que os requisitos possam ser apreendidos com mais facilidade que exista uma melhor organização na sua apresentação e, por fim, chegou a modelos menos sobrecarregados (Oliveira, Antônio Padua Albuquerque, Leite, J.C.S. e Cysneiros, Luiz Marcio, 2008).

Mais detalhes do processo completo do Método ERi*c, podem ser vistos em (Oliveira, 2008) (Oliveira, Antônio Padua Albuquerque, Leite, J.C.S. e Cysneiros, Luiz Marcio, 2008) e da automatização do AGFL em (Cunha, 2014).

4.2.1 Elicitar Metas de Atores

A elicitação das MFs ocorre na primeira etapa do Método ERi*c - Elicitar Metas de Atores - que é composta por 3 subatividades como mostra a Figura 4.2.

A primeira subatividade “Preparar LAL – Léxico Ampliado de Linguagem” está voltada para a identificação dos atores e pela identificação das fontes de informação que podem ser utilizadas no processo de elicitação. A segunda subatividade “Definir a AGFL – Metas dos Agentes vindas do Léxico” é responsável por elicitar os atores envolvidos no processo e pela elicitação das metas dos atores a partir dos símbolos, que inclui a elicitação das MF, por fim, a terceira subatividade “Refinar Metas” é responsável por excluir as duplicidades das metas e dos atores elicitados, agrupar as MC e MF por ator e organizá-las em ordem cronológica.

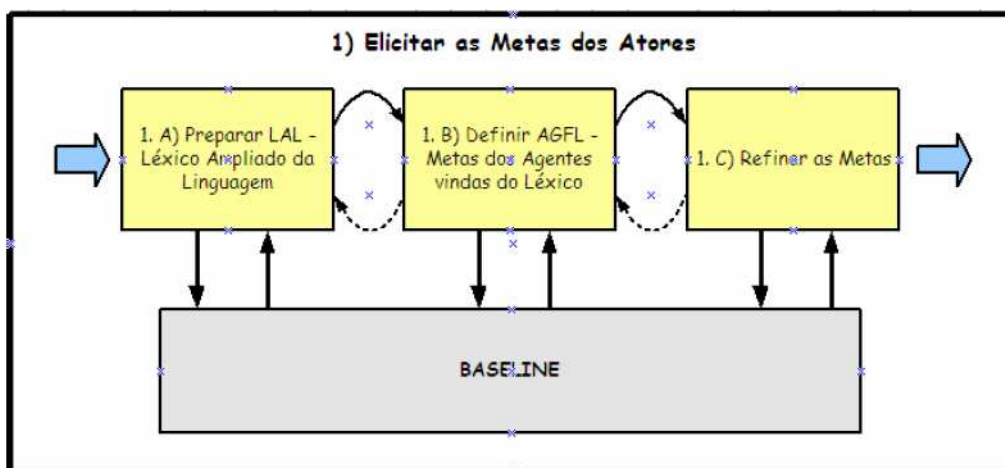


Figura 4.2: Detalhamento da 1ª etapa do Método ERi*c.
Fonte: Oliveira (2008)

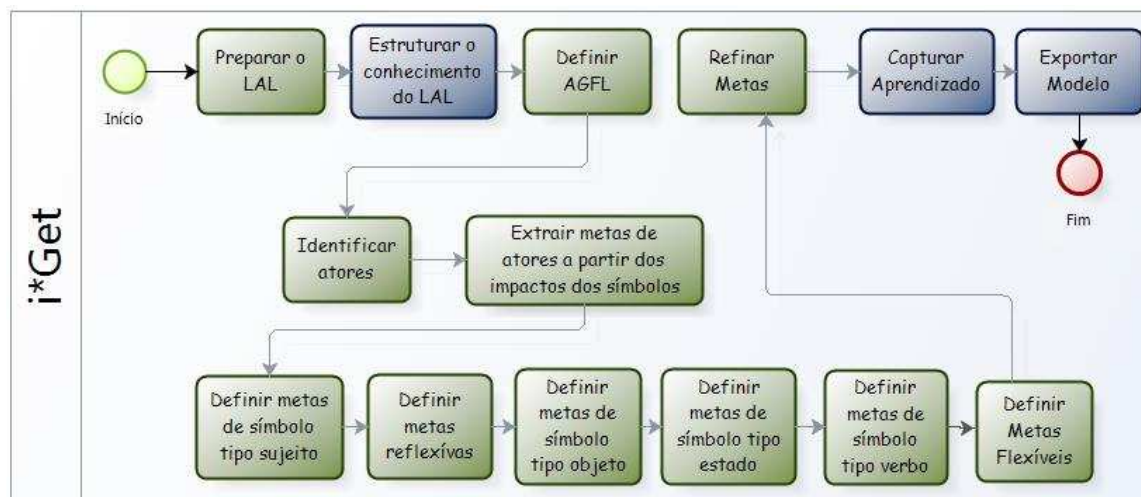
Para facilitar as atividades realizadas pelo engenheiro de requisitos e garantir a rastreabilidade, (Cunha, 2014) cria a i*Get como pode ser visto na Figura 4.3.

A i*Get é composta de 3 atividades que estão na cor azul e de outras 3, na cor verde, vindas do Método ERi*c e semi-automatizadas pelo autor.

Apesar da i*Get possuir 6 atividades semi-automatizadas, neste trabalho somente será descrito a atividade “Definir a AGFL” proveniente do Método ERi*c, que possui como ponto chave a identificação da motivação por trás de cada ação. Ademais, é nesta atividade que é descrito o processo de elicitação das MFs que é

utilizado como referência na implementação da TEKBS. A descrição das demais atividades que compõem a i*Get, pode ser vista em (Cunha, 2014).

Workflow da i*Get



Legenda:

Em verde - Método ERi*c (Oliveira, 2008).

Em Azul - Subatividade do ERi*c “Elicitar Metas de atores” – semi-automatizada

Figura 4.3: Detalhamento da 1ª etapa do Método ERi*c automatizada por Cunha (2014).

4.2.1.1 Definir AGFL – Metas dos Agentes vindas do Léxico

Na descrição do LAL os atores envolvidos no processo são símbolos do tipo sujeito. Eles também estão presentes nas noções e nos impactos que descrevem os demais símbolos do tipo objeto, verbo e estado (Oliveira, Antônio Padua Albuquerque, Leite, J.C.S. e Cysneiros, Luiz Marcio, 2008).

As ações que ocorrem em uma organização são mencionadas por impactos. Uma ação pode ser de dois tipos, concreta ou flexível. Se a ação for concreta ela modificará um estado, passando-o para outro estado. Diferentemente, a ação flexível não modifica um estado, mas adiciona um atributo de qualidade (Oliveira, 2008).

De acordo com (Oliveira, 2008) o termo ação concreta foi adaptado de Sá Carvalho (Carvalho, 1988) e pode ser definido como, uma ação executada por um usuário de Sistemas de Informação apoiada pelas informações desse sistema. A

ação flexível foi apresentada pela primeira vez por (Oliveira, 2008), e complementa a ação concreta, logo, sempre que existir uma ação ela vai ser classificada como concreta ou flexível. Para o Método ERI*c a ação flexível é qualificada com o mesmo significado que o utilizado por MF.

Uma ação flexível é pouco precisa, depende da interpretação dos envolvidos. Pelo fato de as ações modificarem estados, o ponto crucial é identificar o porquê - motivação - escondido em cada ação, seja ela concreta ou flexível. Para uma ação flexível existir é necessário que também existam atores e metas concretas, sendo assim são necessárias duas outras subatividades neste ponto do método: Identificar atores e Extrair as metas dos atores a partir dos impactos dos símbolos, ver Figura 4.3.

Para Identificar os atores é preciso classificá-los seguindo o terceiro passo dos seis descritos na subseção 2.4. Após a classificação, os símbolos que foram classificados como sujeito e os praticantes de ações sobre objetos, são os principais candidatos a serem classificados como atores.

Para Extrair as metas dos atores a partir dos impactos dos símbolos é preciso seguir algumas heurísticas que podem ser encontradas em (Oliveira, 2008) ou utilizar a ferramenta de (Cunha, 2014) que elicita as metas concretas de forma semi-automática por meio de regras, adaptadas de (Oliveira, 2008).

As MFs somente serão elicítadas após a elicitação de todas as MCs. Esse passo precisa ser considerado porque as MFs devem ser associadas as MCs. Para a elicitação da MF deve-se utilizar o template 3.C descrito por (Oliveira, 2008) e que pode ser visto na Figura 4.4. Na i*Get essa mesma estrutura está descrita em uma regra CLIPS que é disparada após a execução das regras que elicítam as metas concretas.

-- impacto	<meta flexível>		<meta concreta associada>	<ator>
	<TIPO> resposta ao por quê?	[TOPICO] sujeito/objeto LAL		
PESQUISADORES				
-- participam da conferência.	ação flexível			
Para que	reconhecimento	[comitê]	conferência seja executada por	coordenador geral

Figura 4.4: Ações flexíveis de símbolos tipo sujeito e tipo objeto.

Fonte: (Oliveira, 2008)

Após as MCs serem elicitadas as MFs provenientes de símbolos tipo sujeito, tipo objeto, tipo verbo e do tipo estado podem ser elicitadas. Como pode ser observado na Figura 4.4, a MF reconhecimento[comitê] está associada a MC “conferência seja executada”, ao ator “coordenador geral”.

4.3 Processo de Construção da TEKBS

A TEKBS foi criada para auxiliar o engenheiro de requisitos na elicitação de MF utilizando sinônimos. A utilização da sinonímia possibilita a extração de metas que não foram explicitamente declaradas e pode contribuir para que um número maior de MFs sejam elicitadas, além de poder minimizar as omissões permitindo a elaboração de um documento de requisitos mais completo e mais próximo da realidade do problema.

Seu processo de elaboração possui influência dos trabalhos de (Oliveira, Antônio Padua Albuquerque, Leite, J.C.S. e Cysneiros, Luiz Marcio, 2008) e de (Cunha, 2014) e é constituída pelas atividades apresentadas na raia (3) da Figura 4.5

A Figura 4.5 mostra na cor verde o workflow da etapa de elicitação do Método ERi*c (Oliveira, Antônio Padua Albuquerque, Leite, J.C.S. e Cysneiros, Luiz Marcio, 2008). Na cor azul - raia (2) - aparecem as atividades que foram desenvolvidas por (Cunha, 2014) e na cor rosa - raia (3) - as atividades projetadas nesta pesquisa.

A técnica TEKBS é composta por 6 atividades. Embora as atividades estejam dispostas em um formato linear este não é um processo no formato “Waterfall”. Podem existir momentos em que o engenheiro de requisitos irá retornar brevemente para uma etapa anterior, ou vai saltar para uma atividade posterior ou ainda iterar entre as etapas, de acordo com a necessidade detectada. Contudo, ao elaborar a Figura 4.5, não foi possível inserir todas as possibilidades de iteração uma vez que deixaria a figura ilegível dificultando o entendimento da técnica e de sua execução.

A primeira atividade da TEKBS “Elicitar Metas Flexíveis” é responsável por capturar as MFs do contexto, que são provenientes das “ações flexíveis” identificadas na base de dados recebida. A segunda atividade “Selecionar e Armazenar Metas Flexíveis” é constituída do “Module Extracting” que escolhe somente as MFs válidas e do “Module SoftgoalrootFile” que é responsável por armazenar a MFs extraídas de

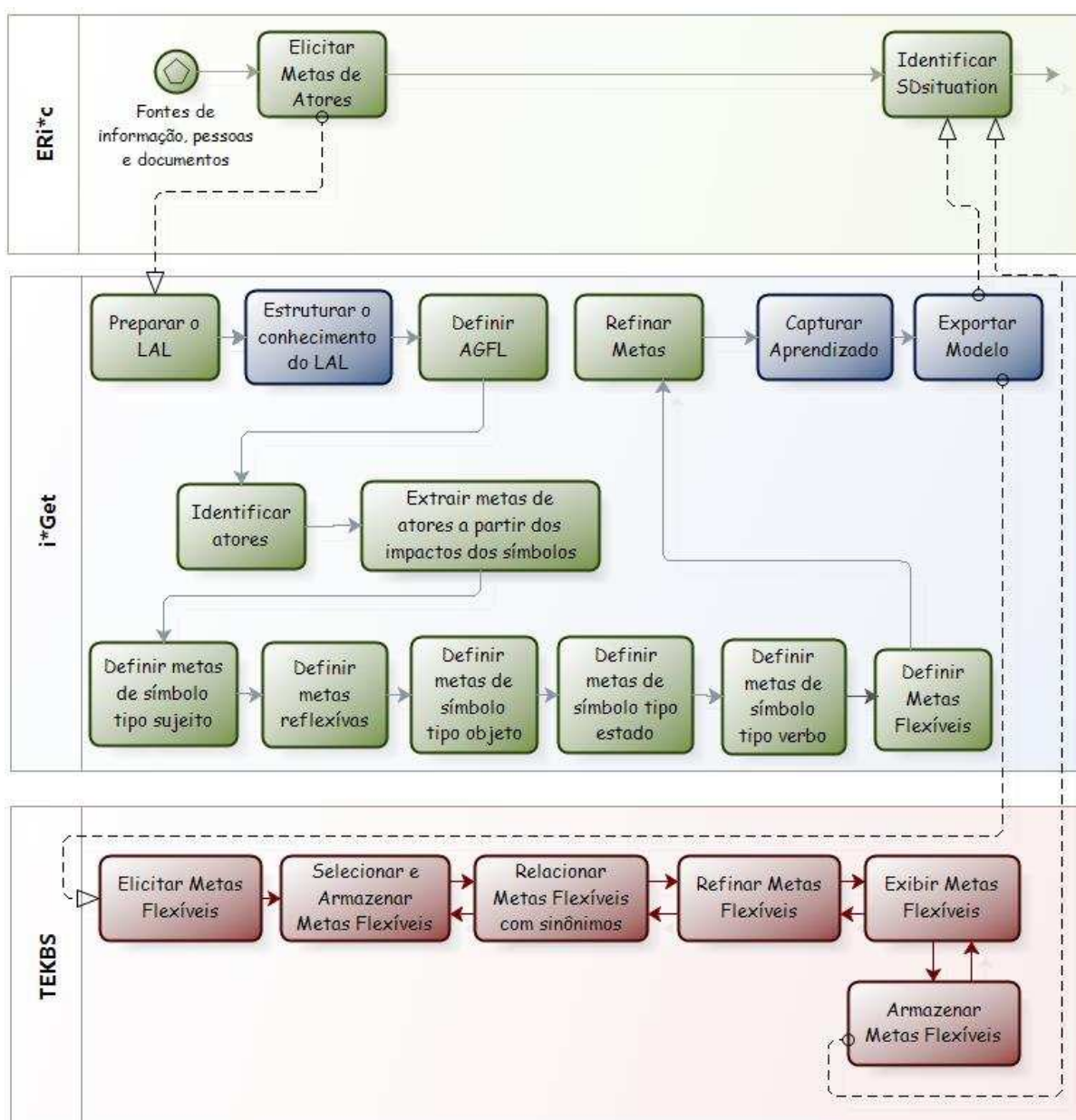
(Cunha, 2014). A terceira atividade “Relacionar Metas Flexíveis com sinônimos” é formada pelo “Module Synonymous” e faz o casamento entre as MFs e seus sinônimos que comporão a Base de Sinônimos – BS.

A quarta atividade “Refinar Metas Flexíveis” é composta de três módulos: “Module Inserting”, “Module Single Delete” e o “Module Duplicates Softgoal”. O “Module Inserting” permite o engenheiro de requisitos inserir novas MFs na base de fatos, o “Module Single Delete” tem a função de excluir MFs de forma individual da base, porém com a solicitação do engenheiro de requisitos e o “Module Duplicates Softgoal” exclui as MFs duplicadas.

A atividade de número cinco “Exibir Metas Flexíveis” é constituída do “Module Display Softgoals” e permite ao engenheiro de requisitos acompanhar o resultado de todas as atividades durante a execução. E, para concluir, a sexta atividade “Armazenar Metas Flexíveis é formada pelo “Module BaseFile” sendo responsável por criar arquivos externos contendo MFs raízes e seus sinônimos, além de gerar uma nova base de sinônimos”.

Nas subseções seguintes será detalhado o processo de construção de cada atividade da técnica bem como o processo de validação.

Workflow do Método ERI*c com i*Get e com a Técnica de Elicitação de Metas Flexíveis Baseada em Conhecimento – TEKBS



Legenda:

Em verde - Método ERI*c (Oliveira, 2008).

Em azul - Knowledge-based Goal Eliciting Tool – i*GET

Em rosa - Knowledge-based Softgoal Elicitation Technique - TEKBS.

4.4 Elicitar Metas Flexíveis com apoio de uma ferramenta de Inteligência Artificial

A linguagem utilizada para a implementação da TEKBS foi o CLIPS que possui duas memórias de trabalho: uma base de fatos e uma base de regras. Para elaborar a base de fatos é necessário estruturar as informações do domínio que estão em linguagem natural para o LAL. A Figura 4.6 exibe o símbolo “conflito” do domínio EC estruturado no LAL utilizando a ferramenta C&L.

Informações sobre o símbolo

Nome:	conflito
Noção:	Estado de indeterminação sobre a aceitação de um artigo .
Classificação:	estado
Impacto(s):	- artigo precisa receber uma outra avaliação.
Sinônimo(s):	conflitos.

Figura 4.6: Exemplo de um símbolo do LAL tipo estado.

Fonte: C&L

A Figura 4.6 apresenta uma entrada típica no LAL, uma ficha que cataloga a descrição do símbolo “conflito” e seu uso pretendido no contexto do problema. Um conflito existe quando um artigo que foi submetido a uma conferência é enviado para três revisores e as revisões geram indecisões quanto à aceitação do artigo. Neste caso, para resolver o conflito o artigo deve ser revisado por um membro do comitê.

No C&L as entradas são autoexplicativas, e após preenchidas, a análise do termo continua com a transcrição da Figura 4.6 para o template 3.C apresentado na Figura 4.7, que é utilizado por (Oliveira, Antônio Padua Albuquerque, Leite, J.C.S. e Cysneiros, Luiz Marcio, 2008) para elicitar MFs. Nesta transcrição, a pergunta a ser sempre respondida pelo engenheiro de requisitos é “porque?”, e o objetivo é completar a semântica de cada termo dentro do domínio. Esta análise é descrita e desenvolvida no trabalho de (Cunha, 2014) e os resultados foram utilizados neste trabalho.

TIPO: ESTADO	<meta flexível>		<meta concreta associada>	<ator>
-- impacto resposta ao por quê?	<TIPO> atributo de qualidade	[TOPICO]> sujeito/objeto LAL		
CONFLITO				
-- artigo precisa receber uma outra avaliação.	ação flexível			
Para que	qualidade	[revisão]	conflito seja resolvido	por comitê

Figura 4.7: Transcrição do símbolo conflito da Fig. 2 para o Template 3.C.

Fonte: (Oliveira, 2008)

4.4.1 Elicitar Metas Flexíveis

Para a elicitação de MFs é necessário declarar e preparar o ambiente CLIPS, para que sua máquina de inferência seja capaz de extrair da base de dados as MFs e as demais informações relacionadas, como as MFs originadas dos sinônimos.

A preparação do ambiente inicia-se com a declaração das variáveis que serão utilizadas ao longo da implementação. A Figura 4.8 exibe a estrutura de parametrização de uma MF – “softgoalElement”, composta da “sgId” que identifica as MFs. Por não existir de forma isolada foram criados mecanismos que mantivessem o vínculo da MF com o domínio, por isso, defini a variável “sgGoalId” que permite identificar a qual MC a MF está relacionada e a variável “sgActorName” que é responsável por mostrar a qual ator pertence a MF.

Por ser composta de um tipo, que é um atributo de qualidade, e de um tópico que pode ser o sistema ou parte dele, foram declaradas variáveis para receber tais valores e que pudessem ser utilizados ao longo da aplicação. Assim, foram declaradas as variáveis “sgQualityAttribute” e “sgSubjectObject” que são responsáveis por receber o tipo e o tópico respectivamente. São essas variáveis que permitem ao longo das atividades da técnica ter acesso ao que chamamos MFs raiz, de onde são originadas as MFs sinônimas.

```

48 (deftemplate softgoalElement "Softgoal element"
49   (slot sgId (type SYMBOL) )
50   (slot sgQualityAttribute (type STRING) )
51   (slot sgSubjectObject (type STRING) )
52   (slot sgGoalId (type SYMBOL) )
53   (slot sgActorName (type STRING) )
54   (slot sgBhId (type SYMBOL) )
55 )

```

Figura 4.8: Estrutura de parametrização da MF.

Após a declaração das variáveis buscou-se criar um mecanismo de módulos para facilitar o acesso às atividades da TEKBS. Esses módulos podem funcionar de forma sequencial ou independente, exceto para o primeiro “Module Extracting” que precisa ser executado todas as vezes que a TEKBS for utilizada, pois é ele o responsável pela extração das MFs da base de dados da i*Get que alimenta a técnica. Dessa forma criou-se um módulo baseado em regras chamado de “main-menu”, ver Figura 4.9. É este módulo um dos responsáveis pela flexibilidade e pelo acesso aos 9 módulos que compõe a TEKBS.

```

105 (defrule main-menu
106   ?Mm <- (menu-level module main)
107   =>
108   (retract ?Mm)
109   (printout t crlf crlf)
110   (printout t
111     "Choose the Module" crlf
112     "by typing a number and pressing the return key" crlf crlf
113     "          1) Module Extracting.          " crlf
114     "          2) Module SoftgoalrootFile      " crlf
115     "          3) Module Synonymous.          " crlf
116     "          4) Module Inserting             " crlf
117     "          5) Module Display Softgoals.    " crlf
118     "          6) Module Single Deleting       " crlf
119     "          7) Module Duplicate Softgoal.   " crlf
120     "          8) Module BaseFile              " crlf
121     "          9) Module Exit.                 " crlf
122     "Your Choice: ")
123
124   (bind ?answer (read))
125   (assert (module choice ?answer))
126   (printout t crlf crlf crlf)
127 )

```

Figura 4.9: Estrutura do módulo principal

O módulo main-menu sozinho não possibilita que o engenheiro de requisitos saia do menu, acesse um módulo e volte ao menu. Para que essa circularidade fosse possível foram implementadas duas outras estruturas como pode ser visto nas Figuras 4.10 e Figura 4.12. A Figura 4.10 mostra uma estrutura de direcionamento do main-menu para o módulo, neste caso o módulo “Module Extracting”, os demais módulos são acionados de forma similar ao “Module Extracting” exceto o “Module Exit”, representando pela Figura 4.11, que ao invés de direcionar para alguma atividade ele finaliza a TEKBS por meio do comando “halt”.

```

130 (defrule ruleModuleChoiceExtracting
131   (module choice 1)
132   =>
133   (printout t "*** You chose Module Extracting ***" crlf)
134   (focus Extracting)
135 )

```

Figura 4.10: Estrutura responsável por acionar o “Module Extracting”

```

188 (defrule ruleModuleChoiceQuit
189   (module choice 9)
190   =>
191   (printout t "*** You chose Module Exit" crlf)
192   (halt)
193 )

```

Figura 4.11: Estrutura responsável por finalizar a TEKBS

A Figura 4.12 faz o processo contrário ao efetuado pela Figura 4.10, aqui o direcionamento vai do módulo para o main-menu. Estas duas regras juntas, representadas nas Figuras 4.10 e 4.12, possibilitam que o engenheiro de requisitos possa caminhar de forma não linear entre os módulos.

```

261 (defrule Extracting::ruleChanceModule " Change focus to Main module"
262   (declare (salience 0))
263   ?fact <-(module choice 1)
264   =>
265   (assert (menu-level module main))
266   (retract ?fact)
267 )

```

Figura 4.12: Regra responsável por retornar ao menu principal.

4.4.2 Selecionar e Armazenar Metas Flexíveis

O processo de extração das MF pertence ao “Module Extracting” e o de armazenamento ao “Module SoftgoalrootFile”. O “Module Extracting” é primordial na estrutura da TEKBS, sendo o único que obrigatoriamente necessita ser executado todas as vezes que iniciar do zero o processo de elicitação, pois é por meio dessa atividade que a ferramenta é alimentada.

Para a extração de MFs da base de dados elaborou-se a rotina “Assertsoftgoalroot” que é responsável por extrair e inserir na base “basesoftgoal” o tipo e o tópico das MFs raízes. A inserção das MFs na “basesoftgoal” auxilia o acesso durante o processo de execução e permite gerar uma memória de trabalho, ver Figura 4.13.

```

189 (defrule Extracting::ruleAssertsoftgoalroot
190 "This rule inserts softgoals in the fact basis."
191
192     (declare (salience 95))
193     (softgoalElement
194       (sgId                ?idSg                )
195       (sgQualityAttribute  ?QualityAttributeSg  )
196       (sgSubjectObject    ?SubjectObjectSg     )
197       (sgGoalId           ?GoalIdSg            )
198       (sgActorName        ?ActorNameSg        ))
199     =>
200     (assert (basesoftgoal (sgId(gensym))
201       (sgQualityAttribute ?QualityAttributeSg)
202       (sgSubjectObject ?SubjectObjectSg)))
203 )

```

Figura 4.13: Regra responsável por guardar as Metas Flexíveis na basesoftgoal.

Para o armazenamento das MFs é preciso acionar o “Module SoftgoalrootFile” que possui a rotina “ruleopensoftgoalbasefile” que cria um arquivo externo chamado de “softgoalFile.clp” contendo todas as MFs raízes. O “softgoalFile.clp” é carregado no aplicativo JAVAS que tem por função localizar no dicionário on-line do Wordnet, usando a API disponibilizada pelo Wordnet, os sinônimos de todos os tipos e todos os tópicos encontrados no arquivo.

Para facilitar o acesso do engenheiro de requisitos as atividades, a memória de trabalho e permitir o acompanhamento de todos os processos foi criado o módulo de exibição, “Module Display Softgoals” e o módulo para arquivar “Module BaseFile”.

O “Module Display Softgoals” pertence à atividade “Exibir Meta Flexível” sendo composto de regras que permitem apresentar as MFs raízes e de seus sinônimos. Um exemplo das regras pode ser visto na Figura 4.14 que mostra a regra “Displaysoftgoalroot”, responsável por exibir as MFs raízes que foram armazenadas na “basesoftgoal”.

```

421 (defrule DisplaySoftgoal::ruleDisplaysoftgoalroot
422 "This rule Display softgoal found in the fact basis."
423
424     (declare (salience 60))
425     (basesoftgoal
426       (sgId                ?idSg                )
427       (sgQualityAttribute  ?QualityAttributeSg   )
428       (sgSubjectObject    ?SubjectObjectSg     )
429     )
430     =>
431     (printout t "Id:(" ?idSg ")
432               Softgoal: "?QualityAttributeSg"
433               [" ?SubjectObjectSg "]" crlf)
434
435 )

```

Figura 4.14: Exibir softgoals raízes

Uma regra similar imprime os sinônimos das MFs raízes. Essa impressão permite a visualização de todas as MFs durante o processo de elicitação sendo possível uma análise do engenheiro de requisitos das MFs elicítadas.

O “Module BaseFile” pertence a atividade “Armazenar Metas Flexíveis”. Sua função principal é arquivar informações para serem acessadas após a finalização da aplicação. Assim, o módulo gera dois novos arquivos, o "BasesoftgoalFile.clp" (Figura 4.15) que recebe todas as MFs seja raiz ou sinônima e "NewSynonymousFile.clp" (Figura 4.16) que recebe todos os sinônimos sendo eles organizados em um formato de base de entrada, o que permite que esse arquivo seja carregado na TEKBS.

```

535 (defrule BaseFile::ruleSelectsoftgoal
536 "This rule saves softagoal found in the
537     fact basis in BasesoftgoalFile.clp."
538
539     (declare (salience 40))
540     (softgoalElement
541       (sgId                ?idSg                )
542       (sgQualityAttribute  ?QualityAttributeSg   )
543       (sgSubjectObject    ?SubjectObjectSg     )
544       (sgGoalId           ?GoalIdSg            )
545       (sgActorName        ?ActorNameSg        ))
546     =>
547     (printout baseFile2 "(" ?idSg ")"
548               ?QualityAttributeSg " " "[" ?SubjectObjectSg "]" crlf)
549 )

```

Figura 4.15: Salva softgoals encontradas na base de fatos em BasesoftgoalFile.clp.

```

582 (defrule BaseFile::ruleUpdateNewSynonymousFile
583     "This rule is responsible for creating the file
584     containing deffacts"
585     (declare (salience 20))
586     (synSoftgoal (ttId ?synId)
587     (syntype ?syntype) (syntopic ?syntopic))
588     =>
589     (printout NewBaseFile "(synSoftgoal (ttId "?synId" )
590     (syntype \""?syntype\"") (syntopic \""?syntopic\""))" crlf)
591 )

```

Figura 4.16: Salva sinônimos encontradas na base de fatos em "NewSynonymousFile.clp"

Neste ponto da descrição já é possível notar que a TEKBS não é linear. Em sua segunda atividade foi possível acionar as atividades “Exibir Metas Flexíveis” e “Armazenar Metas Flexíveis”, a sequência de execução das atividades segue a necessidade de informações do engenheiro de requisitos.

Vale ressaltar que após a criação da "NewSynonymousFile.clp", de determinado domínio contendo todos os sinônimos, é possível executar a TEKBS somente utilizando seus sinônimos. Para isso é necessário que ao iniciar a aplicação o arquivo "NewSynonymousFile.clp" seja carregado. Neste caso deixa de ser obrigatória a execução do “Module Extracting” e passa a ser obrigatória a execução do “Module Synonymous”, pois passa-se a extrair sinônimos e não MFs raízes.

4.4.3 Relacionar Metas Flexíveis com Sinônimos

Após a finalização do processo de elicitación de MFs raízes, existe a necessidade do engenheiro de requisitos avaliar se o que foi levantando era exatamente o que o cliente queria. Sabe-se que esse processo é difícil e custoso por causa da subjetividade das MFs e por isso precisa-se contar com a experiência e conhecimento do engenheiro de requisitos.

É possível que exista nas descrições das necessidades elaboradas pelos clientes MFs que não foram elicítadas devido elas estarem implícitas no domínio. Para auxiliar o engenheiro de requisitos foi criada a base de sinônimos – BS, que chamamos de “synSoftgoal” e que tem por função armazenar todas as MFs sinônimas das MF raiz, ver Figura 4.17.

```

282 (defrule Synonymous::ruleAssertSynsoftgoal
283 "This rule inserts softgoals synonyms."
284
285     (declare (salience 75)      )
286
287     (synonymoustype
288     (type          ?type      )
289     (syntype      ?syntype   ))
290
291     (synonymoustopic
292     (topic        ?topic      )
293     (syntopic    ?syntopic))
294     =>
295     (assert (synSoftgoal (ttId(gensym))
296     (syntype ?syntype) (syntopic ?syntopic) ))
297
298 )

```

Figura 4.17: Base de Sinônimos.

Até este momento da execução, tinha-se uma base para armazenar os sinônimos e um arquivo contendo todas as MFs raízes que seriam utilizadas para buscar os novos sinônimos. Porém era inviável buscar o sinônimo de cada MF raiz de forma isolada. Por isso decidiu-se pela elaboração de um aplicativo que permitisse o acesso direto ao dicionário de sinônimos de língua inglesa Wordnet e a elicitação dos dados de forma que após a elicitação os sinônimos pudessem ser inseridos na TEKBS de forma automática.

A criação do aplicativo foi necessária levando em consideração os poucos recursos de implementação oferecidos pelo ambiente CLIPS. Apesar dessa limitação, citada acima, é possível integrar o CLIPS a outras linguagens como o JAVA e por isso, desenvolveu-se um aplicativo na linguagem JAVA, chamado de JAVAS (Figura 4.18), que recebe um arquivo com a extensão “.clp”, neste caso o "softgoalFile.clp" contendo as MFs raízes e que retorna um outro arquivo na mesma extensão contendo somente seus sinônimos.

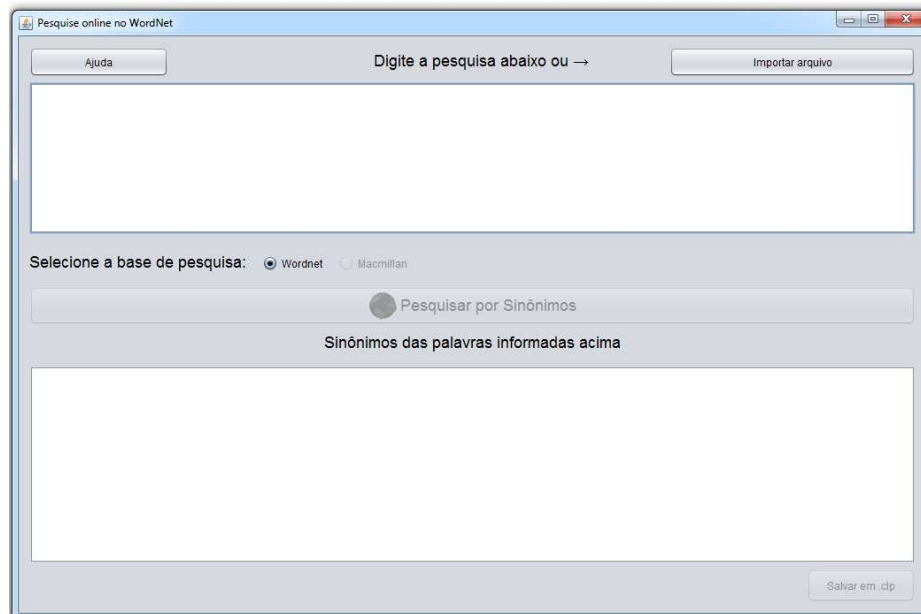


Figura 4.18: Tela do programa JAVAS que acessa o Wordnet e retorna sinônimos das Metas Flexíveis

Dentro do processo de execução da TEKBS pode ser necessário realizar consultas individuais de sinônimos de alguma MF raiz sem precisar criar arquivos no formato CLIPS. Para realizá-las é preciso inserir no aplicativo o tipo e o tópico nesta ordem cada um em uma linha e solicitar que a pesquisa seja feita (Figura 4.19).

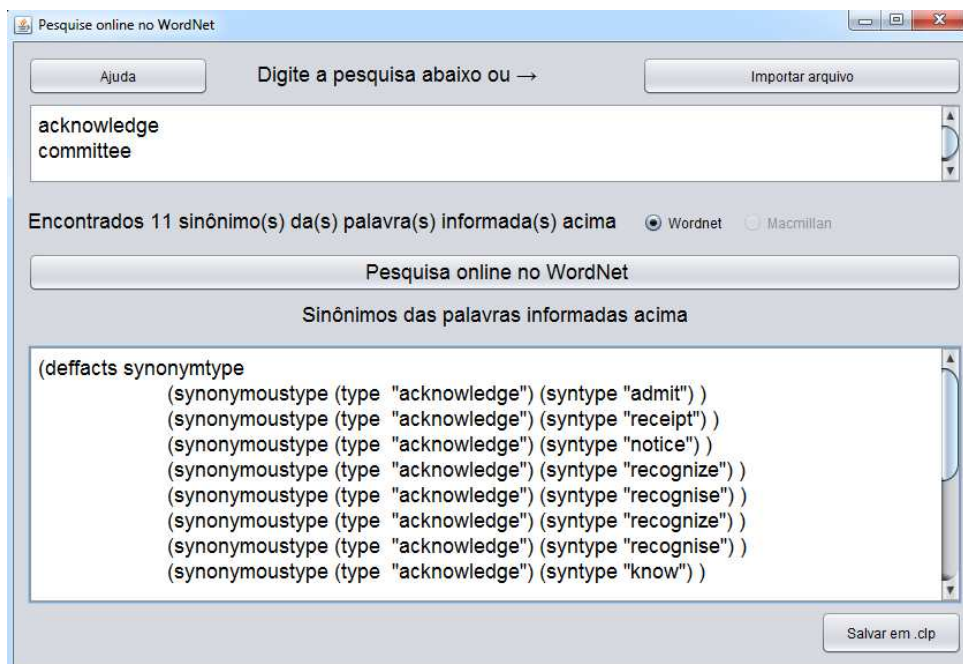


Figura 4.19: Consulta individual de sinônimo de Meta Flexível

Por ser utilizado para localizar sinônimos de diferentes domínios o arquivo de retorno sempre será nomeado pelo engenheiro de requisitos que também deve escolher o local onde o arquivo vai ser salvo. Ao clicar em “salvar em .clp” abri uma nova janela que permite a escolha do diretório no computador onde o arquivo pode ser salvo.

4.4.4 Refinar Metas Flexíveis

Após o processo de eliciação de MFs raízes e de busca dos sinônimos é possível encontrar três necessidades: a) excluir MFs ou sinônimos duplicados; b) excluir MFs que não fazem parte do escopo do problema; e c) inserir MFs que fazem parte do escopo e que não estavam presentes no processo inicialmente. A inserção e a exclusão podem ser justificadas pelos processos de mudanças comuns em projetos de software.

Para atender a essa demanda foram criados três módulos, sendo eles: o “Module Duplicate Softgoal”, o “Module Single Deleting” e o “Module Inserting”. Apesar de todos pertencentes à atividade de “Refinar Metas Flexíveis” da TEKBS os módulos são independentes, sendo possível executar cada um de forma isolada.

O “Module Duplicate Softgoal” é o responsável por excluir metas duplicadas, ele provê um mecanismo capaz de verificar a existência de duplicidades, considerando a existência de duas ou mais MFs com Tipo e tópico idênticos.

Quando o “Module Duplicate Softgoal” é acionado não é considerado se os sinônimos encontrados são relevantes e nem se eles são de fato sinônimos da MFs raiz. Essas atividades são executadas na interação do módulo com o engenheiro de requisito utilizando a função “DeleteOneSynSoftgoal” do módulo “Module Single Deleting”.

Para eliminar as duplicidades criou-se a regra “ruleDeleteSynSoftgoal” que exclui duplicidade das MFs sinônimas, ver Figura 4.19.

```

495 (defrule DuplicateSoftgoal::ruleDeleteSynSoftgoal
496 "This rule deletes duplicate synsoftgoal found in the facts base."
497
498 (declare (salience 50))
499 ?fact3 <- (synSoftgoal
500 (ttId ?ttId1)
501 (synTopic ?syntopic1)
502 (synType ?syntype1)
503
504 )
505
506 ?fact4 <- (synSoftgoal
507 (ttId ?ttId2)
508 (synTopic ?syntopic2)
509 (synType ?syntype2)
510
511 )
512
513 )
514 (test (and (neq ?ttId1 ?ttId2)
515 (eq ?syntype2 ?syntype1 )
516 (eq ?syntopic2 ?syntopic1)))
517 =>
518 (retract ?fact4)
519 )

```

Figura 4.20: Regra responsável por excluir duplicidade.

A regra desconsidera a igualdade das identificações das MFs sinônimas representadas pela variável “ttId”, além das outras identificações e demais relações existentes na base de dados. Aqui somente é considerado o sinônimo do tipo e o sinônimo do tópico da MF raiz e se existe mais de um sinônimo igual, caso exista, ocorrerá à exclusão mantendo somente um sinônimo na memória de trabalho.

O “Module Single Deleting” é responsável por excluir MF que não fazem parte do escopo do problema. Ele fornece um sistema de comparação entre as informações inseridas pelo engenheiro de requisitos com as que já existem na base “synSoftgoal”.

Para possibilitar a exclusão foi criada uma função chamada de “DeleteOneSynSoftgoal” como pode ser visto na Figura 4.20.

```

405 (defunction SingleDeleting::DeleteOneSynSoftgoal ( )
406 (printout t "Enter below the two softgoals field:" crlf crlf
407 "the synonyms of the <[TYPE]QUALITY ATTRIBUTE>
408 and the <[TOPIC]SUBJECT/OBJECT LAL>." crlf crlf)
409
410 (printout t "Enter with QUALITY ATTRIBUTE: ")
411 (bind ?dsyntype (readline))
412 (printout t "Enter with SUBJECT/OBJECT LAL: ")
413 (bind ?dsyntopic (readline))
414 (if (do-for-all-facts ((?f synSoftgoal)) (and (eq ?f:syntype ?dsyntype)
415 (eq ?f:syntopic ?dsyntopic))
416 (retract ?f)
417 TRUE)
418 then
419 (printout t "The SynSoftgoals were successfully deleted" crlf)
420 else
421 (printout t "Found no SynSoftgoal, enter a new SynSoftgoal" crlf))
422 )

```

Figura 4.21: Função responsável por deletar uma única Meta Flexível.

A função “DeleteOneSynSoftgoal” recebe e armazena a MF a ser deletada, além de comparar as MF recebidas com as MF existentes na memória de trabalho. Se a MF for encontrada ela é deletada e é exibido “The SynSoftgoals were successfully deleted”, caso contrário, a mensagem “Found no SynSoftgoal, enter a new SynSoftgoal” é exibida para o engenheiro e um novo processo pode ser iniciado.

Para concluir, o “Module Inserting” permite inserir MF que fazem parte do escopo e que não estavam presentes no processo inicialmente. Assim como a “DeleteOneSynSoftgoal” ele permite o engenheiro de requisitos inserir os dados de tipo e de tópico, os armazena em variáveis e insere na base “synSoftgoal”.

A inserção é possível porque foi criada a função “createElement” como pode ser observado na Figura 4.21.

```

323 (deffunction Inserting::createElement ( )
324   (printout t "Enter below the two softgoals field:" crlf crlf
325             "the synonyms of the <[TYPE]QUALITY ATTRIBUTE>
326             and the <[TOPIC]SUBJECT/OBJECT LAL>." crlf crlf)
327
328   (printout t "Enter with QUALITY ATTRIBUTE: ")
329   (bind ?syntype (readline))
330   (printout t "Enter with SUBJECT/OBJECT LAL: ")
331   (bind ?syntopic (readline))
332   (assert (synSoftgoal (ttId(gensym)) (syntype ?syntype) (syntopic ?syntopic)))
333   (printout t crlf crlf "Synonym of softgoal successfully inserted " crlf)
334 )

```

Figura 4.22: Função responsável por inserir sinônimos de Metas Flexíveis.

Com a finalização do processo de inserção, a MF sinônima poderá ser utilizada em todas as atividades da TEKBS, inclusive as já executadas anteriormente, sendo necessário somente acionar o módulo desejado.

Após a implementação das atividades foi utilizado um exemplo de aplicação do domínio Expert Committee Systems que está descrito no Capítulo 5 com a finalidade de validar a TEKBS.

Capítulo 5

5 EXEMPLO DE APLICAÇÃO DA TÉCNICA TEKBS

Para demonstrar o funcionamento da TEKBS foi feita a aplicação da técnica utilizando a descrição do Expert Committee System – EC (Deloach, Wood e Sparkman, 2001). Junto à aplicação foram avaliados além do processo de execução, sua flexibilidade e o acoplamento das MFs sinônimas com as MFs raízes elicitadas. Para verificar qual seria o comportamento da técnica se utilizasse outros dicionários de língua inglesa, foi feita uma comparação simplificada de todo o processo da TEKBS que utiliza o dicionário Wordnet caso ela utilizasse o Macmillan com thesaurus ou Thesaurus, e concluindo, foi realizada uma comparação do processo automatizado com uso de sinônimos com o processo manual realizado por (Oliveira, 2008) sem a utilização de sinônimos e a descrição da validação da técnica.

5.1 Processo de validação da TEBKS

Por ser considerado um estudo não experimental, este trabalho não tinha como objetivo a experimentação no mundo real da TEKBS. Contudo, foram realizadas aplicações experimentais com bases de dados reduzida do EC no laboratório de pesquisa do Departamento de Informática da Universidade Federal de Viçosa, com engenheiros de requisitos que permitiram visualizar modificações no contexto da elicitação, além do exemplo de aplicação (ver subseção 5.3), descritos neste capítulo, junto à descrição completa do EC, que é o problema escolhido para teste. A escolha desta descrição está relacionada à existência de diversos estudos (Oliveira e Cysneiros, 2006) (Coelho, Cirilo, Kulesza, von Staa, Rashid e Lucena, 2007) (Lobato, Garcia, Romanovsky e Lucena, 2008) (Oliveira, Antônio Padua Albuquerque, Leite, J.C.S. e Cysneiros, Luiz Marcio, 2008) (Nunes, Kulesza, Sant'Anna, Nunes, Garcia e de Lucena, 2009) (Trondal e Peters, 2013) que exploraram seu contexto, incluindo a elicitação de MFs raízes de forma manual realizada por (Oliveira, Antônio Padua Albuquerque, Leite, J.C.S. e Cysneiros, Luiz

Marcio, 2008) (Oliveira, 2008), que serviu para esta pesquisa como parâmetro de comparação da efetividade da técnica TEKBS (ver subseção 5.4.3).

Outra comparação realizada envolveu a modificação do processo da TEKBS que utiliza o dicionário de língua inglesa Wordnet. Foram feitas duas trocas: a) realização do processo da TEKBS substituindo o Wordnet pelo dicionário de língua inglesa Macmillan com thesaurus; b) uma segunda troca de dicionário foi feita dessa vez utilizando o Theaurus (ver subseção 5.4.4).

Vale destacar que a TEKBS é uma extensão do trabalho de (Oliveira, 2008), Método ERi*c, que passou por avaliações de sua estratégia. Apresentamos na ordem cronológica da execução 3 (três) ocorrências que consideramos significativas, sendo que as duas últimas ficaram fora do ambiente controlado de pesquisa:

- a. A primeira está incluída na tese do Método ERi*c (Oliveira, 2008) a qual relata a experimentação do método: “Para avaliar e validar o método Engenharia de Requisitos Intencional - ERi*c (Oliveira, A. P. A., Leite, J. C. S. e Cysneiros, L.M., 2008) foi aplicado o “Processo de Experimentação em Engenharia de Software” obtido em (Mafra e Travassos, 2006), que foi definido por (Wohlin, Runeson, Host, Regnell e Wesslén, 2000) e estendido por (Amaral, 2003).” A avaliação foi montada em duas etapas. Na primeira etapa participaram da experimentação 17 alunos e na segunda etapa participaram 9 alunos da graduação da Universidade do Estado do Rio de Janeiro - UERJ e da graduação da PUC-Rio da área de Engenharia de Software.
- b. A segunda avaliação foi executada na participação do evento “Comparing Modeling Approach Workshop – 2012 quando foi desenvolvido pelo método ERi*c o trabalho: CAR CRASH MANAGEMENT SYSTEM - Selected Approach: Intentional Requirements Engineering”.
- c. A terceira avaliação está descrita no artigo “The Experience of Using ERi*c in a Telecom Corporation” aceito para publicação na Seventh International i* Workshop (iStar14). O trabalho apresenta o uso do método ERi*c no contexto de contatos de investimento de uma grande empresa de telecomunicações. Essa experimentação em um problema real serviu como base para a dissertação de fim de curso de um aluno da UERJ (Sperandei, Oliveira, Leite e Werneck, 2010).

Sendo assim, além das aplicações experimentais e do exemplo de aplicação da TEKBS e das comparações com (Oliveira, 2008) e com os dicionários Macmillan com theaurus e Thesaurus, temos a base da técnica, Método ERi*c, experimentada.

5.2 Expert Committee System

O “Expert Committee System - EC” (Deloach, Wood e Sparkman, 2001) é um sistema multi-agente aberto, utilizado para apoiar o gerenciamento de submissões e ao processo de revisão realizado em conferências ou workshops (Garcia, 2004). Por ser um sistema baseado em agentes de software, o EC pode auxiliar os pesquisadores nas atividades de envio de trabalhos e processo de revisão que são as que mais consomem tempo.

Agentes são utilizados no EC como assessores de software que assumem diferentes papéis, os principais são: autor do trabalho, revisor, membro do comitê do programa e chair. Diversas são as atividades, receber artigo, escolher revisor, enviar artigo para revisor, etc, que precisam ser realizadas e elas são executadas de acordo com as prioridades. O processo de realização das revisões é feito mediante negociação entre o chair, os membros do comitê do programa e os revisores (Garcia, 2004).

Até o prazo de submissão, os autores podem submeter seus artigos e modificá-los. Após a data limite para submissão, o chair recebe os artigos e distribui um conjunto de artigos para os revisores considerando a sua área de pesquisa. O sistema tem um parâmetro de configuração que afirma que o artigo precisa ser revisado por “n” revisores. Para isso, o chair continua tentando alocar os revisores durante um período limitado de tempo, caso não consiga a responsabilidade de revisão passa a ser do chair (Silva, Noya e Lucena, 2005).

Para realizar a alocação de documentos para revisão o chair precisa enviar uma proposta de revisão de artigo para um revisor que vai avaliar a proposta e confirmar com o chair se aceita ou não revisar o artigo.

5.3 Execução da TEKBS utilizando o EC

A execução da TEKBS utilizando o EC foi feita a partir da segunda atividade da TEKBS “Selecionar e Armazenar Metas Flexíveis” (Figura 4.5), uma vez que a

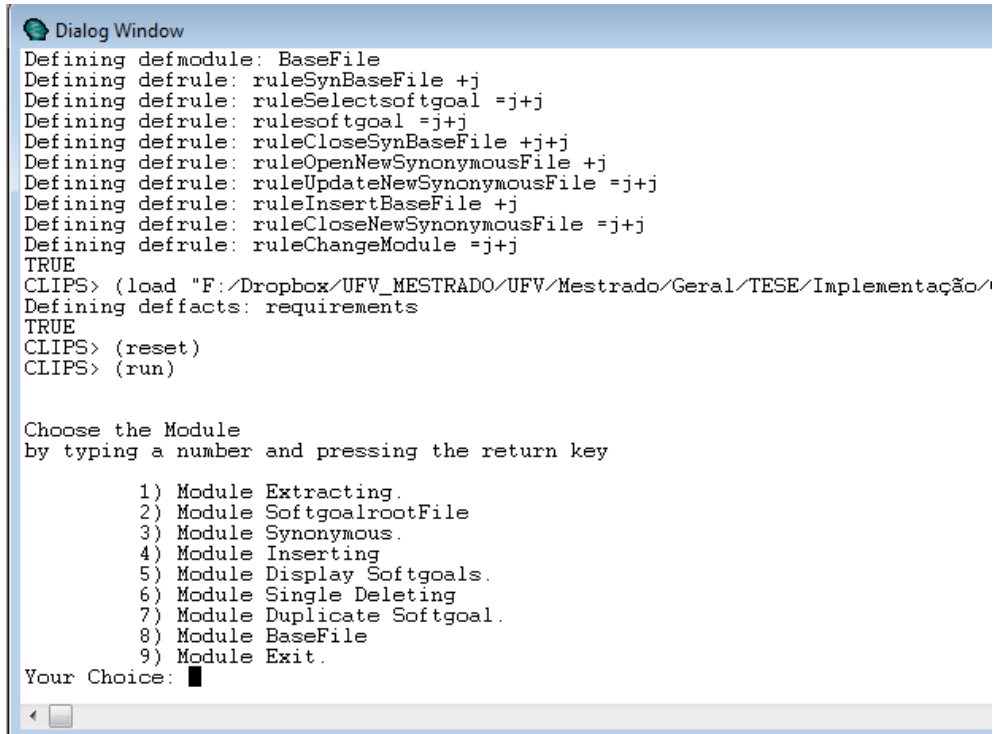
primeira atividade “Elicitar Metas Flexíveis” tem por finalidade estruturar as informações do domínio de acordo com a linguagem CLIPS e declarar as variáveis que serão utilizadas ao longo da implementação.

Para iniciar o processo de utilização da TEKBS é necessário abrir a ferramenta CLIPS e carregar a base de regras e a base de fatos, nesta ordem, uma vez que é preciso declarar todas as variáveis e criar todos os módulos para depois inserir os dados. No caso da TEKBS é necessário carregar a base de regras (Apêndice A) chamada de “rulebase.clp” na versão 1.0 e as bases de fatos (Apêndice B) “knowledgeBase.clp” que contém todas as informações elicítadas por (Cunha, 2014).

5.3.1 Selecionar e Armazenar Metas Flexíveis

Com as bases já carregadas no CLIPS, subseção 5.3 foram inseridos os comandos (reset) e (run) para executá-las. O comando (reset) possui três funcionalidades: a) remover fatos que possam existir na lista de fatos do CLIPS e remover regras registradas na agenda; b) inserir o fato inicial que no CLIPS é considerado o f-0; e c) inserir fatos existentes na base de fatos com instruções “deffacts”. Com a aplicação dessas três instruções garante-se o funcionamento de forma correta da técnica sem interferência de outros dados. O “(run)” é responsável por carregar todos os fatos da base de fatos e executar as regras da base de regras, logo para ter acesso as informações da TEKBS utilizou ambo comandos.

A Figura 5.1 mostra a primeira tela após carregar as bases no CLIPS e inserir os comandos (reset) e (run).



```

Dialog Window
Defining defmodule: BaseFile
Defining defrule: ruleSynBaseFile +j
Defining defrule: ruleSelectsoftgoal =j+j
Defining defrule: rulesoftgoal =j+j
Defining defrule: ruleCloseSynBaseFile +j+j
Defining defrule: ruleOpenNewSynonymousFile +j
Defining defrule: ruleUpdateNewSynonymousFile =j+j
Defining defrule: ruleInsertBaseFile +j
Defining defrule: ruleCloseNewSynonymousFile =j+j
Defining defrule: ruleChangeModule =j+j
TRUE
CLIPS> (load "F:/Dropbox/UFV_MESTRADO/UFV/Mestrado/Geral/TESE/Implementação/
Defining deffacts: requirements
TRUE
CLIPS> (reset)
CLIPS> (run)

Choose the Module
by typing a number and pressing the return key

    1) Module Extracting.
    2) Module SoftgoalrootFile
    3) Module Synonymous.
    4) Module Inserting
    5) Module Display Softgoals.
    6) Module Single Deleting
    7) Module Duplicate Softgoal.
    8) Module BaseFile
    9) Module Exit.
Your Choice: █

```

Figura 5.1: Utilização dos comandos (reset) e (run).

Na Figura 5.1 percebe-se que todas as regras e módulos foram carregados e que após a compilação foi exibido o menu permitindo o engenheiro de requisitos escolher qual módulo desejaria executar.

Por iniciar o sistema foi preciso escolher o “Module Extracting” para que fosse criada a memória de trabalho inicial dentro do CLIPS. Com a escolha do “Modulo Extracting”, o tipo e o tópico que compõem a MF raiz foi selecionada da base de dados de dados da i*Get e inserida em uma nova base chamada de “basesoftgoal”. Além disso, foi criado um arquivo chamado "softgoalFile.clp" contendo todas as MFs raízes elicitadas (Apêndice C) que posteriormente serão utilizadas para encontrar seus sinônimos utilizando o aplicativo JAVAS; Figura 5.4.

A Figura 5.2 ilustra a execução dos módulos, sendo possível verificar a circularidade fornecida pelo menu e pelos módulos, onde após a finalização do módulo acionado imediatamente o engenheiro de requisitos foi direcionado para a escolha de uma nova atividade, podendo ser acionado qualquer outro módulo. Neste caso foi acionado o módulo de exibição de MFs para auxiliar o acompanhamento em tempo real da elicitação.

```

Dialog Window

*** You chose Module Extracting ***

Data successfully extracted

Choose the Module
by typing a number and pressing the return key

    1) Module Extracting.
    2) Module SoftgoalrootFile
    3) Module Synonymous.
    4) Module Inserting
    5) Module Display Softgoals.
    6) Module Single Deleting
    7) Module Duplicate Softgoal.
    8) Module BaseFile
    9) Module Exit.
Your Choice: 2

*** You chose Module SoftgoalrootFile ***
[Opened file softgoalFile.clp.]
[Closed file softgoalFile.clp].

Choose the Module
by typing a number and pressing the return key

    1) Module Extracting.
    2) Module SoftgoalrootFile
    3) Module Synonymous.
    4) Module Inserting
    5) Module Display Softgoals.
    6) Module Single Deleting
    7) Module Duplicate Softgoal.
    8) Module BaseFile
    9) Module Exit.
Your Choice: 5

*** You chose Module DisplaySoftgoal ***
Id:(gen12)Softgoal: quality [review]
Id:(gen11)Softgoal: punctuality [publication]
Id:(gen9)Softgoal: acknowledge [committee]
Id:(gen6)Softgoal: honest [review]
Id:(gen5)Softgoal: quality [article]
Id:(gen3)Softgoal: no delay [review]
Id:(gen2)Softgoal: fair [review]
Id:(gen1)Softgoal: correct [evaluation]

```

Figura 5.2: Execução do Module Extracting e Print Softgoals.

O processo descrito desde a inserção das bases de dados e conhecimento no CLIPS até a extração, armazenamento e exibição das MFs raízes pode ser visto no fluxo de atividades na Figura 5.3.

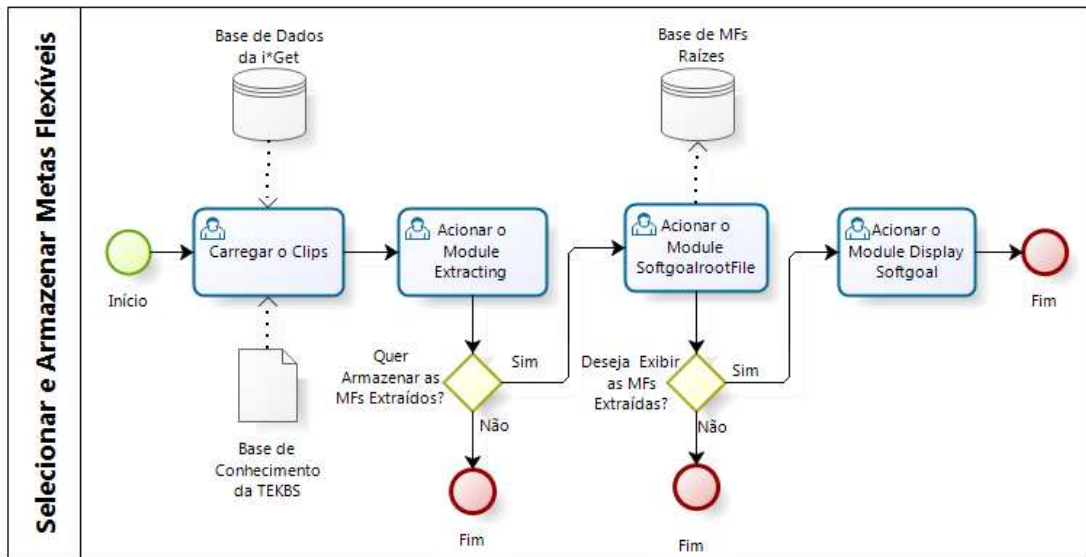


Figura 5.3: Fluxo da atividade Selecionar e Armazenar Metas Flexíveis.

5.3.2 Relacionar Metas Flexíveis com Sinônimos

Para relacionar as MFs raízes com os sinônimos o primeiro passo foi acessar o dicionário de sinônimos de língua inglesa Wordnet por meio do aplicativo JAVAS. Para que fosse realizada a pesquisa foi carregado no JAVAS o arquivo "softgoalFile.clp" e solicitado a realização da pesquisa por meio do botão "Pesquisa online no Wordnet".

A Figura 5.4 mostra a utilização do JAVAS para levantar sinônimos das MFs raízes.

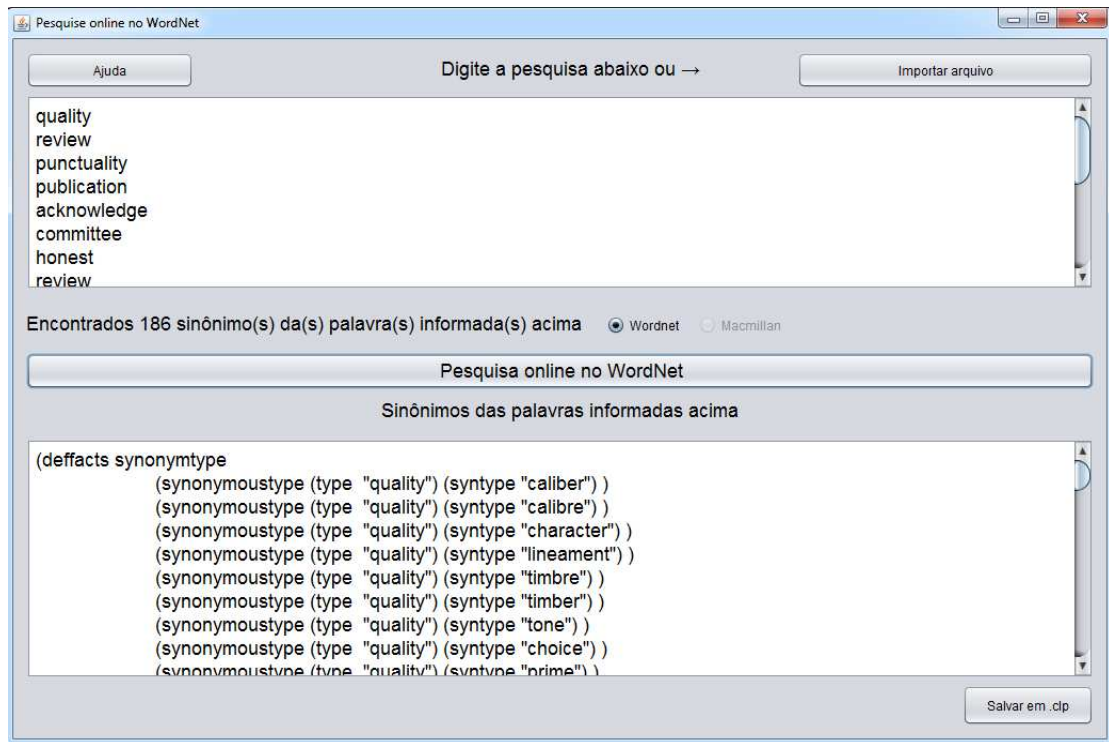


Figura 5.4: Levantamento de sinônimos utilizando o JAVAS.

O resultado foi exibido e para utilizá-lo na TEKBS o engenheiro de requisitos precisou salvar a pesquisa na extensão “.clp” (Apêndice D) e posteriormente recarregá-la na TEKBS.

Para carregá-la, primeiro foi preciso acionar o “Module Exit” para a opção “load” ficar ativa. No entanto, o procedimento de carregar não forneceu ao engenheiro de requisitos possibilidades de utilização das informações, sendo assim, fez-se necessário a compilação da base com os comandos (reset) e (run) para que os dados da BS pudessem ficar disponíveis na memória de trabalho e o menu disponível para o engenheiro de requisitos.

A Figura 5.5 mostra o processo de inserção da BS na TEKBS.

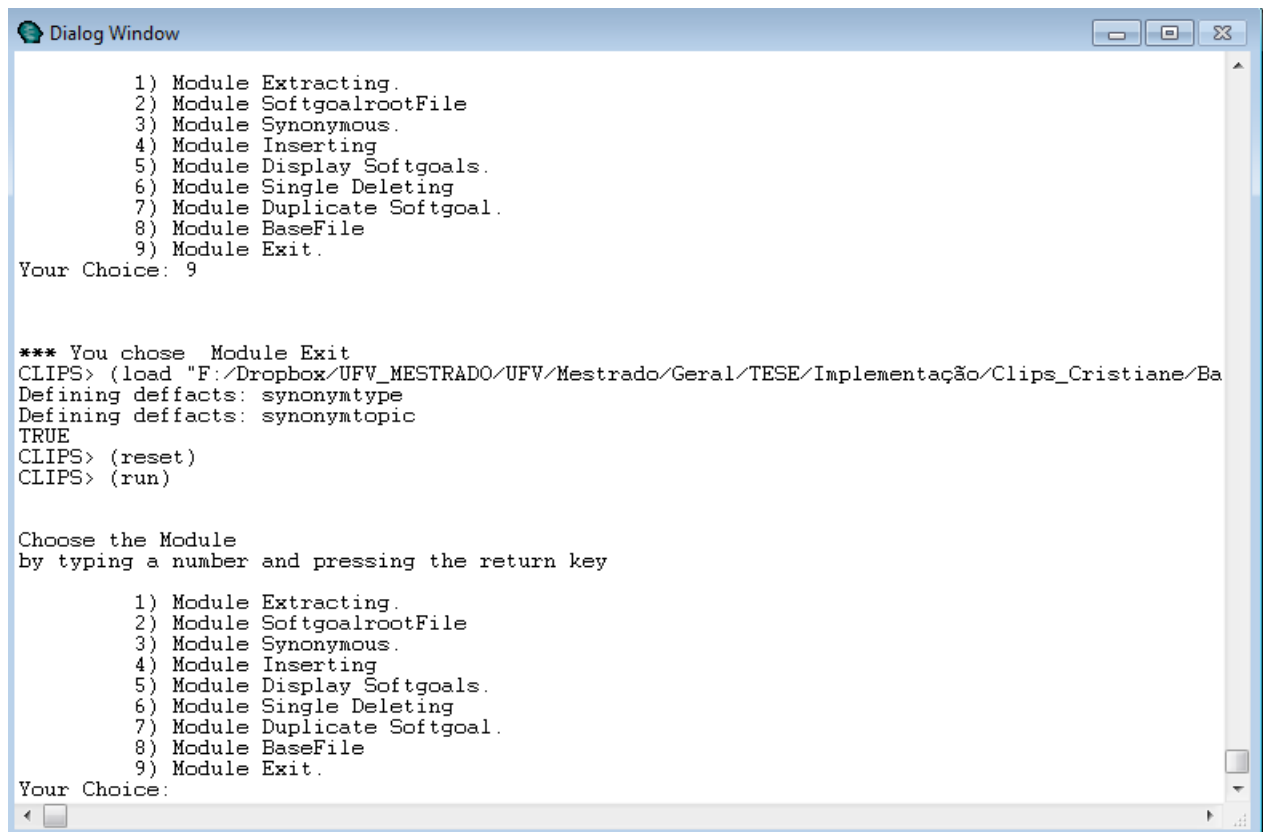


Figura 5.5: Inserção da BS na TEKBS.

Após a inserção da BS na TEKBS acionou-se o “Module Synonymous” que extraiu os sinônimos da BS e inseriu na base “synSoftgoal” criando uma memória de trabalho de sinônimos para serem utilizadas ao longo do processo.

Pela necessidade de acompanhar as atividades em tempo real, o engenheiro de requisitos solicitou que os sinônimos também fossem exibidos. A Figura 5.6 mostra parte dos sinônimos carregados.

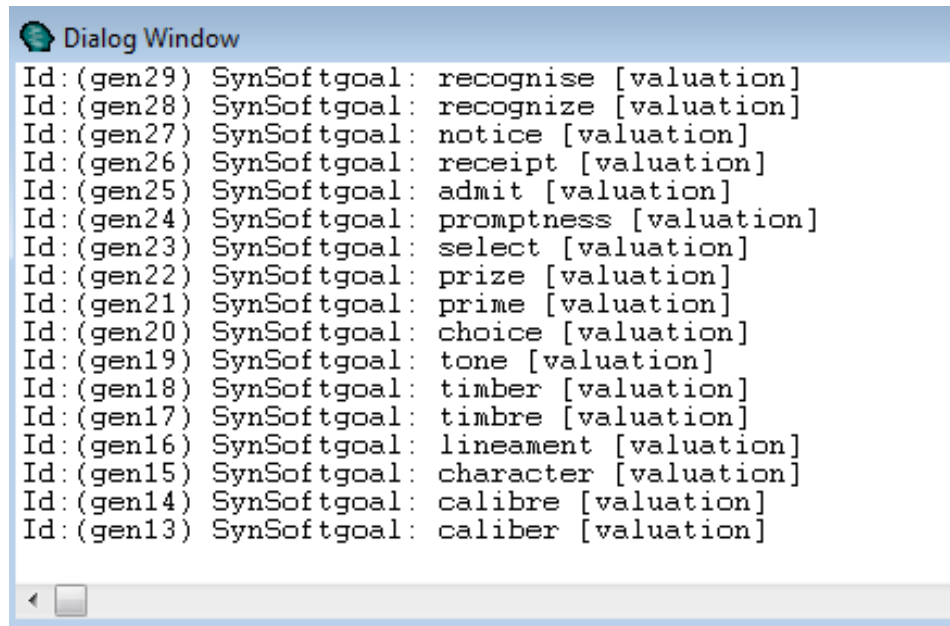


Figura 5.6: Metas Flexíveis sinônimas impressas na tela do CLIPS.

Assim como as MFs raízes seus sinônimos ao serem inseridos na memória de trabalho recebem uma identificação. Essa identificação facilita o processo de manipulação das MFs.

Para facilitar o entendimento da atividade “Relacionar Metas Flexíveis com Sinônimos” foi criado um fluxo mostrando todas as atividades, bem como os artefatos de entrada e saída de cada fase como pode ser visto na Figura 5.7.

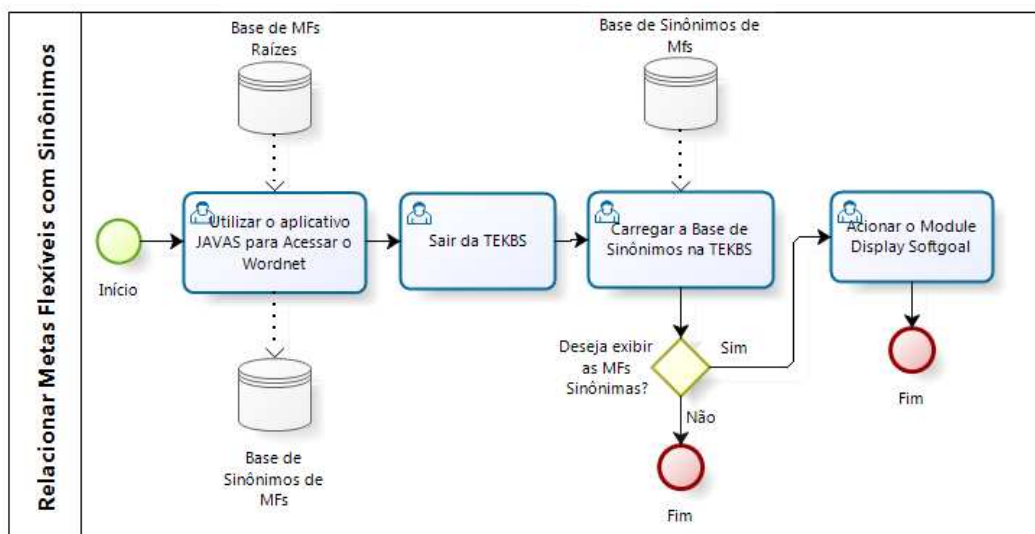


Figura 5.7: Fluxo da atividade Relacionar Metas Flexíveis com Sinônimos.

Como visto nas Figuras 5.3 e 5.7, neste ponto da execução da TEKBS o engenheiro de requisitos tem a sua disposição uma memória de trabalho com MFs raízes e MFs sinônimas, porém é preciso avaliar a acoplamento entre as MFs raízes e seus sinônimos.

Para verificação do acoplamento utiliza-se a expertise do engenheiro de requisitos que vai analisar se os sinônimos encontrados pela técnica são de fato sinônimos ou relevantes para o contexto de aplicação do EC. O processo de análise pode ser visto na subseção 5.4.

Após a análise o engenheiro de requisitos utilizou a atividade de “Refinar Metas Flexíveis” para alinhar os sinônimos com o domínio EC. O processo de refinamento foi feito por meio dos módulos: a) “Module Single Deleting”; b) “Module Duplicate Softgoal”; e c) “Module Inserting” e será descrito nas próximas subseções e seu fluxo completo pode ser visto na Figura 5.11.

5.3.3 Refinamento de Metas Flexíveis

Após a finalização do processo de análise de acoplamento das MFs sinônimas, o engenheiro de requisitos volta à TEKBS para alinhar as informações contidas com o contexto do EC.

Na etapa de refinamento é possível o engenheiro de requisitos inserir novas MFs, excluir MFs e ainda excluir as duplicidades, podendo ser executada na ordem de necessidade do engenheiro.

Para o contexto do EC a primeira atividade foi excluir as MFs sinônimas que não fazem parte do contexto acionando o “Module Single Deleting”. A primeira MF sinônima a ser excluída foi a “caliber[valuation]” como mostra a Figura 5.8. O processo foi repetido inúmeras vezes até todas as MFs serem retiradas.

```

Dialog Window
Id:(gen23) SynSoftgoal: select [valuation]
Id:(gen22) SynSoftgoal: prize [valuation]
Id:(gen21) SynSoftgoal: prime [valuation]
Id:(gen20) SynSoftgoal: choice [valuation]
Id:(gen19) SynSoftgoal: tone [valuation]
Id:(gen18) SynSoftgoal: timber [valuation]
Id:(gen17) SynSoftgoal: timbre [valuation]
Id:(gen16) SynSoftgoal: lineament [valuation]
Id:(gen15) SynSoftgoal: character [valuation]
Id:(gen14) SynSoftgoal: calibre [valuation]
Id:(gen13) SynSoftgoal: caliber [valuation]

Choose the Module
by typing a number and pressing the return key

    1) Module Extracting.
    2) Module SoftgoalrootFile
    3) Module Synonymous.
    4) Module Inserting
    5) Module Display Softgoals.
    6) Module Single Deleting
    7) Module Duplicate Softgoal.
    8) Module BaseFile
    9) Module Exit.
Your Choice: 6

*** You chose Module Single Deleting ***
Enter below the two softgoals field:

the synonyms of the <[TYPE]QUALITY ATTRIBUTE>
                    and the <[TOPIC]SUBJECT/OBJECT LAL>.

Enter with QUALITY ATTRIBUTE: caliber
Enter with SUBJECT/OBJECT LAL: valuation

The SynSoftgoals were successfully deleted

```

Figura 5.8: Exclusão de MF Sinônimas fora do contexto EC.

No início do processo de seleção de sinônimos tínhamos 1827 MFs sinônimas elicitadas, dessas, somente 306 permaneceram na memória de trabalho, ver Figura 5.9, por possuírem o mesmo significado das MFs raízes. Após a análise de contexto que tinha como objetivo analisar as MFs pertencentes ao contexto do EC e após o refinamento das MFs, somente 80 MFs se mantiveram na memória de trabalho, sendo 8 MFs raízes e 72 MFs sinônimas.

```

*** You chose Module PrintSoftgoal ***
Id:(gen1827) SynSoftgoal: set [reappraisal]
Id:(gen1826) SynSoftgoal: adjust [reappraisal]
Id:(gen1825) SynSoftgoal: slump [reappraisal]
Id:(gen1824) SynSoftgoal: decline [reappraisal]
Id:(gen1823) SynSoftgoal: sort out [reappraisal]
Id:(gen1822) SynSoftgoal: discipline [reappraisal]
Id:(gen1821) SynSoftgoal: even up [reappraisal]
Id:(gen1820) SynSoftgoal: even off [reappraisal]
Id:(gen1819) SynSoftgoal: even out [reappraisal]
Id:(gen1818) SynSoftgoal: make up [reappraisal]
Id:(gen1817) SynSoftgoal: counterbalance [reappraisal]
Id:(gen1816) SynSoftgoal: chasten [reappraisal]
Id:(gen1815) SynSoftgoal: objurgate [reappraisal]
Id:(gen1814) SynSoftgoal: castigate [reappraisal]
Id:(gen1813) SynSoftgoal: chastise [reappraisal]

*** You chose Module PrintSoftgoal ***
Id:(gen306) SynSoftgoal: promptness [rating]
Id:(gen305) SynSoftgoal: recognise [rating]
Id:(gen304) SynSoftgoal: recognize [rating]
Id:(gen303) SynSoftgoal: admit [rating]
Id:(gen302) SynSoftgoal: fair [rating]
Id:(gen301) SynSoftgoal: true [rating]
Id:(gen300) SynSoftgoal: reliable [rating]
Id:(gen299) SynSoftgoal: dependable [rating]
Id:(gen298) SynSoftgoal: honorable [rating]
Id:(gen297) SynSoftgoal: select [rating]
Id:(gen296) SynSoftgoal: choice [rating]
Id:(gen295) SynSoftgoal: evenhandedly [rating]
Id:(gen294) SynSoftgoal: fairly [rating]
Id:(gen293) SynSoftgoal: honest [rating]
Id:(gen292) SynSoftgoal: clean [rating]

```

Figura 5.9: Memória de trabalho antes e após a utilização do "Module Single Delinting".

Com a finalização o processo de exclusão de MFs é permitido ao engenheiro de requisitos inserir novas MFs que segundo sua avaliação são necessárias para o entendimento do contexto do EC e que não foram elicitadas.

Para efetuar a inserção é preciso acionar o “Module Inserting” responsável por receber os tipos e os tópicos e inseri-los na memória de trabalho. Uma das MFs inseridas foi a “genuine[search]” como mostra a Figura 5.10.

O processo de inserção assim como de exclusão é contínuo podendo ser efetuado inúmeras vezes ao longo do processo de elicitação. Assim, caso ocorra alguma alteração de escopo de projeto é possível modificar as MFs elicitadas e posteriormente o documento de requisitos.

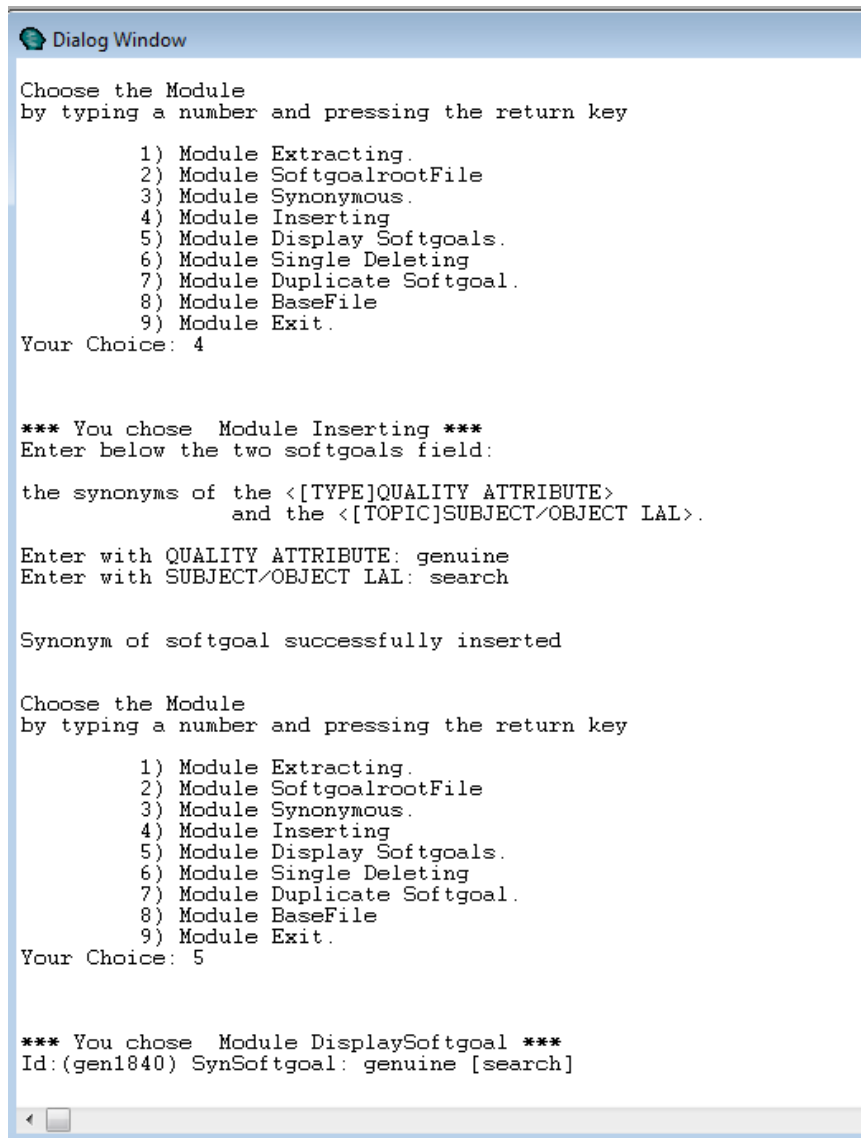


Figura 5.10: Inserção de MF sinônimas.

Por fim, o engenheiro de requisitos acionou o “Module Deletes Duplicate Softgoal”, para eliminar da memória de trabalho MFs duplicadas, mantendo somente uma meta de cada.

Como é exibido pela Figura 5.11 ao final do processo de refinamento o engenheiro de requisitos possui uma memória de trabalho completamente alinhada com suas necessidades para um contexto específico, neste caso para o contexto de EC, que poderá facilitar o entendimento do problema, criação de modelos, proporcionar maior facilidade e qualidade na descrição documento de Especificações de Requisitos de Software.

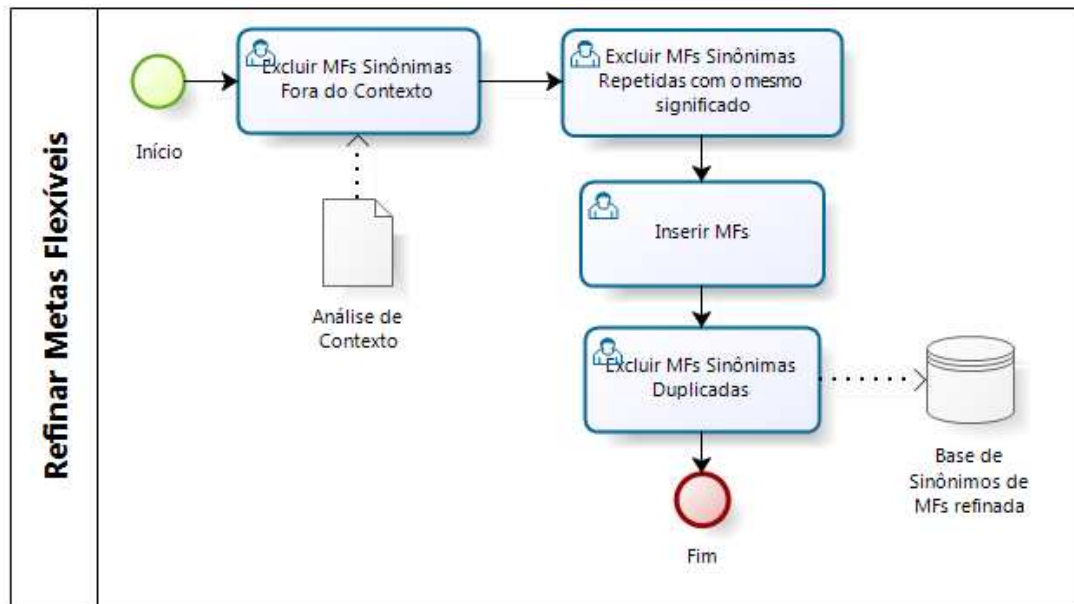


Figura 5.11: Fluxo da atividade Refinar Metas Flexíveis

5.3.4 Armazenar Metas Flexíveis

Com a finalização do processo de refinamento, a TEKBS permite ao engenheiro de requisitos armazenar todas as informações elicítadas. Para que essa ação pudesse ocorrer foi preciso acionar o módulo “Module BaseFile” pertencente a atividade “Armazenar Metas Flexíveis” que permite a criação de dois arquivos, um contendo todas as MFs elicítadas chamado de "BasesoftgoalFile.clp" e outro que cria uma nova base de dados de sinônimos válidos para o contexto analisando chamado de "NewSynonymousFile.clp", ver Figura 5.12.

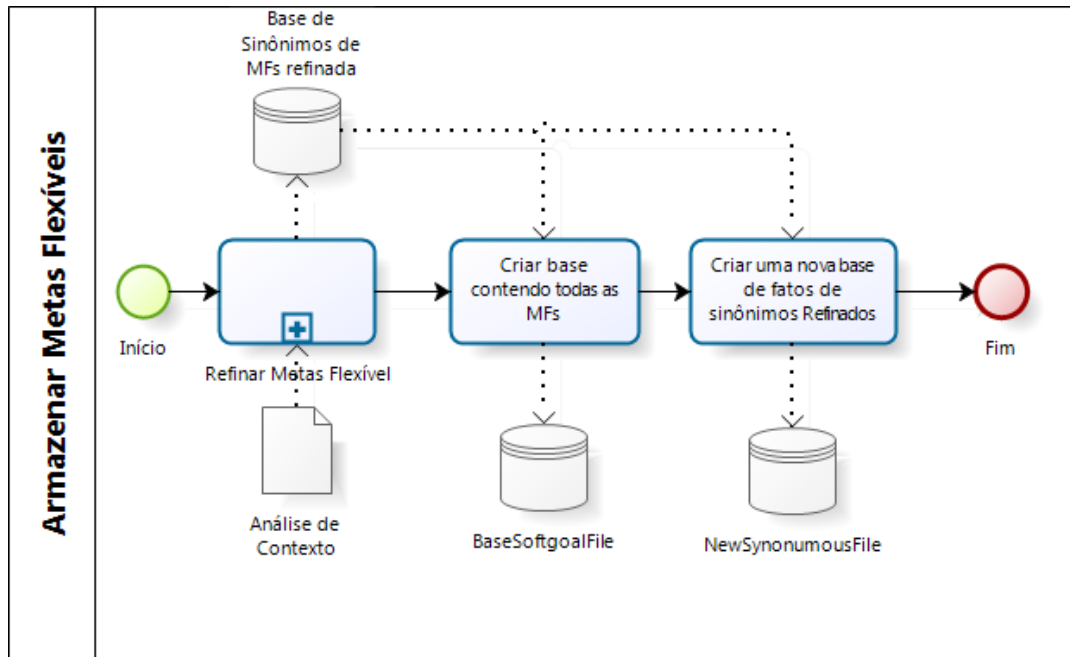


Figura 5.12: Fluxo da atividade Armazenar Metas Flexíveis.

Com esses dois arquivos o engenheiro de requisitos pode acessar as informações constantemente e caso tenha necessidade de alterar as MFs elicítadas utilizando o processo de refinamento, não é preciso refazer todo o processo, somente carregar a "NewSynonymousFile.clp" acionar o módulo de interesse e fazer as alterações.

Concluindo, para sair da TEKBS basta acionar o "Module Exit" que vai finalizar todos os processos.

5.4 Análise de acoplamento das Metas Flexíveis Sinônimas

A verificação de acoplamento entre MFs raízes e MFs sinônimas é de extrema importância, pois ela garantirá que o refinamento das MFs seja feito de forma a manter as informações elicítadas pela TEKBS contenha somente MFs pertencentes ao domínio, neste caso ao EC.

O processo de análise foi composto de sete passos:

1. Elicitar os sinônimos das MFs raízes utilizando a base de sinônimos de língua inglesa Wordnet;

2. Analisar o significado de cada parte da MF sinônima isoladamente e verificar se condizem com o significado das partes da MF raiz;
3. Selecionar somente os tipos e os tópicos que mantiveram o sentido das MFs raízes;
4. Combinar os tipos e os tópicos selecionados no item 3, gerando novas MFs;
5. Avaliar se as novas MFs elicitadas fazem parte do contexto de estudo;
6. Das MFs que pertencem ao contexto, quais são novas e quais já haviam sido elicitadas por (Oliveira, 2008) que realizou a análise do EC manualmente e quais já haviam sido elicitadas por (Cunha, 2014) que semi-automatizou o processo de (Oliveira, 2008); e
7. Comparar os resultados se a base de sinônimos utilizada tivesse sido o Macmillan (Macmillan, 2014) ou o Thesaurus (Thesaurus, 2014).

A descrição do primeiro passo da análise pode ser vista na subseção 5.3.2, pois é uma atividade da TEKBS. Os demais passos serão descritos nas subseções seguintes.

5.4.1 Processo de seleção de sinônimos

O processo de seleção dos sinônimos abrange os passos dois, três e quatro da subseção 5.4. Para buscar os sinônimos utilizou-se somente o primeiro nível de sinônimos das MFs raízes, pois a partir do segundo nível, sinônimo de sinônimo, as buscas se tornavam irrelevantes.

A primeira atividade foi analisar o significado de cada parte que compõe a MF – Tipo[tópico]. Para essa investigação foi feita duas listas, uma contendo todos os tipos e outras contendo todos os tópicos separados por MFs raízes. Para cada uma das listas foi verificado na página web do Wordnet qual era a descrição de significado e contrastado com o significado literal da palavra utilizando o Dicionário Michaelis Inglês/Português (Michaelis, 2010). Com essas duas informações foram selecionadas as MFs que tinham o mesmo significado que a MF raiz.

A página web do Wordnet foi utilizada por apresentar junto aos resultados das pesquisas o significado de cada um dos sinônimos retornados. Essa mesma funcionalidade não está presente quando a pesquisa de sinônimos é feita usando o JAVAS que retorna todos os sinônimos da palavra passada sem considerar ou apresentar ao engenheiro de requisitos seu significado.

Na pesquisa realizada no Wordnet, de todos os 63 tipos e dos 29 tópicos selecionados sem repetição, 18 tipos e 17 tópicos possuíam o mesmo significado que a MF raiz, perfazendo um total de 306 MFs sinônimas que foram analisadas junto ao contexto do EC.

5.4.2 Análise de contexto das Metas Flexíveis sinônimas

A análise de contexto foi realizada utilizando as 306 MFs sinônimas selecionadas formadas da união dos 18 tipos combinados com cada um dos 17 tópicos. Ao final dessa combinação possuía-se 17 grupos de MFs cada um com 18 MFs diferentes em sua nomenclatura. Porém, desses 17 grupos 8 foram descartados, uma vez que já existiam entre os 9 grupos restantes grupos de MFs com o mesmo significado.

Uma nova seleção foi realizada com os 9 grupos de combinações de MFs considerando as descrições: a) não pertencer ao contexto do EC considerados como “Fora de contexto”; b) pertencer ao contexto, mas é uma MF repetida dentro do grupo de combinação analisado, chamada de “repetida”; e por fim c) ser pertencente ao contexto do EC nomeada de “válidas”.

O Gráfico 5.1, mostra a distribuição das MFs considerando os 8 grupos inicialmente descartado e o Gráfico 5.2 exhibe todas as MFs descartadas por não pertencem ao contexto de EC e as válidas que fazem parte do contexto.

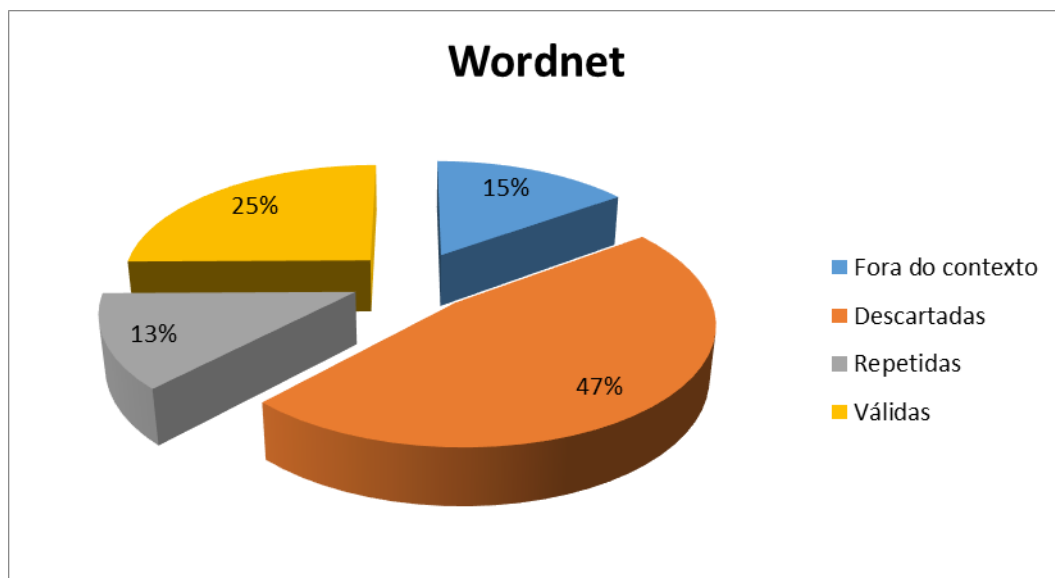


Gráfico 5.1: Metas Flexíveis selecionadas por grupos de análises.

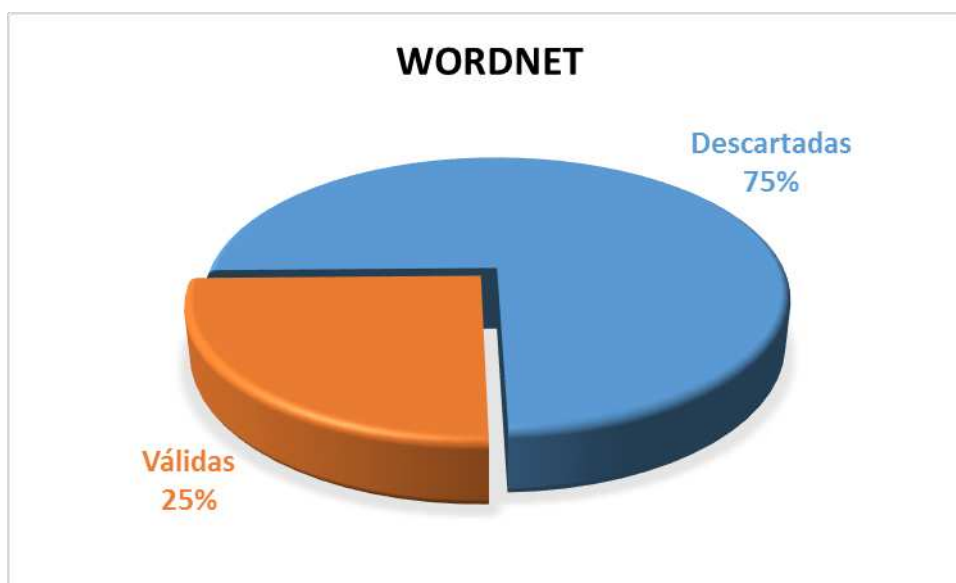


Gráfico 5.2: MFs descartadas e MFs válidas.

Como pode ser visto na análise do Gráfico 5.1, 47% das MFs sinônimas foram inicialmente descartadas por existirem outros grupos com o mesmo significado. 15% estavam fora do contexto do EC, 13% representava repetição de alguma outra MF dentro do grupo de combinação analisado e 25% das MFs foram consideradas válidas para o contexto do EC.

O Gráfico 5.2 ilustra o número de MFs descartadas (75%) e o número de MFs válidas (25%). Além das 47% que foram inicialmente descartadas. Os 15% que

estavam fora do contexto e os 13% que eram repetidas também foram desconsideradas.

Por meio dessa análise o engenheiro possui em suas mãos um total de 77 MFs para auxiliar na descrição do documento de requisitos. Essas MFs poderão ser inseridas no contexto da TEKBS para melhorar o entendimento do problema e a descrição do documento de requisitos.

Contudo, foi preciso analisar um novo fator, quais dessas 77 MFs elicítadas já haviam sido elicítadas pela i*Get (Cunha, 2014) e já pertenciam a memória de trabalho da TEKBS evitando a inserção de MF já presentes gerando redundância na memória de trabalho.

Para essa análise foi feita uma nova comparação entre as MFs válidas e a base de dados da i*Get (Cunha, 2014). O Gráfico 5.3 exibe o resultado da comparação.

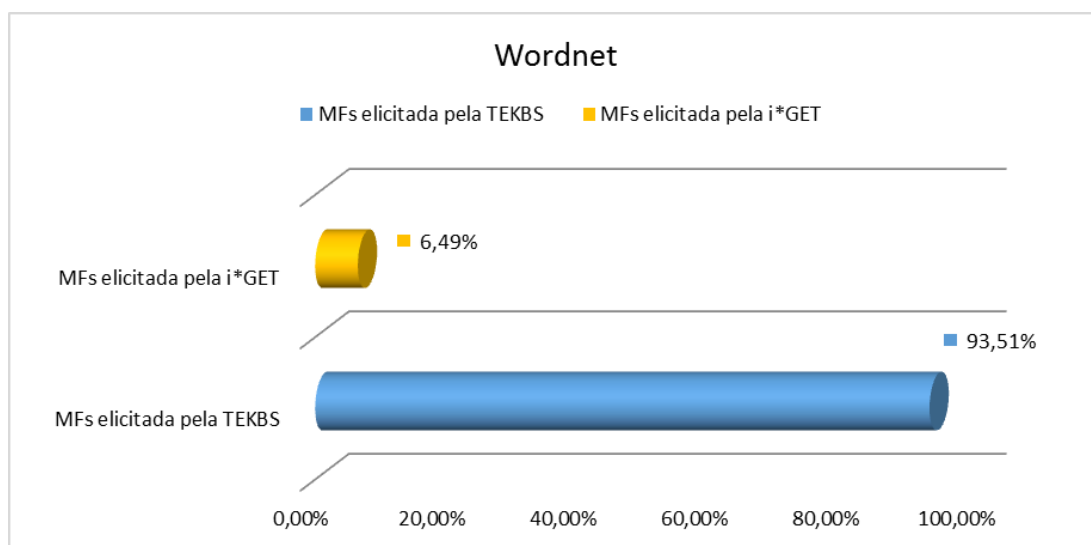


Gráfico 5.3: Comparação entre as MF Válidas e Base de Dados de (Cunha, 2014).

Como pode ser visto no Gráfico 5.3 um total de 93,51% (72) MFs novas foram elicítadas acrescentando à memória de trabalho novas informações. Além disso, das 72 MFs novas um total de 37 identificava tópicos que ainda não tinham sido considerados como: “rating” no sentido de aceite ou rejeição, “review”, “article”, “inspection” e “clause” que combinados com os tipos indicavam necessidades ainda não satisfeitas ou analisadas. As 35 restantes apesar de pertencerem a grupos de tópicos já identificados, “reappraisal”, “valuation”,

“commission” e “publishing”, também indicaram novas necessidades que precisavam ser consideradas. Um exemplo é a MF “*true [valuation]*” que insere na memória de trabalho a necessidade de realizar uma correta [avaliação], mas não pode garantir que ela seja feita de forma confiável e nem imparcial. Neste caso, essas necessidades passam a ser atendidas pelas MFs “*reliable [valuation]*” e “*evenhandedly [valuation]*”, podendo ser garantido que as 3 necessidades serão levadas em consideração na execução do processo.

Essas informações vêm comprovar a adequabilidade e efetividade da técnica TEKBS ao processo de elicitação. O impacto da sua utilização é positivo uma vez que o número maior de MFs foram elicítadas. Com as novas MFs, novos itens que poderiam deixar de ser abordados no entendimento do problema e na descrição do documento de requisitos, passam a ser considerados, podendo trazer ao final do processo um sistema mais alinhado a necessidade do cliente.

Com os dados atualizados, as novas MFs podem ser inseridas na TEKBS e as MFs que não fazem parte do contexto do EC eliminadas. O processo de Refinamento de Metas Flexíveis pode ser visto na subseção 5.3.3.

5.4.3 Comparação da TEKBS com o processo Manual realizado por (Oliveira, 2008)

Como relatado, Oliveira (Oliveira, 2008) realizou uma análise do Expert Committee System junto ao processo de criação do Método ERi*c.

Em sua primeira etapa “Elicitar Metas de Atores”, mais especificamente na subatividade “Definir AGFL” ele também faz a elicitação de MF para o domínio EC, porém de forma manual. Assim, utilizou-se a Tabela 4.2.16 contida na Tese de (Oliveira, 2008) na página 147, onde ele reúne todas as MC e MF elicítadas em seu trabalho para comparar com a elicitação realizada pela TEKBS.

No processo de elicitação de (Oliveira, 2008) 37 MFs foram elicítadas, porém dessas 16 se repetiam em algum momento da análise, ficando um total de 21 MFs elicítadas desconsiderando as duplicidades. No processo realizado pela TEKBS, 306 MFs foram elicítadas, dessas, 229 foram desconsideradas como foi mostrado na

subseção 5.4.2, restando um total de 77 MFs consideradas válidas para o contexto EC.

Uma visão geral da diferença de elicitação pelos dois processos pode ser visto no Gráfico 5.4.

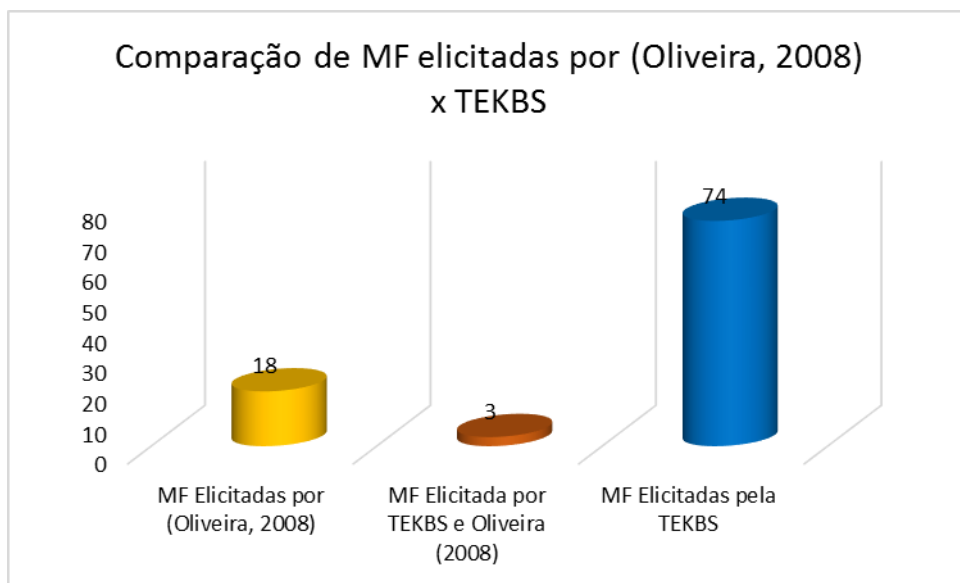


Gráfico 5.4: Comparação de MF elicadas por (Oliveira, 2008) x TEKBS.

A análise do Gráfico 5.4, revela que de todas as 77 MFs elicadas pela TEKBS somente 3 estavam presentes no processo manual realizado por (Oliveira, 2008) tendo a TEKBS um ganho de 74 novas MFs elicadas. Ressalta-se que no processo manual foram elicadas 18 outras MFs que não foram elicadas pela TEKBS diretamente, no entanto muitas delas podem ser alcançadas unindo algumas MFs elicadas pela TEKBS. Um exemplo é a MF qualidade[publicação] elicada por (Oliveira, 2008) e que não foi elicada diretamente pela TEKBS, mas que pode ser alcançada pelas união das MF pontualidade [publicação], justa[publicação], correta[publicação], confiável[publicação], imparcialidade[publicação], etc. Assim, pode-se garantir que a grande maioria das MFs elicadas por (Oliveira, 2008) e que não foram elicadas diretamente pela TEKBS consigam ser atendidas de forma

parcial ou integral. Outra possibilidade, é a inserção das 18 MFs elicítadas utilizando o “Module Inserting” da atividade “Refinar Metas Flexíveis”.

5.4.4 Comparação do processo realizado pela TEKBS utilizando os Dicionários de sinônimos Macmillan e Thesaurus.

Apesar das dificuldades de acesso as informações relacionadas aos dicionários Macmillan (Macmillan, 2014) e Thesaurus (Thesaurus, 2014), uma análise de sua utilização foi feita utilizando a página web de cada uma para verificar quais seriam os resultados se a TEKBS utilizasse alguns deles para auxiliar o processo de elicitação.

Como não foi possível ter acesso às API’s das páginas o primeiro passo foi criar a base de sinônimos manualmente. Para isso utilizou-se as MFs raízes e localizou manualmente todos os sinônimos utilizando os dois dicionários.

Após a obtenção dos sinônimos foram seguidos os passos de 2 - 6 utilizados para análise do Wordnet. A Tabela 5.1 compila os dados de análise dos significados dos tipos e tópicos utilizando os 3 dicionários.

Tabela 5.1: Tabela comparativa dos dicionários de sinônimos de Língua Inglesa.

Metas Flexíveis Sinônimas	Elicitadas sem Repetição	Mesmo significado	% Mesmo significado
Wordnet	1827	306	16,75
Macmillan	5904	420	7,11
Thesaurus	2460	504	20,49

A Tabela 5.1 ilustra que apesar do Macmillan permitir elicitar um grande número de MFs sinônimas, somente 7,11% dos tipos e dos tópicos possuíam o mesmo significado que as MFs raízes. Neste ponto da análise o melhor dicionário foi o Thesaurus que permitiu elicitar 2460 MFs sinônimas e dessas 20,49% possuíam o tipo e o tópico com o mesmo significado que a MFs raiz, o Wordnet ficou entre os dois dicionários com 16,75% de tipos e tópicos com o mesmo significado que as MFs raízes.

Com a classificação de significados dos tipos e dos tópicos gerou-se os grupos de combinações de MFs para cada dicionário selecionado considerando as mesmas descrições utilizadas para o Wordnet: a) não pertencer ao contexto do EC considerados como “Fora de contexto”; b) pertencer ao contexto, mas é uma MF repetida dentro do grupo de combinação analisado, chamada de “repetida”; e por fim c) ser pertencente ao contexto do EC nomeada de “válidas”.

Antes da seleção foi preciso excluir os grupos de combinações que possuíam MFs com o mesmo significado em toda a sua extensão. Assim, foram descartados 7 grupos de combinações de MFs do Wordnet, 5 grupos de combinações de MFs do Macmillan e 1 grupo de combinação de MFs do Thesaurus, após essa exclusão o restante foi selecionado com as descrições acima para os 3 dicionários como pode ser vista no Gráfico 5.5.

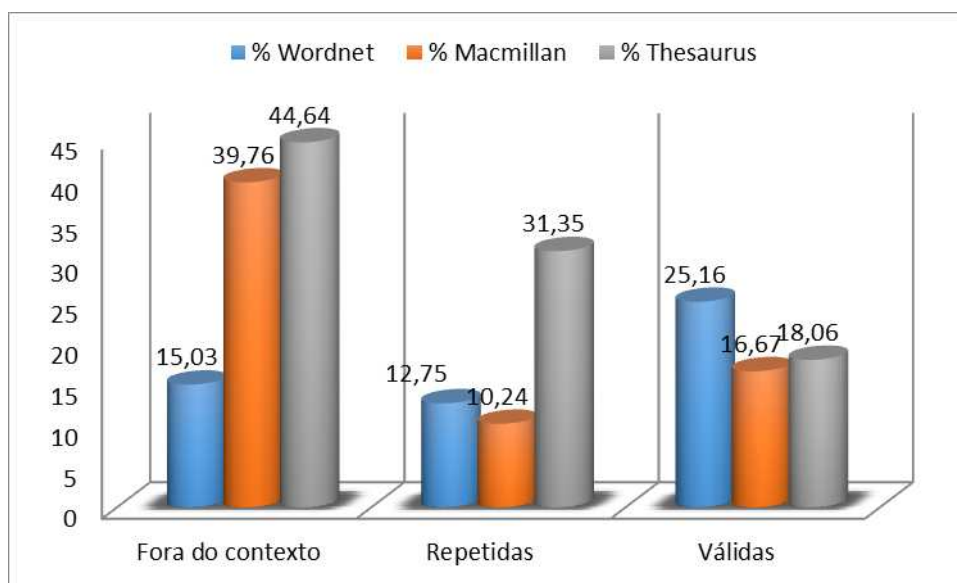


Gráfico 5.5: Metas Flexíveis selecionadas por grupos de análise e por dicionários.

Na análise de contexto das MFs, a pesquisa feita utilizando o Thesaurus retorna o maior número de tipos e tópicos com o mesmo significado das MFs raízes. Após a combinação dos tipos e dos tópicos, para gerar novas MFs, um número muito grande de MFs fora do contexto do EC, 44,64%, e de MFs repetidas, 31,35%, foram identificadas e desconsideradas. Por outro lado, a pesquisa utilizando o Wordnet que era intermediário no número de retorno de tipos e tópicos com o mesmo significado

no processo de combinação criou o menor número de MFs fora do contexto do EC, 15,03% e manteve-se como intermediário na análise das MFs repetidas (12,75%). A pesquisa com o Macmillan retornou 39,76% de MFs fora do contexto do EC e 10,25% de MFs repetidas.

Para as MFs válidas foram desconsideradas todas as repetições juntamente com as MFs fora do contexto. Na análise considerando o contexto do EC o dicionário de língua inglesa que permitiu elicitar o maior número de MFs válidas foi o Thesaurus, que retornou 18,06% - 91 MFs válidas nas pesquisas, em segundo lugar vem o Wordnet com 25,16% - 77 MFs válidas e por último o Macmillan com 16,67%, que representa 70 MFs válidas.

Apesar de o Gráfico 5.4 destacar o Wordnet como o melhor em porcentagens, quando transformado em valores absolutos o Thesaurus permitiu elicitar mais MFs válidas, mesmo com um número maior de MFs desconsideradas. Isso foi possível porque a pesquisa por sinônimos utilizando o Thesaurus foi a que retornou o maior número de MFs com o mesmo sentido, como exposto na Tabela 5.1.

Até o presente momento, analisou-se a validade das MFs diante do contexto EC, o mesmo contexto já foi analisado por (Oliveira, 2008) de forma manual e por (Cunha, 2014) de forma semi-automatizada. Por isso fez-se a comparação da elicitação das MFs válidas junto as duas análises para descobrir quantas MFs já haviam sido elicidadas. A comparação tem por objetivo evitar a inserção de MFs já elicidadas na base de dados da TEKBS, na busca de garantir sua integridade. Os resultados foram compilados na Tabela 5.2.

Tabela 5.2: Comparação de MF elicidadas pela TEKBS versus as elicidadas por (Oliveira, 2008) e por (Cunha, 2014).

Descrição	Wordnet	%	Macmillan	%	Thesaurus	%
MFs elicidadas pela TEKBS	70	90,91	67	95,71	89	97,80
MFs elicidadas por Oliveira (2008) ou (Cunha, 2014)	7	9,09	3	4,29	2	2,20
TOTAL	77	100	70	100	91	100

Na Tabela 5.2, das 77 MFs válidas elicitadas pela KBGET utilizando o dicionário de língua inglesa Wordnet 7 já haviam sido elicitada por (Oliveira, 2008) ou por (Cunha, 2014), quando o dicionário utilizado no processo de elicitacoo é o Macmillan o valor de MFs elicitadas em conjunto reduzem para 3MFs. Por fim, quando utiliza-se o dicionário de língua inglesa o Thesaurus a elicitacoo conjunta cai para somente 2 MFS.

Analisando os dados da Tabela 5.2 foi possvel confirmar a superioridade do Thesaurus no processo de elicitacoo de sinnimos. Dos dicionrios consultados ele foi o que mais retornou MFs que no haviam sido elicitadas nem por (Oliveira, 2008) e nem pela i*Get (Cunha, 2014). O Wordnet ficou como intermedirio com 7 MFs j elicitadas das 77 MFs vlidas e por ltimo o Macmillan com 3 MFs j elicitadas das 70 MFs vlidas.

De forma a simplificar o entendimento das atividades descritas no subitem 5.4 foi gerado um o fluxo de atividades, Figura 5.13, separando as atividades executadas pela TEKBS e as que so feitas com auxlio do engenheiro de requisitos. A Figura 5.13 é dividida em duas raias a primeira (TEKBS) com atividades realizadas de forma automtica pela TEKBS e a segunda (Engenheiro de requisitos) com as atividades que so desenvolvidas de forma semi-automticas com auxlio do engenheiro de requisitos na anlise de acoplamento dos sinnimos das metas flexveis. Todo o processo ser refeito para cada um dos dicionrios de lngua inglesa e somente depois a comparaoo entre os dicionrios é realizada.

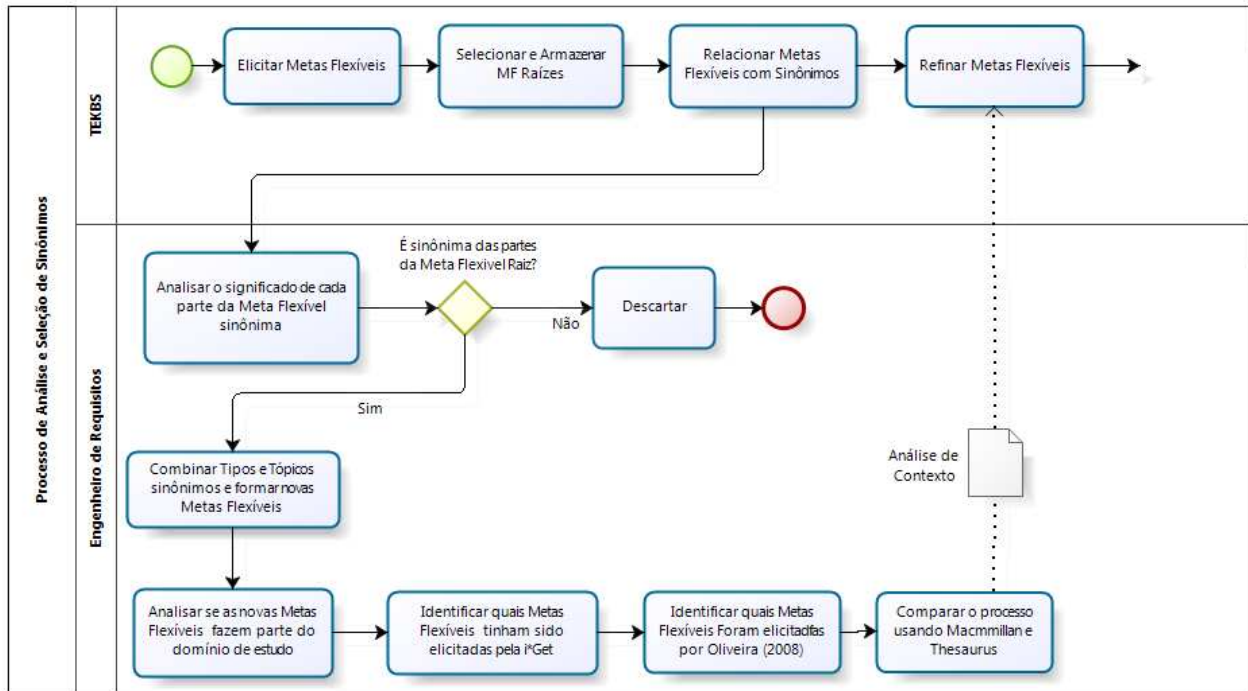


Figura 5.13: Processo de Análise e Seleção de Sinônimos.

5.5 Matriz de Influência das Metas Flexíveis

Um dos pontos fundamentais para a elaboração da MI foi a possibilidade de auxiliar o engenheiro de requisitos no processo de tomada de decisão e de acompanhamento das relações entre as MFs ao longo do ciclo de vida do software.

Baseada em parte do conceito da MR que possibilita a identificação e visualização do relacionamento de dependência entre os requisitos (Thommazo, Malimpensa, Oliveira, Olivatto e Fabbri, 2012) elaborou-se a MI.

Apesar de ter sido elaborada na fase de elicitação, sua utilização será mais efetiva nas fases posteriores com a evolução do processo de desenvolvimento do software que na grande maioria das vezes passam por modificações do projeto, causada principalmente pela mudança de requisitos.

As principais causas de modificações dos requisitos e consequentemente do projeto, estão ligadas as dificuldades encontradas na fase de elicitação de requisitos, ver subseção 2.1, e na dificuldade de controlar as alterações e os impactos que elas podem causar.

Assim, quando um requisito de MF precisar ser satisfeito, o engenheiro de requisitos saberá quais os possíveis impactos causados por ele. O mesmo

procedimento poderá ser percebido no processo de modificação, inclusão ou exclusão de alguma MF necessária para atender as necessidades das partes interessadas.

Para a elaboração da MI no contexto do EC, foram utilizadas somente as MF raízes, considerando um comportamento similar aos seus sinônimos. Essa consideração foi possível, porque percebeu-se que apesar dos sinônimos permitirem descobrir novas MFs, a sua grande maioria permanecia direcionada a uma funcionalidade específica. Um exemplo é a MF raiz “*correct [evaluation]*” que influencia positivamente a MF raiz “*quality [article]*”. Analisando alguns sinônimos da MF raiz “*correct [evaluation]*” como “*fair [valuation]*”, “*evenhandedly [valuation]*”, “*dependable [valuation]*” todos continuam contribuem positivamente com a qualidade do artigo.

Porém, ressalta-se que durante o processo de sinonímia, novas funcionalidades podem ser descobertas, como “*reliable [clause]*”, *true [inspection]* que apesar de não serem sinônimos diretos de “*correct [evaluation]*” também auxiliam a identificação da qualidade. Contudo ao analisar a relação entre sinônimos, novas influências podem surgir fornecendo ao engenheiro de requisitos um número maior de dados para auxiliar o processo de tomada de decisão.

A Figura 5.14 exhibe a MI de MFs raízes para o contexto do EC.

Matriz de Influência das Metas Flexíveis Raiz								
	correct [evaluation]	fair [review]	no delay [review]	acknowledge [committee]	quality [article]	honest [review]	punctuality [publication]	quality [review]
correct [evaluation]	✓				+	+		+
fair [review]	+	✓		+	+	+		+
no delay [review]			✓	+			+	
acknowledge [committee]			+	✓			+	
quality [article]					✓		+	
honest [review]				+	+	✓		
punctuality [publication]				+			✓	+
quality [review]					+			✓

Figura 5.14: Matriz de Influências de Metas Flexíveis Raiz no contexto do EC

Na Figura 5.14 as linhas são as que influenciam e as colunas as influenciadas, uma leitura da direita para a esquerda. A diagonal principal com as marcas vermelhas é considerada neutra, pois nelas a MF que influencia é a mesma MF que é influenciada. É possível notar que apesar da MF “*fair[review]*” influenciar outras MFs, ela não sofre influência de nenhuma neste contexto. No caso da *acknowledge[committee]* e *punctuality[publication]* além de influenciarem positivamente outras MFs são as que mais sofrem influência e ambas são influenciadas pela MF “*no delay[review]*”. Este tipo de visão auxilia o processo de tomada de decisão e acompanhamento quando o engenheiro de requisitos precisa movimentar, modificar, excluir ou incluir novas funcionalidades e conseqüentemente novas MFs. Neste caso, se o engenheiro de requisitos necessitar retirar a MF “*no delay[review]*” saberá que afetará diretamente o reconhecimento do comitê e a pontualidade da publicação.

5.6 Generalização do processo de TEKBS

O processo inicialmente descrito nesta dissertação, foi utilizando abordagens específicas, como o LAL e o ERI*c para estruturar as informações da UdI e o Wordnet para localização de sinônimos, contudo, as atividades da TEKBS podem ser realizadas com diversas UdIs, como por exemplo em XML⁷ (eXtensible Markup Language) e os sinônimos podem ser localizados utilizando qualquer dicionário de língua inglesa, como mostramos na Seção 5.4.

No processo geral, para elaborar a base de fatos ao invés de estruturar as informações do domínio que estão em linguagem natural para o LAL e dele para o CLIPS, basta estruturar as informações do XML direto para o CLIPS. Assim, toda e qualquer UdI que possa ser estruturada no formato CLIPS poderá ser utilizada como uma entrada para a TEKBS e dela extraídos as MFs raízes e as sinônimas.

Como forma de facilitar o entendimento geral do processo da TEKBS foram elaborados dois diagramas de atividades, o primeiro, ver Figura 5.15, exhibe o processo completo da TEKBS sem detalhamento e o segundo, ver Figura 5.16, ilustra o processo com seus respectivos módulos.

⁷ <http://www.w3schools.com/xml/>

A Figura 5.15 pode ser comparada com a Figura 5.13 que mostra o processo de Análise e seleção de sinônimos. O processo é dividido em duas raias, uma executada pela TEKBS e outra com auxílio do engenheiro de requisitos. Na Figura 5.15 além das modificações de entrada e da utilização dos dicionários, o processo realizado pelo engenheiro de requisitos fica mais enxuto, aqui não é preciso comparar os dados com uma base anterior, pois a base inicia-se na própria TEKBS e a comparação com outros dicionários só será realizada caso seja de interesse do engenheiro de requisitos, não sendo considerada uma atividade do processo executado pelo engenheiro de requisitos como é na Figura 5.13.

A Figura 5.16 aborda o mesmo conteúdo da Figura 5.15, porém de forma detalhada, nela é possível verificar como é a interação das atividades com seus respectivos módulos, a interação entre os módulos e deles com o engenheiro de requisitos, as decisões inicialmente tomadas junto ao processo e quais as possíveis finalizações das atividades.

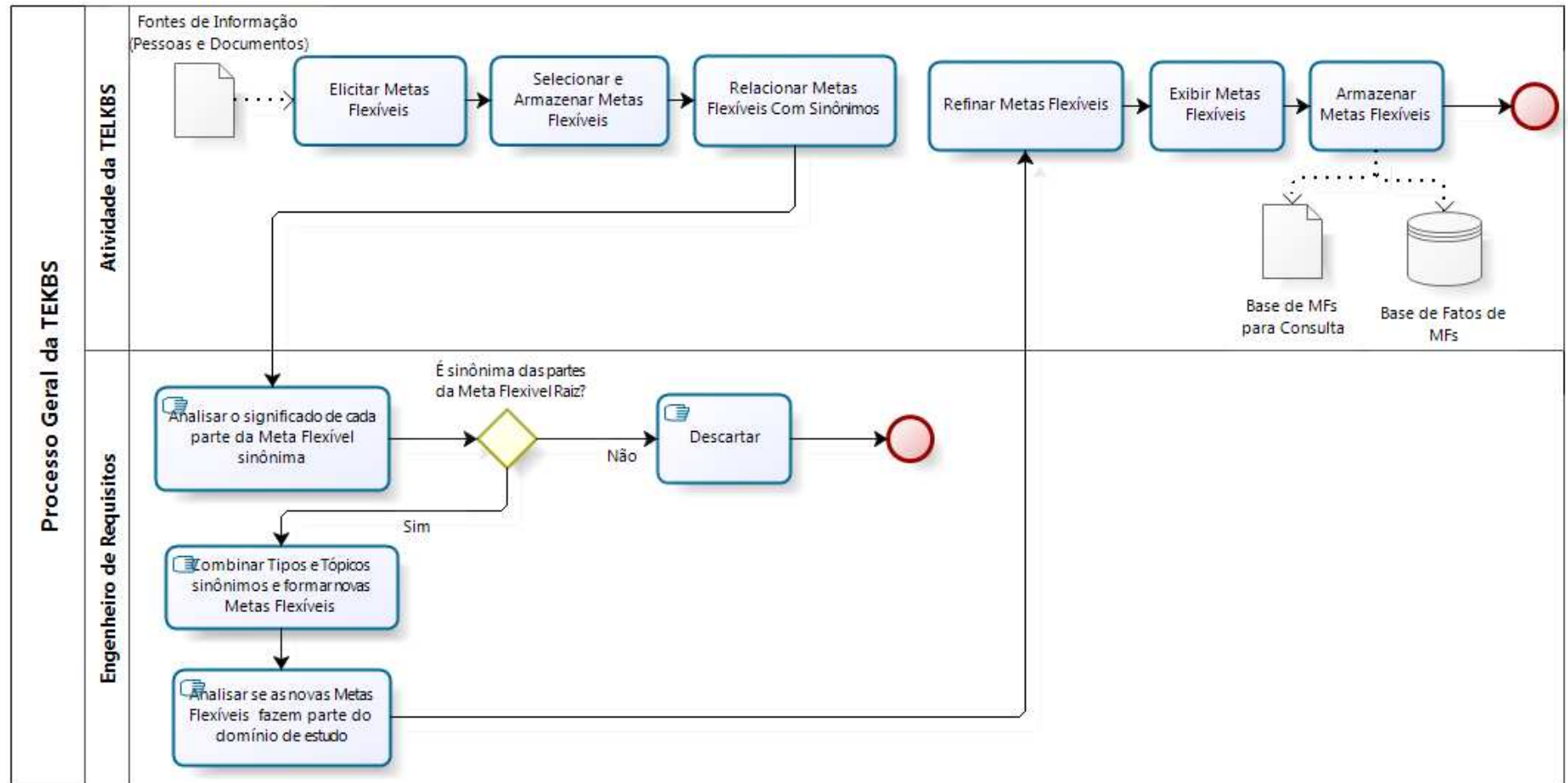


Figura 5.15: Processo da TEKBS generalizado

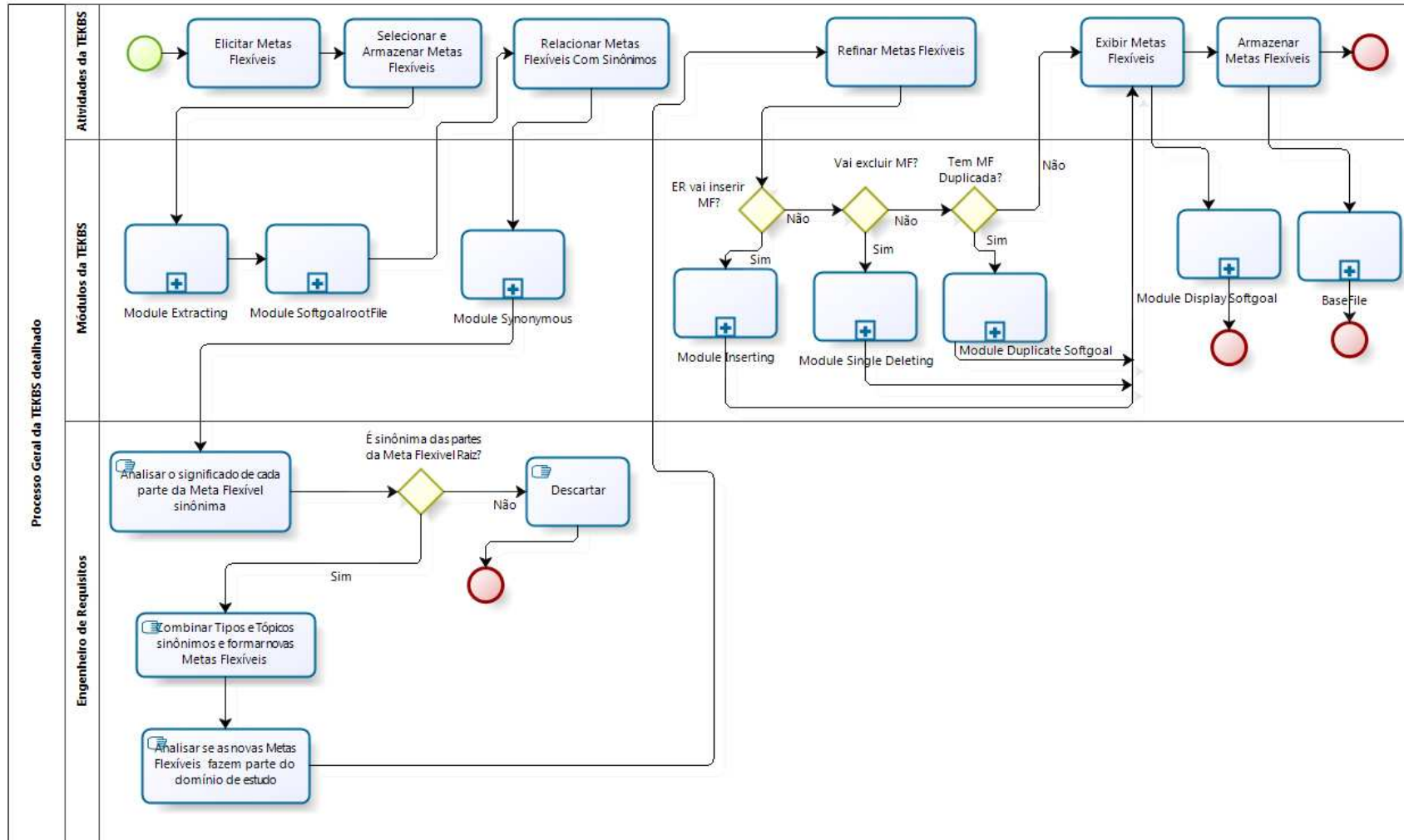


Figura 5.16: Processo generalizado da TEKBS detalhado.

Capítulo 6

6 CONSIDERAÇÕES FINAIS

O processo de elicitação é complexo e necessita da expertise do engenheiro de requisitos na busca de dados e informações nos diversos mecanismos utilizados para extrair requisitos. A utilização do LAL torna a elicitação do domínio mais próximo da realidade dos atores interessados e facilita o entendimento do engenheiro de requisitos.

A utilização da inteligência artificial no processo de elicitação de MFs por meio do CLIPS facilitou o processo de aprendizagem por ser composta de regras e fatos que auxiliam na solução de problemas que podem ser remodelados, além de permite expansão das bases com facilidade, separação de controle de conhecimento, conhecimento modular e flexibilidade no processo de modelagem.

O objetivo geral, projetar e implementar uma técnica que auxilie a elicitação de Metas Flexíveis e a verificação das influências mútuas entre as Metas Flexíveis na elaboração do modelo iStar, assim como os específicos: a) Identificar o acoplamento entre as Metas Flexíveis e seus sinônimos, que possam impactar as decisões dos engenheiros de requisitos na elaboração do modelo iStar; b) Elaborar uma rede (grafo) de colaborações e/ou de influências das Metas Flexíveis e seus sinônimos, alimentando-a com pesos do ambiente do problema, para que as simulações sejam o mais próximo do real quanto possível; e c) Testar a técnica por meio de casos de estudo, foram atendidos uma vez que a automatização parcial adotada neste trabalho ameniza as dificuldades do processo de elicitação de MFs, proporcionando ao engenheiro de requisitos mais segurança, a possibilidade de projetos melhor elaborados e de verificação das influências das MFs durante o processo de mudanças inerentes ao projeto de software. Estes procedimentos podem ser vistos no estudo de caso realizado, o Expert Committee – EC. Trabalhando com esse domínio foi possível observar que o número de MF elicitado utilizando a TEKBS foi superior ao número de MF elicitado pela ferramenta de (Cunha, 2014) e pela extração manual realizada em (Oliveira, 2008). Possuindo um número maior de MFs o engenheiro de requisitos

pode perceber um número maior de necessidades dos envolvidos, ou ainda refinar o entendimento de suas necessidades por possuir grande mapeamento do entendimento do problema.

Apesar de a TEKBS ter encontrado diferentes MFs na elicitação comparado ao processo manual feita por (Oliveira, 2008) e a automatizada realizada pela TEKBS, verificou-se que é possível unir várias MFs elicidadas pelas TEKBS para atender a grande maioria das MFs elicidadas por (Oliveira, 2008). Esse processo é possível considerando a diversidade de MFs que é proporcionada a TEKBS pela utilização da sinonímia.

No processo de verificação de acoplamento das MFs, foi possível perceber que o processo de elicitação utilizando o Wordnet é auspicioso e vem facilitar a fase de entendimento do problema. Apesar da pesquisa feita utilizando o dicionário de língua inglesa Thesaurus ter sido a que mais permitiu elicitar MFs sinônimas válidas para o contexto do EC.

Com os resultados aqui alcançados pode-se dizer que a Técnica TEKBS é adequada e efetiva no processo de elicitação de requisitos, sendo promissora sua utilização para auxiliar o engenheiro de requisitos não somente na elicitação, mas no entendimento do problema e na produção de um documento de requisitos mais claro e com mais qualidade.

Com a elaboração da MI, percebeu-se que apesar de não ser possível rastrear as MFs pela MI, possuir um mecanismo que permite a visualização imediata dos impactos facilita o processo de decisão, uma vez que torna possível conhecer quais MFs serão atingidas diante de alguma modificação.

Uma dificuldade da pesquisa foi encontrar trabalhos que apoiassem as atividades aqui desenvolvidas, apesar de existirem estudos que contemplem o processo de elicitação, sua grande maioria está voltada para a elicitação de RNFs e não de MFs. Dentro das consultas realizadas nenhum trabalho foi encontrado na mesma linha de pesquisa utilizando IA para auxiliar o processo de elicitação de MFs. O que foi possível localizar foi trabalhos sem correlação direta, como mencionado no decorrer do trabalho.

6.1 Principais contribuições

As principais contribuições trazidas por esta pesquisa foram:

- a) Auxiliar o engenheiro de requisitos no processo de eliciação de MFs;
- b) Possibilitar a extração de MFs não declaradas explicitamente via a utilização de sinônimos;
- c) Contribuir para que um número maior de MFs sejam elicítadas além de minimizar as omissões, permitindo a elaboração de um documento de requisitos mais completo e mais próximo da realidade do problema; e
- d) Fornecer um mecanismo que facilite a análise de influências entre MFs.

6.2 Trabalhos futuros

A seguir, são apresentadas algumas sugestões de trabalhos futuros:

- a) **Criação do diagrama de Painel de Intencionalidade - IP.** Segundo Oliveira (Oliveira, Antônio Padua Albuquerque, Leite, J.C.S. e Cysneiros, Luiz Marcio, 2008) o IP auxilia na representação das MFs, identifica as relações entre as MF por meio de dependências, correlações, contribuições e equivalências em um único diagrama, sua elaboração facilitará o entendimento das relações e das influências entre as MFs, MCs e atores envolvidos o que poderá possibilitar um ganho na qualidade nos modelos.
- b) **Unificação do processo de (Cunha, 2014) com a TEKBS.** Sabe-se que o escopo de um domínio não é estático e que alterações podem ocorrer ao longo de todo o processo de eliciação e desenvolvimento. Com a alteração do escopo as MFs raízes serão modificadas e um novo processo de eliciação deve ser iniciado, considerando a necessidade de reestruturar a descrição do LAL alinhando-o ao novo escopo. Após o alinhamento, o processo de (Cunha, 2014) deverá ser novamente executado para que novas MFs raízes possam ser elicítadas executando novamente a técnica TEKBS para extrair novas MFs raízes e seus sinônimos.

- c) **Verificação semântica** - Criar mecanismos automáticos capazes de verificar a incompatibilidade semântica entre as MFs raízes e seus sinônimos. A abordagem linguística alinhada ao contexto pode evitar a inserção de dados não precisos. Atualmente essa atividade é feita mediante interação com o engenheiro de requisitos, que é o responsável pela validação dos sinônimos das MFs elicitadas. A validação neste caso está condicionada ao conhecimento do engenheiro.
- d) **Experimentação**. Apesar de ter sido testada e comprovada a viabilidade da TEKBS, acredita-se que a experimentação com casos reais, traga ganhos: no entendimento do problema de eliciação de MF, na verificação de dificuldades e de facilidades encontradas pelos engenheiros na utilização da TEKBS, como também para o refinamento da própria técnica.
- e) **Rastreabilidade e MI**. Refinar a MI das MFs raízes e ampliar para os sinônimos inserindo mecanismos que possibilite rastreá-las, oferecendo mais informação para o gerente de requisitos e permitindo conhecer mais a fundo os impactos causados na inclusão, exclusão e modificação de MFs.

7 REFERÊNCIAS BIBLIOGRÁFICAS

ALSHAZLY, A. A.; ELFATATRY, A. M.; ABOUGABAL, M. S. Detecting defects in software requirements specification. **Alexandria Engineering Journal**, 2014.

AMARAL, E. A. G. G. **Empacotamento de experimentos em engenharia de software**. 2003. 164 Dissertação (Master). Programa de Engenharia de Sistemas e Computação, COPPE/UFRJ, Universidade Federal do Rio de Janeiro. , Rio de Janeiro, RJ, Brasil.

ANTONELLI, L.; ROSSI, G.; LEITE, J. C. S.; OLIVEROS, A. **Buenas prácticas en la especificación del dominio de una aplicación**. In: FERNÁNDEZ, N. C. e ANTONELLI, L., Workshop em Engenharia de Requisitos - WER, 2013. p.1-14.

ARAÚJO, B. O.; LANA, C. A.; PAULA, J. N. C.; MEIRA, A. D.; VIDIGAL, C. C. **Oficinas vocacionais, redes sociais e educação à distância na inclusão social e profissional de jovens viçosenses.**, VI Encontro Mineiro de Engenharia de Produção - EMEPRO 2010, 2010, Coronel Fabriciano. p.1-11.

BAÍA, J. W.; BRAGA, J. L. **Uso de Sinônimos na Identificação de Atributos de Transparência**. Workshop em Engenharia de Requisitos 2013, Montevideo. p.94-104.

BEATTY, J.; WIEGERS, K. **Software Requirements**. Pearson Education, 2013.

BHALALUSESA, R. P.; ARSHAD, M. R. M. **Using Search Results Metadata to Discover Effective Learning Objects for Mobile Devices**. Proceedings of the First International Conference on Advanced Data and Information Engineering (DaEng-2013), 2014, Springer. p.605-612.

BOEHM, B. W. Software engineering economics. In: (Ed.). **Pioneers and Their Contributions to Software Engineering**: Springer, 2001. p.99-150.

BROOKS JR., F. P. **Bullet, No Silver: Essence and Accidents of Software Engineering**. **IEEE Computer Apr 1987**, v. 20, n. 4, p. 10-19, 1987.

CAMPOS, J. P. **Extração de candidatos a aspectos a partir de descrições de fluxo de casos de uso**. 2009. 103f. (Mestre). Departamento de Informática, Universidade Federal de Viçosa, 2009.

CAPERS, J. **Applied Software Measurement: Assuring Productivity and Quality**. **McGraw-Hill, New York**, v. 17, n. 1, p. 2, 1996.

CARES, C. **From the i* Diversity to a Common Interoperability Framework**. 2012. PhD Thesis, Polytechnic University of Barcelona, Spain

CARRIZO, D.; DIESTE, O.; JURISTO, N. Systematizing requirements elicitation technique selection. **Information and Software Technology**, v. 56, n. 6, p. 644-669, 2014.

CARVALHO, L. C. Análise de sistemas: o outro lado da informática. **Rio de Janeiro**, 1988.

CASTAÑEDA, V.; BALLEJOS, L. C.; CALIUSCO, M. L. **Improving the Quality of Software Requirements Specifications with Semantic Web Technologies**. Workshop em Engenharia de Requisitos -WER, 2012. p.1-14.

CHAKRABORTY, A.; BAOWALY, M. K.; AREFIN, A.; BAHAR, A. N. The Role of requirements engineering in software development life cycle. **Jornal of Emergi Trend in Computing and Information Sciences**, v. 3, n. 5, p. 723-729, 2012.

CHARETTE, R. N. Why software fails. **IEEE spectrum**, v. 42, n. 9, p. 36, 2005.

CHIZZOTTI, A. A pesquisa qualitativa em ciências humanas e sociais: evolução e desafios. **Revista Portuguesa de Educação**, v. 16, n. 2, p. 221-236, 2003.

CHRISTEL, M. G.; KANG, K. C. **Issues in requirements elicitation**. DTIC Document. 1992

CHUNG, L.; NIXON, B.; YU, E.; MYLOPOULOS, J. Non-functional Requirements in Software Engineering. **Kluwer Academic Publishers**, 2000.

CHUNG, L.; NIXON, B. A. **Dealing with non-functional requirements: three experimental studies of a process-oriented approach**. Software Engineering, 1995. ICSE 1995. 17th International Conference on, 1995, IEEE. p.25-25.

COELHO, R.; CIRILO, E.; KULESZA, U.; VON STAA, A.; RASHID, A.; LUCENA, C. **Jat: A test automation framework for multi-agent systems**. Software Maintenance, 2007. ICSM 2007. IEEE International Conference on, 2007, IEEE. p.425-434.

COLOMER, D.; LÓPEZ, L.; CARES, C.; FRANCH, X. **Model Interchange and Tool Interoperability in the i* Framework: A Proof of Concept**. In: MARIA LENCASTRE, H. E. E., EDUARDO FIGUEIREDO, Workshop em Engenharia de Requisitos - WER 2011, 2011, Rio de Janeiro. p. 369-382.

CUNHA, L. G. **Mapeamento de especificações LAL - Léxico Ampliado de Linguagem em modelos intencionais**. 2014. 102f. (Dissertação). Departamento de Informática, Universidade Federal de Viçosa, 2014.

CYSNEIROS, L.; YU, E.; LEITE, J. C. S. P. **Cataloguing non-functional requirements as softgoal networks**. Proc. of requirements engineering for adaptable architectures. 11th international requirements engineering conference, 2003. p.13-20.

CYSNEIROS, L. M. **Requisitos não funcionais: da elicitação ao modelo conceitual**. 2001. 224 f. (Doutor). Departamento de Informática, Pontifícia Universidade Católica do Rio de Janeiro, Rio de Janeiro, 2001.

_____. **Using i* to Elicit and Model Transparency in the Presence of Other Non-Functional Requirements: A Position Paper**. In: JAELSON CASTRO, J. H., NEIL MAIDEN, ERIC YU, 6th International i* Workshop (iStar 2013), 2013, Valencia, Spain. p.19-24.

DAVIS, A. M. **Software requirements: objects, functions, and states**. Prentice-Hall, , 1993.

DELOACH, S. A.; WOOD, M. F.; SPARKMAN, C. H. Multiagent systems engineering. **International Journal of Software Engineering and Knowledge Engineering**, v. 11, n. 03, p. 231-258, 2001.

DURKIN JACK; DURKIN JOHN. **Expert systems: design and development**. Prentice Hall PTR, 1998.

ERNST, N.; BORGIDA, A.; JURETA, I. J.; MYLOPOULOS, J. An Overview of Requirements Evolution. In: (Ed.). **Evolving Software Systems**: Springer, 2014. p.3-32.

FARINHA, C.; SILVA, M. M. **Requirements Elicitation With Focus Groups: Lessons Learnt**. Proceedings of the 21st European Conference on Information Systems, 2013. p.1-12.

FELICÍSSIMO, C. H.; LEITE, J.; BREITMAN, K.; SILVA, L. **C&L: Um Ambiente para Edição e Visualização de Cenários e Léxicos**. XVIII Simpósio Brasileiro de Engenharia de Software - SBES, 2004. p.43-48.

FELLBAUM, C. **WordNet: an eletronic lexical database**. Cambridge, MA: MIT Press, 1998.

FRANCH, X. **The i* framework: The way ahead**. Sixth International Conference on Research Challenges in Information Science (RCIS), 2012 2012, Paris, France. IEEE. p.1-3.

FRIEDMAN-HILL, E. **JESS in Action**. Manning Greenwich, CT, 2003. ISBN 1930110898.

_____. JESS, the rule engine for the java TM plataform. Estados Unidos, 2013. Disponível em: < <http://www.jessrules.com/jess/index.shtml> >. Acesso em: 22/06/2014.

FRIEDMAN, E. **Jess in action: rule-based systems in java**. 2003. ISSN 1930110898.

GARCIA, A. F. **From Objects to Agents: an Aspect Oriented Approach**. 2004. 298 f. (Doutor). Departamento de Informática, Pontificia Universidade Católica do Rio de Janeiro, Rio de Janeiro, 2004.

GIARRATANO, J. C. CLIPS User's Guide. NASA, Lyndon B. Johnson Space Center. **Information Systems Directorate, Software Technology Branch, Houston, TX**, 1993.

GIL, A. C. **Como Elaborar Projetos de Pesquisa**. 5 ed. São Paulo: Atlas, 2010.

GODOY, A. S. Introdução à pesquisa qualitativa e suas possibilidades. **Revista de administração de empresas**, v. 35, n. 2, p. 57-63, 1995.

GOGUEN, J. A.; LINDE, C. **Techniques for requirements elicitation**. Requirements Engineering, 1993., Proceedings of IEEE International Symposium on, 1993, IEEE. p.152-164.

HEDLEY, J. jsoup: Java html parser. 2009. Disponível em: < <http://jsoup.org/> >. Acesso em: 18/06/2014.

HOFMANN, H. F.; LEHNER, F. Requirements engineering as a success factor in software projects. **IEEE software**, v. 18, n. 4, p. 58-66, 2001.

IEEE, C. S. S. E. S. C. **Recommended Practice for Software Requirements Specifications**. IEEE Std 830-1998, 1998.

JUN, L.; QIUZHEN, W.; QINGGUO, M. The effects of project uncertainty and risk management on IS development project performance: A vendor perspective. **International Journal of Project Management**, v. 29, n. 7, p. 923-933, 2011.

JURETA, I. J.; FAULKNER, S.; SCHOBENS, P.-Y. A more expressive softgoal conceptualization for quality requirements analysis. In: (Ed.). **Conceptual Modeling-ER 2006**: Springer, 2006. p.281-295.

KAMPS, J.; MARX, M.; MOKKEN, R. J.; DE RIJKE, M. Using wordnet to measure semantic orientations of adjectives. 2004.

KAUR, J. Word Sense Disambiguation (WSD). **International Journal For Technological Research In Engineering**, v. 1, n. 5, p. 256 - 259, 2014.

KOKKALIS, A. **Extracting and Integrating Meta-data from online sources**. 2014. (Doctor). Department of Communication Systems, KTH Royal, 2014.

LEITE, J. C. S.; OLIVEIRA, A. P. A. **A client oriented requirements baseline**. Requirements Engineering, 1995., Proceedings of the Second IEEE International Symposium on, 1995, IEEE. p.108-115.

LEITE, J. C. S. D. P. **Viewpoint resolution in requirements elicitation**. 1988. 170f. (Doctor). Computer Science, University of California, Irvine, 1988.

LEITE, J. D. P.; FRANCO, A. **A Client Strategy for Conceptual Model Acquisition**. Proceedings of the International Symposium on Requirements Engineering. IEEE Computer Society Press, 1993, IEEE. p.243-246.

LIU, Y.; SCHEUERMANN, P.; LI, X.; ZHU, X. Using wordnet to disambiguate word senses for text classification. In: (Ed.). **Computational Science–ICCS 2007**: Springer, 2007. p.781-789.

LOBATO, C.; GARCIA, A.; ROMANOVSKY, A.; LUCENA, C. An aspect-oriented software architecture for code mobility. **Software: Practice and Experience**, v. 38, n. 13, p. 1365-1392, 2008.

LUCENA, M.; SILVA, C. T.; SANTOS, E.; ALENCAR, F. M.; CASTRO, J. **Modularizando Modelos i*: uma Abordagem baseada em Transformação de Modelos**. WER, 2009.

MACMILLAN, D. Macmillan Dictionary. 2014. Disponível em: < <http://www.macmillandictionary.com/> >. Acesso em: 22/06/2014.

MATSUOKA, J.; LEPAGE, Y. **Ambiguity spotting using WordNet semantic similarity in support to recommended practice for software requirements specifications**. Natural Language Processing and Knowledge Engineering (NLP-KE), 2011 7th International Conference on, 2011, IEEE. p.479-484.

MCCABE, T. J. A complexity measure. **Software Engineering, IEEE Transactions on**, n. 4, p. 308-320, 1976.

MICHAELIS, H. **Moderno dicionário inglês: inglês-português, português-inglês**. Melhoramentos, 2010.

MILLER, G. WordNet: a lexical database for English. **Communications of the ACM**, v. 38, n. 11, p. 39-41, 1995.

MONSALVE, E. S.; LEITE, J. C. S. D. P. **Using i* for Transparent Pedagogy**. In: JAELOSON CASTRO, J. H., NEIL MAIDEN, ERIC YU, 6th International i* Workshop 2013, 2013, Valencia, Spain. p.25-30.

MYLOPOULOS, J.; CHUNG, L.; NIXON, B. Representing and using nonfunctional requirements: A process-oriented approach. **Software Engineering, IEEE Transactions on**, v. 18, n. 6, p. 483-497, 1992.

NAPOLITANO, F. **Uma Estratégia Baseada em Simulação para Validação de Modelos em i***. 2009. 165f (Masters). Departamento de Informática, PUC–Rio, Dissertação de Mestrado 2009.

NEGNEVITSKY, M. **Artificial intelligence: a guide to intelligent systems**. Pearson Education, 2005.

NUNES, C.; KULESZA, U.; SANT'ANNA, C.; NUNES, I.; GARCIA, A. F.; DE LUCENA, C. J. P. Assessment of the Design Modularity and Stability of Multi-Agent System Product Lines. **J. UCS**, v. 15, n. 11, p. 2254-2283, 2009.

NUSEIBEH, B.; EASTERBROOK, S. **Requirements engineering: a roadmap**. Proceedings of the Conference on the Future of Software Engineering, 2000, ACM. p.35-46.

OLIVEIRA, A. P. A. **Engenharia de Requisitos Intencional: Um Método de Elicitação, Modelagem e Análise de Requisitos**. 2008. Departamento de Informática, PUC-Rio, Tese de Doutorado, PUC-Rio, 2008.

OLIVEIRA, A. P. A.; BRAGA, J. L.; LANA, C. A.; CUNHA, L. G. **Utilizando sistemas de conhecimento para a identificação de metas flexíveis em uma linguagem de domínio**. Requirements Engineering@Brazil, 2013. p.1-6.

OLIVEIRA, A. P. A.; CYSNEIROS, L. M. **Defining strategic dependency situations in requirements elicitation**. The IX Workshop on Requirements Engineering, 2006, Rio de Janeiro. p.1-12.

OLIVEIRA, A. P. A.; LEITE, J. C. S. **The Experience of Using ERI* c in a Telecom Corporation**. In: DALPIAZ, F. e HORKOFF, J., Seventh International i* Workshop, 2014.

OLIVEIRA, A. P. A.; LEITE, J. C. S.; CYSNEIROS, L. M. **AGFL-Agent Goals from Lexicon-Eliciting Multi-Agent Systems Intentionality**. iStar, 2008, Citeseer. p.29-32.

OLIVEIRA, A. P. A.; LEITE, J. C. S.; CYSNEIROS, L. M. **Método ERI* c-Engenharia de Requisitos Intencional**. Workshop em Engenharia de Requisitos - WER 2008. p.155-166.

PA, N. C.; ZIN, A. M. Requirement Elicitation: Identifying the Communication Challenges between Developer and Customer. **International Journal of New Computer Architectures and their Applications (IJNCAA)**, v. 1, n. 2, p. 371-383, 2011.

PACHECO, C.; GARCIA, I. A systematic literature review of stakeholder identification methods in requirements elicitation. **Journal of Systems and Software**, v. 85, n. 9, p. 2171-2181, 2012.

PIERITZ, H. I. **Spirit: uso como jogador em jogos de empresas**. . 2003. 277f (Doutorado). Departamento de engenharia de produção e sistemas, Universidade Federal de Santa Catarina, 2003.

POHL, K. **Requirements engineering: An overview**. RWTH, Fachgruppe Informatik, 1997.

RAJAGOPAL, P.; LEE, R.; AHLWEDE, T.; CHIANG, C.-C.; KAROLAK, D. **A new approach for software requirements elicitation**. Sixth International Conference on Software Engineering, Artificial Intelligence, Networking and Parallel/Distributed Computing and First ACIS International Workshop on Self-Assembling Wireless Networks (SNPD/SAWN), 2005, IEEE. p.32-42.

RAWLINS, M. C.; CHUNG, L. In Pursuit of Better EDI: An Introduction to the Use of Non-Functional Requirements in Designing EDI Standards and Architectures. **for X12 Strategic Implementation Task Group and UN EDI Standards Committee**, 1999.

REED, K. Software Engineering-A New Millennium? **IEEE Software**, v. 17, n. 4, p. 107-107, 2000.

RICHARDSON, R.; SMEATON, A.; MURPHY, J. **Using WordNet as a knowledge base for measuring semantic similarity between words**. Technical Report Working Paper CA-1294, . School of Computer Applications, Dublin City University. 1994

RILEY, G. A Tool for Building Expert Systems. 2013. Disponível em: < <http://clipsrules.sourceforge.net/> >. Acesso em: 25/03/2014.

SAYÃO, M.; LEITE, J. C. S. D. P. Rastreabilidade de requisitos. **RITA**, v. 12, n. 1, p. 57-86, 2005.

SEN, A.; JAIN, S. **An Agile Technique for Agent Based Goal Refinement to Elicit Soft Goals in Goal Oriented Requirements engineering**. Advanced Computing and Communications, 2007. ADCOM 2007. International Conference on, 2007, IEEE. p.41-47.

SERRANO, M. **Desenvolvimento Intencional de Software Transparente Baseado em Argumentação**. 2011. 282 f. (Doutor). Departamento de Informática Pontifícia Universidade Católica do Rio de Janeiro, Rio de Janeiro, 2011.

SHAMS-UL-ARIF, M.; KHAN, M. Q.; GAHYUR, S. Requirements Engineering Processes, Tools/Technologies, & Methodologies. **International Journal of Reviews in computing**, p. 41-56, 2009.

SILVA, C.; BORBA, C.; CASTRO, J. **A Goal Oriented Approach to Identify and Configure Feature Models for Software Product Lines**. In: MARIA LENCASTRE, H. E. E., EDUARDO FIGUEIREDO, Workshop em Engenharia de Requisitos - WER 2011, 2011, Rio de Janeiro. p.395 - 406.

SILVA, L. F. **Uma Estratégia Orientada a Aspectos para a Modelagem de Requisitos**. 2006. 222f (Doutorado). Informática, Pontifícia Universidade Católica do Rio de Janeiro, 2006.

SILVA, V. T.; NOYA, R. C.; LUCENA, C. J. P. **Using the UML 2.0 activity diagram to model agent plans and actions**. Proceedings of the fourth international joint conference on Autonomous agents and multiagent systems, 2005, ACM. p.594-600.

SINTA, G. Sistemas Inteligentes Aplicados, Expert Sinta V. 1.1 Manual do usuário. **Universidade Federal do Ceará**, 1996.

SOMMERVILLE, I. **Software Engineering**. 9. Addison-Wesley, 2011.

SPERANDEI, H.; OLIVEIRA, A. P. A.; LEITE, J. C. S.; WERNECK, V. **Modelagem do Processo de Aprovação de Investimentos de uma Grande Empresa com ERI*c**. 2010. (Graduação). Instituto de Matemática e Estatística, Universidade do Estado do Rio de Janeiro, 2010.

SPIRANDELLI, L. P.; SANTOS, G. H.; RODRIGUES, L.; BANDOS, M. F. C. Sistemas Especialistas: Um estudo de caso com o Expert Sinta. **Revista Eletrônica de Sistemas de Informação e de Gestão Tecnológica**, v. 1, n. 1, 2011. ISSN 2237-0072.

SPIRIT. Probabilistisches Schließen in Wissensbasierten Systemen. Fakultät, 2014. Disponível em: < <http://www.xspirit.de/> >. Acesso em: 22/06/2014.

SUTCLIFFE, A.; SAWYER, P. **Requirements elicitation: Towards the unknown unknowns**. Requirements Engineering Conference (RE), 2013 21st IEEE International, 2013, IEEE. p.92-104.

THESAURUS. Thesaurus.com. 2014. Disponível em: < <http://thesaurus.com/> >. Acesso em: 22/06/2014.

THOMMAZO, A. D.; MALIMPENSA, G.; OLIVEIRA, T. R.; OLIVATTO, G.; FABBRI, S. C. P. F. **Requirements Traceability Matrix: Automatic Generation and Visualization**. Software Engineering (SBES), 2012 26th Brazilian Symposium on, 2012, IEEE. p.101-110.

TORRES, K.; BRAGA, J. Construtor de Ontologias Baseado no WordNet. **UFV–Universidade Federal de Viçosa**, 2002.

TRONDAL, J.; PETERS, B. G. The rise of European administrative space: lessons learned. **Journal of European Public Policy**, v. 20, n. 2, p. 295-307, 2013.

VINAY, S.; AITHAL, S.; SUDHAKARA, G. Effect of Contribution Links on Choosing Hard Goals in GORE Using AHP and TOPSIS. In: (Ed.). **Emerging Research in Electronics, Computer Science and Technology**: Springer, 2014. p.737-745.

WASSERMAN, A. I. Toward a discipline of software engineering. **IEEE software**, v. 13, n. 6, p. 23-31, 1996.

WAZLAWICK, R. S. Uma reflexão sobre a pesquisa em ciência da computação à luz da classificação das ciências e do método científico. **Revista de Sistemas de Informação da FSMA**, n. 6, p. 3-10, 2010.

WOHLIN, C.; RUNESON, P.; HOST, M.; REGNELL, B.; WESSLÉN, A. **Experimentation in software engineering: An Introduction**. . The Kluwer International Series in Software Engineering, Norwel - USA: Kluwer Academic Publishers, 2000.

WORDNET. Wordnet Search - 3.1. 2014. Disponível em: <
<http://wordnetweb.princeton.edu/perl/webwn>>. Acesso em: 22/06/2014.

XAVIER, L.; ALENCAR, F. M. R.; CASTRO, J.; PIMENTEL, J. **Integração de Requisitos Não-Funcionais a Processos de Negócio: Integrando BPMN and NFR**. Workshop em Engenharia de Requisitos - WER 2010. p.29-40.

XU, Z.; YAN, D. **Designing and Implementing of the Webpage Information Extracting Model Based on Tags**. Intelligence Science and Information Engineering (ISIE), 2011 International Conference on, 2011, IEEE. p.273-275.

YU, E.; LIU, L.; LI, Y. Modelling strategic actor relationships to support intellectual property management. In: (Ed.). **Conceptual Modeling—ER 2001**: Springer, 2001. p.164-178.

YU, E. S. K. **Modelling strategic relationships for process reengineering**. 1995. 131 (Phd). Computer Science, University of Toronto, 1995.

YU , E. S. K. Social Modeling and i*. In: (Ed.). **Conceptual Modeling: Foundations and Applications**: Springer, 2009. p.99-121.

ZOWGHI, D.; COULIN, C. Requirements elicitation: A survey of techniques, approaches, and tools. In: (Ed.). **Engineering and managing software requirements**: Springer, 2005. p.19-46.

8 APÊNDICE A. BASE DE REGRAS

**** KNOWLEDGE-BASED SOFTGOAL ELICITATION TECHNIQUE****

;;#####

```
;;This program is responsible for softgoals elicitar and its synonyms
;;code function created by Cristiane Aparecida Lana
;;last updated on 05/14/2014
;;Master's degree at the Federal University of Viçosa
```

;;#####

```
..*****
,
*****
```

;; Templates

```
..*****
,
*****
```

```
(defmodule MAIN (export ?ALL))
```

```
(deftemplate elementLel "The LEL's element"
(slot id (type SYMBOL) )
(slot name (type STRING) )
(slot type (allowed-symbols subject verb state object) )
)
```

```
(deftemplate behavioralResponse "The behavioral response of the LEL element"
(slot bhId (type SYMBOL) )
(slot bhLelElementName (type STRING) )
(slot bhDescription (type STRING) )
(slot bhType (type SYMBOL) (allowed-symbols hardgoal softgoal) (default
hardgoal))
(slot bhProposedType (type SYMBOL) (allowed-symbols hardgoal softgoal null)
(default null))
(slot bhCertainFactor (type FLOAT) (default 0.0))
(slot bhActionId (type SYMBOL) (default null))
)
```

```
(deftemplate actorElement "Actor element"
(slot actorId (type SYMBOL) )
(slot actorName (type STRING) )
)
```

```
(deftemplate goalElement "Goal element"
(slot goalId (type SYMBOL) )
(slot goalElementName (type STRING) )
(slot goalSubjectObject (type STRING) )
(slot goalVerb (type STRING) )
```

```
(slot goalSubject (type STRING) )
(slot goalBhId (type SYMBOL) )
)
```

```
(deftemplate softgoalElement "Softgoal element"
(slot sgId (type SYMBOL) )
(slot sgQualityAttribute (type STRING) )
(slot sgSubjectObject (type STRING) )
(slot sgGoalId (type SYMBOL) )
(slot sgActorName (type STRING) )
(slot sgBhId (type SYMBOL) )
)
```

```
(deftemplate action "The action present in the behavioral responses"
(slot actionId (type SYMBOL) )
(slot actionStem (type STRING) (default "null"))
(slot actionVerb (type STRING) )
(slot actionType (type SYMBOL) )
(slot actionSuccess (type INTEGER) (default 1) )
(slot actionOccurrence (type INTEGER) (default 2) )
)
```

```
(deftemplate basesoftgoal "basesoftgoal receives softgoals extracted facts base of
(Cunha, 2014)"
(slot sgId (type SYMBOL) )
(slot sgQualityAttribute (type STRING) )
(slot sgSubjectObject (type STRING) )
)
```

```
(deftemplate synonymoustype "Gets the Type of softgoal"
(slot type (type STRING) )
(slot syntype (type STRING) )
)
```

```
(deftemplate synonymoustopic "Gets the topic of softgoal"
(slot topic (type STRING) )
(slot syntopic (type STRING) )
)
```

```
(deftemplate synSoftgoal "synsoftgoal receives synonyms for softgoals extracted
from the base facts of (Cunha, 2014)"
(slot ttId (type SYMBOL) )
(slot synTopic (type STRING) )
(slot synType (type STRING) )
)
```

```

..*****
;,
*****
;; MAIN MODULE
..*****
;,
*****

```

```

(deffacts MAIN::init
(menu-level module main)
)

```

```

(defrule main-menu
?Mm <- (menu-level module main)
=>
(retract ?Mm)
(printout t crlf crlf)
(printout t
"Choose the Module" crlf
"by typing a number and pressing the return key" crlf crlf
" 1) Module Extracting. " crlf
" 2) Module SoftgoalrootFile " crlf
" 3) Module Synonymous. " crlf
" 4) Module Inserting " crlf
" 5) Module Display Softgoals. " crlf
" 6) Module Single Deleting " crlf
" 7) Module Duplicate Softgoal. " crlf
" 8) Module BaseFile " crlf
" 9) Module Exit. " crlf
>Your Choice: ")

```

```

(bind ?answer (read))
(assert (module choice ?answer))
(printout t crlf crlf crlf)
)

```

```

.....MAIN Rules.....

```

```

(defrule ruleModuleChoiceExtracting
(module choice 1)
=>
(printout t "*** You chose Module Extracting ***" crlf)
(focus Extracting)
)

```

```

(defrule ruleSoftgoalrootFile
(module choice 2)
=>
(printout t "*** You chose Module SoftgoalrootFile ***")
(focus SoftgoalrootFile)
)

```

```
(defrule ruleModuleChoiceSynonymy
(module choice 3)
=>
(printout t "*** You chose Module Synonymy ***"crLf)
(focus Synonymous)
)
```

```
(defrule ruleModuleChoiceInserting
(module choice 4)
=>
(printout t "*** You chose Module Inserting ***"crLf)
(focus Inserting)
)
```

```
(defrule ruleModuleChoiceDisplaySoftgoal
(module choice 5)
=>
(printout t "*** You chose Module DisplaySoftgoal ***"crLf)
(focus DisplaySoftgoal)
)
```

```
(defrule ruleModuleChoiceSingleDelete
(module choice 6)
=>
(printout t "*** You chose Module Single Deleting ***"crLf)
(focus SingleDeleting)
)
```

```
(defrule ruleModuleChoiceDuplicateSoftgoal
(module choice 7)
=>
(printout t "*** You chose Module Deletes Duplicate Softgoal ***"crLf)
(focus DuplicateSoftgoal)
)
```

```
(defrule ruleModuleChoiceBaseFile
(module choice 8)
=>
(printout t "*** You chose Module BaseFile ***"crLf)
(focus BaseFile)
)
```

```
(defrule ruleModuleChoiceQuit
(module choice 9)
=>
(printout t "*** You chose Module Exit"crLf)
(halt)
)
```

```

..*****
,;
*****
;; Extracting Module
..*****
,;
*****

```

```

(defmodule Extracting
(import MAIN ?ALL)
)

```

```

.....; Rules Extract Softgoal ;.....

```

```

(defrule Extracting::ruleAssertsoftgoalroot "This rule inserts softgoals in the fact
basis."

```

```

(declare (salience 95))
(softgoalElement
(sgId ?idSg )
(sgQualityAttribute ?QualityAttributeSg )
(sgSubjectObject ?SubjectObjectSg )
(sgGoalId ?GoalIdSg )
(sgActorName ?ActorNameSg ))
=>
(assert (basesoftgoal (sgId(gensym))(sgQualityAttribute
?QualityAttributeSg)(sgSubjectObject ?SubjectObjectSg)))
)

```

```

(defrule Extracting::rulePrintmessageInserting "Print informational message."

```

```

(declare (salience 0))
=>
(printout t crlf "Data successfully extracted" crlf)
)

```

```

(defrule Extracting::ruleChanceModule " Change focus to Main module"

```

```

(declare (salience 0))
?fact <-(module choice 1)
=>
(assert (menu-level module main))
(retract ?fact)
)

```

```

..*****
,;
*****
;; SoftgoalrootFile Module
..*****
,;
*****

```

```

(defmodule SoftgoalrootFile
(import MAIN ?ALL)
)

```

```

:;;;;; Softgoal root File Softgoal ;;;;;;

```

```

(defrule SoftgoalrootFile::DeleteSoftgoal
"This rule deletes duplicate root softgoal found in the facts base."
(declare (salience 98))
(basesoftgoal
(sgId ?sgId1)
(sgSubjectObject ?SubjectObjectSg1)
(sgQualityAttribute ?QualityAttributeSg1))

?fact2 <- (basesoftgoal
(sgId ?sgId2)
(sgSubjectObject ?SubjectObjectSg2)
(sgQualityAttribute ?QualityAttributeSg2))
(test (and (neq ?sgId1 ?sgId2)
(eq ?SubjectObjectSg2 ?SubjectObjectSg1)
(eq ?QualityAttributeSg2 ?QualityAttributeSg1)))
=>
(retract ?fact2)

)

```

```

(defrule SoftgoalrootFile::ruleOpensofgoalBaseFile "This rule is responsible for
creating the file containing softgoals"
(declare (salience 90))
=>
(open "softgoalFile.clp" baseFile "w")
(printout t crlf "[Opened file softgoalFile.clp.] " crlf )
(assert (startWriteInFile))
)

```

```

(defrule SoftgoalrootFile::ruleSaveBF "This rule saves the Type in softgoalFile.clp"
(declare (salience 85))
(basesoftgoal
(sgId ?idSg )
(sgQualityAttribute ?QualityAttributeSg )
(sgSubjectObject ?SubjectObjectSg ))

=>
(printout baseFile "Type: " ?QualityAttributeSg crlf )

```



```

(synonymoustopic
(topic ?topic )
(syntopic ?syntopic))
=>
(assert (synSoftgoal (ttId(gensym)) (synType ?syntype)(synTopic ?syntopic) ))
)

(defrule Synonymous::rulePrintmessageSynonymous "Print informational message."
(declare (salience 0))
=>
(printout t crlf "All the synonyms were inserted in Synsoftgoal" crlf)
)

(defrule Synonymous::ruleChanceModule " Change focus to Main module"
(declare (salience 0))
?fact <-(module choice 3)
=>
(assert (menu-level module main))
(retract ?fact)
)

..*****
;;
*****
; Inserting Module
..*****
;;
*****

(defmodule Inserting
(import MAIN ?ALL)
)

(deffunction Inserting::createElement ( )
(printout t "Enter below the two softgoals field:" crlf crlf
"the synonyms of the <[TYPE]QUALITY ATTRIBUTE>
and the <[TOPIC]SUBJECT/OBJECT LAL>." crlf crlf)

(printout t "Enter with QUALITY ATTRIBUTE: ")
(bind ?SynType (readline))
(printout t "Enter with SUBJECT/OBJECT LAL: ")
(bind ?SynTopic (readline))
(assert (synSoftgoal (ttId(gensym)) (synType ?SynType)(synTopic ?SynTopic)))
(printout t crlf crlf "Synonym of softgoal successfully inserted " crlf)
)

::::::::::::::::::::::::::::::::::::: Rules Inserting Softgoal :::::::::::::::::::::::::::::::::::::::

(defrule Inserting::rulePrintsTypeTopic " Change focus to Main module"
(declare (salience 0))

```

```

?fact <-(module choice 4)
=>
(createElement)
(assert (menu-level module main))
(retract ?fact)
)

..*****
,,
*****
; DisplaySoftgoal Module
..*****
,,
*****

;; Print softgoals and Synsoftgoals
(defmodule DisplaySoftgoal
(import MAIN ?ALL)
)

;;;;;;;;;;;;; Rules DisplaySoftgoal Softgoal ;;;;;;;;;;;;;;

(defrule DisplaySoftgoal::rulePrintsTypeTopic "This rule Display Synsoftgoal found
in the fact basis."
(declare (salience 65))
(synSoftgoal
(ttId ?tId )
(synType ?Type)
(synTopic ?Topic)
)
=>
(printout t "Id:( " ?tId ") SynSoftgoal: "?Type" [" ?Topic "]" crlf)
)

(defrule DisplaySoftgoal::ruleDisplaysoftgoalroot
"This rule Display softgoal found in the fact basis."
(declare (salience 60))
(basesoftgoal
(sgId ?idSg )
(sgQualityAttribute ?QualityAttributeSg )
(sgSubjectObject ?SubjectObjectSg )
)
=>
(printout t "Id:( " ?idSg ")Softgoal: "?QualityAttributeSg" [" ?SubjectObjectSg "]"
crlf)
)

(defrule DisplaySoftgoal::ruleChangeModule " Change focus to Main module"
(declare (salience 0))
?fact <-(module choice 5)
=>
(assert (menu-level module main))

```

```

(retract ?fact)
)

..*****
;
*****

; Single Deleting Module

..*****
;
*****

;; excluir metas repetidas

(defmodule SingleDeleting
(import MAIN ?ALL)
)

::::::::::::::::::::::::::::::::::::; Funicton Single Deleting ::::::::::::::::::::::::::::::

(deffunction SingleDeleting::DeleteOneSynSoftgoal ( )
(printout t "Enter below the two softgoals field:" crlf crlf
"the synonyms of the <[TYPE]QUALITY ATTRIBUTE>
and the <[TOPIC]SUBJECT/OBJECT LAL>." crlf crlf)

(printout t crlf "Enter with QUALITY ATTRIBUTE: ")
(bind ?dsynType (readline))
(printout t crlf "Enter with SUBJECT/OBJECT LAL: ")
(bind ?dsynTopic (readline))
(if (do-for-all-facts ((?f synSoftgoal))(and (eq ?f:synType ?dsynType)
(eq ?f:synTopic ?dsynTopic))
(retract ?f)
TRUE)
then
(printout t crlf "The SynSoftgoals were successfully deleted" crlf crlf)
else
(printout t crlf "Found no SynSoftgoal, enter a new SynSoftgoal" crlf))
)

(defrule SingleDeleting::ruleChangeModule " Change focus to Main module"
(declare (salience 0))
?fact <-(module choice 6)
=>
(DeleteOneSynSoftgoal)
(assert (menu-level module main))
(retract ?fact)
)

```

```

..*****
,,
*****
; Duplicate Softgoal Module
..*****
,,
*****

```

```

(defmodule DuplicateSoftgoal
(import MAIN ?ALL)
)

```

```

,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,; Rules Double Deleting ;,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,

```

```

;;(defrule DuplicateSoftgoal::ruleDeleteSynSoftgoal
;;"This rule deletes duplicate synsoftgoal found in the facts base."
;; (declare (salience 50))
;; ?fact3 <- (synSoftgoal
;; (ttId ?ttId1)
;; (synTopic ?syntopic1)
;; (synType ?syntype1)
;; )
;; ?fact4 <- (synSoftgoal
;; (ttId ?ttId2)
;; (synTopic ?syntopic2)
;; (synType ?syntype2)
;; )

;; (test (and (neq ?ttId1 ?ttId2) (eq ?syntype2 ?syntype1 )(eq ?syntopic2 ?syntopic1)))
;; =>
;; (retract ?fact4)
;;)

```

```

(defrule DuplicateSoftgoal::ruleDeleteSoftgoal
"This rule deletes duplicate synsoftgoal found in the facts base."

```

```

(declare (salience 50))
(synSoftgoal
(ttId ?ttId1)
(synTopic ?syntopic1)
(synType ?syntype1))

```

```

?fact2 <- (synSoftgoal
(ttId ?ttId2)
(synTopic ?syntopic2)
(synType ?syntype2))

```

```

(test (and (neq ?ttId1 ?ttId2)
(eq ?syntopic2 ?syntopic1)
(eq ?syntype2 ?syntype1)))
=>
(retract ?fact2))

```

```
(defrule DuplicateSoftgoal::rulePrintmessageDeleting "Print informational message."
(declare (salience 0))
=>
(printout t crlf crlf "All of SynSoftgoals repeated were deleting" crlf)

)
```

```
(defrule DuplicateSoftgoal::ruleChangeModule " Change focus to Main module"
(declare (salience 0))
?fact <-(module choice 7)
=>
(assert (menu-level module main))
(retract ?fact)
)
```

```
..*****
,,
*****
; Base File Module
..*****
,,
*****
;; Creates final base
```

```
(defmodule BaseFile
(import MAIN ?ALL)
)
```

```
;;;;;;;;;;;;; Rules Base File Softgoal ;;;;;;;;;;;;;;
```

```
(defrule BaseFile::ruleSynBaseFile "This rule is responsible for creating the file
containing
softgoals root and their Synonyms"
(declare (salience 55))
=>
(open "BasesoftgoalFile.clp" baseFile2 "a")
(printout t crlf "[Opened file BasesoftgoalFile.clp.] " crlf )
(assert (startWriteInFile))
)
```

```
(defrule BaseFile::ruleSelectsoftgoal "This rule saves softgoal found in the fact
basis in BasesoftgoalFile.clp."
(declare (salience 40))
(softgoalElement
(sgId ?idSg )
(sgQualityAttribute ?QualityAttributeSg )
(sgSubjectObject ?SubjectObjectSg )
(sgGoalId ?GoalIdSg )
(sgActorName ?ActorNameSg ))
```

```
=>
(printout baseFile2 ("?"idSg)" ?QualityAttributeSg " " "[" ?SubjectObjectSg "]"
crlf)
)
```

```
(defrule BaseFile::rulesoftgoal "This rule is responsible for saving synonyms in
BasesoftgoalFile.clp"
```

```
(declare (salience 45) )
(synSoftgoal
(ttId ?SynId )
(synType ?syntpe )
(synTopic ?syntpic))
```

```
=>
(printout baseFile2 ("?"SynId ") ?syntpe " " "[" ?syntpic "]" crlf)
)
```

```
(defrule BaseFile::ruleCloseSynBaseFile "This rule is responsible for closing the file
softgoalFile.clp"
```

```
(declare (salience 30))
?fact<-(startWriteInFile) ;; assignment
```

```
=>
(retract ?fact) ;; removes priority
(close baseFile2)
(printout t "[Closed file BasesoftgoalFile.clp].crlf)
)
```

```
(defrule BaseFile::ruleOpenNewSynonymousFile "This rule is responsible for
creating the file
containing actions and their classification"
```

```
(declare (salience 35))
=>
(open "NewSynonymousFile.clp" NewBaseFile "a")
(printout t crlf "[Opened file NewSynonymousFile.clp.]" crlf)
;create header to define facts in main module
(printout NewBaseFile "(deffacts MAIN::SynonymSoftgoal" crlf)
(assert (startWriteInFile))
)
```

```
(defrule BaseFile::ruleUpdateNewSynonymousFile
"This rule is responsible for creating the file containing deffacts"
```

```
(declare (salience 20))
(synSoftgoal (ttId ?sId) (synType ?stype) (synTopic ?stopic))
=>
(printout NewBaseFile "(synSoftgoal (ttId ?sId) ) (synType \""?stype\") (synTopic
\""?stopic\")" crlf)
)
```

```
(defrule BaseFile::ruleInsertBaseFile " "
(declare (salience 0))
```

```
=>
```

```
;the bracket closes the definition of actions facts
```

```
(printout NewBaseFile " ")
)
```

```
)
```

```
(defrule BaseFile::ruleCloseNewSynonymousFile "This rule is responsible for
closing the file NewSynonymousFile.clp"
```

```
(declare (salience 15))
```

```
?fact<-(startWriteInFile)
```

```
=>
```

```
(retract ?fact)
```

```
(close NewBaseFile)
```

```
(printout t "Closed file NewSynonymousFile.clp."crlf)
```

```
)
```

```
(defrule BaseFile::ruleChangeModule " Change focus to Main module"
```

```
(declare (salience 0))
```

```
?fact <-(module choice 8)
```

```
=>
```

```
(assert (menu-level module main))
```

```
(retract ?fact)
```

```
)
```

```
:: Fim da TEKBS
```

9 APÊNDICE B: BASE DE FATOS

::

;Um exemplo de uso. Fonte:

;The expert committee

;Agent-Oriented Modelling: Software Versus the World

(deffacts MAIN::requirements "Expert Committee"

(elementLel (id gen1) (name "area of the article") (type object))

(elementLel (id gen3) (name "author") (type subject))

(behavioralResponse (bhId gen4) (bhLelElementName "author") (bhDescription
"sends article to the conference") (bhType hardgoal) (bhProposedType null)
(bhCertainFactor 0.0) (bhActionId null))

(behavioralResponse (bhId gen5) (bhLelElementName "author") (bhDescription
"receives result from review of the article") (bhType hardgoal) (bhProposedType
null) (bhCertainFactor 0.0) (bhActionId null))

(behavioralResponse (bhId gen6) (bhLelElementName "author") (bhDescription
"prepares camera-ready of the accepted article") (bhType hardgoal)
(bhProposedType null) (bhCertainFactor 0.0) (bhActionId null))

(elementLel (id gen7) (name "article") (type object))

(behavioralResponse (bhId gen8) (bhLelElementName "article") (bhDescription "it is
reviewed by three reviewers") (bhType hardgoal) (bhProposedType null)
(bhCertainFactor 0.0) (bhActionId null))

(behavioralResponse (bhId gen9) (bhLelElementName "article") (bhDescription
"when the review process generate a conflict, the article is reviewed by a committee
member") (bhType hardgoal) (bhProposedType null) (bhCertainFactor 0.0)
(bhActionId null))

(elementLel (id gen10) (name "camera-ready") (type object))

(behavioralResponse (bhId gen11) (bhLelElementName "camera-ready")
(bhDescription "prepared by the author") (bhType hardgoal) (bhProposedType null)
(bhCertainFactor 0.0) (bhActionId null))

(elementLel (id gen13) (name "chair") (type subject))

(behavioralResponse (bhId gen14) (bhLelElementName "chair") (bhDescription
"receives articles from the authors") (bhType hardgoal) (bhProposedType null)
(bhCertainFactor 0.0) (bhActionId null))

(behavioralResponse (bhId gen15) (bhLelElementName "chair") (bhDescription
"draws up proposal for review for each reviewer, putting name and area of the
article, authors and institution") (bhType hardgoal) (bhProposedType null)
(bhCertainFactor 0.0) (bhActionId null))

(behavioralResponse (bhId gen17) (bhLelElementName "chair") (bhDescription " sends
articles submitted to the event to reviewers already defined") (bhType
hardgoal) (bhProposedType null) (bhCertainFactor 0.0) (bhActionId null))

(behavioralResponse (bhId gen18) (bhLelElementName "chair") (bhDescription
"sends the review result to the author") (bhType hardgoal) (bhProposedType null)
(bhCertainFactor 0.0) (bhActionId null))

(elementLel (id gen19) (name "committee member") (type subject))

(behavioralResponse (bhId gen21) (bhLelElementName "committee member")
 (bhDescription "votes for the conflicts resolution") (bhType hardgoal)
 (bhProposedType null) (bhCertainFactor 0.0) (bhActionId null))
 (elementLel (id gen22) (name "conference") (type object))
 (behavioralResponse (bhId gen24) (bhLelElementName "conference")
 (bhDescription "authors submit articles") (bhType hardgoal) (bhProposedType null)
 (bhCertainFactor 0.0) (bhActionId null))
 (elementLel (id gen25) (name "conflict") (type state))
 (elementLel (id gen27) (name "deadline for acceptance of articles") (type object))
 (behavioralResponse (bhId gen28) (bhLelElementName "deadline for acceptance of
 articles ") (bhDescription "the articles are not accepted after deadline") (bhType
 hardgoal) (bhProposedType null) (bhCertainFactor 0.0) (bhActionId null))
 (elementLel (id gen29) (name "deadline of proposal") (type object))
 (behavioralResponse (bhId gen30) (bhLelElementName "deadline of proposal")
 (bhDescription "the reviewer must respond the proposal until the deadline") (bhType
 hardgoal) (bhProposedType null) (bhCertainFactor 0.0) (bhActionId null))
 (elementLel (id gen31) (name "deadline of review") (type object))
 (behavioralResponse (bhId gen32) (bhLelElementName "deadline of review ")
 (bhDescription "reviewers must respond the reviews until the deadline") (bhType
 hardgoal) (bhProposedType null) (bhCertainFactor 0.0) (bhActionId null))
 (elementLel (id gen33) (name "evaluate reviews") (type verb))
 (elementLel (id gen35) (name "general coordinator") (type subject))
 (behavioralResponse (bhId gen36) (bhLelElementName "general coordinator")
 (bhDescription "invites researchers to be members of the program committee")
 (bhType hardgoal) (bhProposedType null) (bhCertainFactor 0.0) (bhActionId null))
 (behavioralResponse (bhId gen37) (bhLelElementName "general coordinator")
 (bhDescription "decides the deadlines for receipt of articles ,for the proposal
 acceptance and review of the articles") (bhType hardgoal) (bhProposedType null)
 (bhCertainFactor 0.0) (bhActionId null))
 (elementLel (id gen38) (name "institution") (type object))
 (elementLel (id gen40) (name "list of articles") (type object))
 (behavioralResponse (bhId gen41) (bhLelElementName "list of articles")
 (bhDescription "prepared by the chair of the conference") (bhType hardgoal)
 (bhProposedType null) (bhCertainFactor 0.0) (bhActionId null))
 (elementLel (id gen42) (name "prepare camera-ready") (type verb))
 (behavioralResponse (bhId gen43) (bhLelElementName "prepare camera-ready")
 (bhDescription "if not delivered, the article is not accepted") (bhType hardgoal)
 (bhProposedType null) (bhCertainFactor 0.0) (bhActionId null))
 (elementLel (id gen44) (name "proposal") (type object))
 (behavioralResponse (bhId gen45) (bhLelElementName "proposal") (bhDescription
 "sent by the chair to reviewers") (bhType hardgoal) (bhProposedType null)
 (bhCertainFactor 0.0) (bhActionId null))
 (behavioralResponse (bhId gen47) (bhLelElementName "proposal") (bhDescription
 "composed by the information of up to 3 articles") (bhType hardgoal)
 (bhProposedType null) (bhCertainFactor 0.0) (bhActionId null))
 (behavioralResponse (bhId gen48) (bhLelElementName "proposal") (bhDescription
 "article that isn't accepted should be relocated to another reviewer") (bhType
 hardgoal) (bhProposedType null) (bhCertainFactor 0.0) (bhActionId null))
 (elementLel (id gen49) (name "quality") (type state))
 (elementLel (id gen51) (name "relation of reviewers") (type object))

(behavioralResponse (bhId gen52) (bhLelElementName "relation of reviewers")
 (bhDescription "prepared by the chair of the conference") (bhType hardgoal)
 (bhProposedType null) (bhCertainFactor 0.0) (bhActionId null))
 (elementLel (id gen53) (name "researcher") (type subject))
 (behavioralResponse (bhId gen54) (bhLelElementName "researcher") (bhDescription
 "submits articles") (bhType hardgoal) (bhProposedType null) (bhCertainFactor 0.0)
 (bhActionId null))
 (elementLel (id gen56) (name "review article") (type verb))
 (behavioralResponse (bhId gen57) (bhLelElementName "review article")
 (bhDescription "executed by the reviewers") (bhType hardgoal) (bhProposedType
 null) (bhCertainFactor 0.0) (bhActionId null))
 (elementLel (id gen59) (name "reviewer") (type subject))
 (behavioralResponse (bhId gen60) (bhLelElementName "reviewer") (bhDescription
 "reviews up to 3 articles submitted to the conference") (bhType hardgoal)
 (bhProposedType null) (bhCertainFactor 0.0) (bhActionId null))
 (elementLel (id gen62) (name "vote conflict") (type verb))
 (behavioralResponse (bhId gen63) (bhLelElementName "vote conflict")
 (bhDescription "Withdraws the article submitted from the situation of 1 acceptance,
 1 rejection and an indifferent vote") (bhType hardgoal) (bhProposedType null)
 (bhCertainFactor 0.0) (bhActionId null))
 (actorElement (actorId gen64) (actorName "reviewer"))
 (actorElement (actorId gen65) (actorName "researcher"))
 (actorElement (actorId gen66) (actorName "general coordinator"))
 (actorElement (actorId gen67) (actorName "committee member"))
 (actorElement (actorId gen68) (actorName "chair"))
 (actorElement (actorId gen69) (actorName "author"))
 (goalElement (goalId gen70) (goalElementName "vote conflict") (goalSubjectObject
 "conflict") (goalVerb "resolved") (goalSubject "member committee") (goalBhId
 gen63))
 (behavioralResponse (bhId gen61) (bhLelElementName "reviewer") (bhDescription
 "can be invited to vote about an conflict in articles review") (bhType softgoal)
 (bhProposedType null) (bhCertainFactor 0.0) (bhActionId null))
 (behavioralResponse (bhId gen58) (bhLelElementName "review article")
 (bhDescription "must obey the deadline of review") (bhType softgoal)
 (bhProposedType null) (bhCertainFactor 0.0) (bhActionId null))
 (goalElement (goalId gen72) (goalElementName "review article") (goalSubjectObject
 "article") (goalVerb "published") (goalSubject "chair") (goalBhId gen57))
 (behavioralResponse (bhId gen55) (bhLelElementName "researcher") (bhDescription
 "participates in the conference") (bhType softgoal) (bhProposedType null)
 (bhCertainFactor 0.0) (bhActionId null))
 (goalElement (goalId gen73) (goalElementName "researcher") (goalSubjectObject
 "articles") (goalVerb "published") (goalSubject "chair") (goalBhId gen54))
 (behavioralResponse (bhId gen50) (bhLelElementName "quality") (bhDescription
 "includes good writing, clarity of ideas, good presentation and relevance of the
 proposal") (bhType softgoal) (bhProposedType null) (bhCertainFactor 0.0)
 (bhActionId null))
 (behavioralResponse (bhId gen46) (bhLelElementName "proposal") (bhDescription
 "reviewer evaluates the proposal and informs the chair if accepted or not review the
 articles") (bhType softgoal) (bhProposedType null) (bhCertainFactor 0.0)
 (bhActionId null))

(goalElement (goalId gen79) (goalElementName "prepare camera-ready")
(goalSubjectObject "article") (goalVerb "published") (goalSubject "chair") (goalBhId
gen43))
(behavioralResponse (bhId gen39) (bhLelElementName "institution") (bhDescription
" the author and reviewer can't be affiliated to the same institution") (bhType
softgoal) (bhProposedType null) (bhCertainFactor 0.0) (bhActionId null))
(goalElement (goalId gen81) (goalElementName "general coordinator")
(goalSubjectObject "deadline") (goalVerb "fixed") (goalSubject "chair") (goalBhId
gen37))
(goalElement (goalId gen83) (goalElementName "general coordinator")
(goalSubjectObject "committee") (goalVerb "compound") (goalSubject "researcher")
(goalBhId gen36))
(goalElement (goalId gen84) (goalElementName "researcher") (goalSubjectObject
"conference") (goalVerb "realized") (goalSubject "chair") (goalBhId nil))
(goalElement (goalId gen85) (goalElementName "chair") (goalSubjectObject
"conference") (goalVerb "realized") (goalSubject "null") (goalBhId nil))
(behavioralResponse (bhId gen34) (bhLelElementName "evaluate reviews")
(bhDescription "evaluates the quality of review of the articles") (bhType softgoal)
(bhProposedType null) (bhCertainFactor 0.0) (bhActionId null))
(behavioralResponse (bhId gen26) (bhLelElementName "conflict") (bhDescription
"article need to receive another review") (bhType softgoal) (bhProposedType null)
(bhCertainFactor 0.0) (bhActionId null))
(behavioralResponse (bhId gen23) (bhLelElementName "conference")
(bhDescription "researchers participate in the conference") (bhType softgoal)
(bhProposedType null) (bhCertainFactor 0.0) (bhActionId null))
(goalElement (goalId gen88) (goalElementName "committee member")
(goalSubjectObject "conflict") (goalVerb "resolved") (goalSubject "null") (goalBhId
gen21))
(behavioralResponse (bhId gen20) (bhLelElementName "committee member")
(bhDescription "evaluates the reviews") (bhType softgoal) (bhProposedType null)
(bhCertainFactor 0.0) (bhActionId null))
(goalElement (goalId gen89) (goalElementName "chair") (goalSubjectObject
"review") (goalVerb "communicated") (goalSubject "null") (goalBhId gen18))
(goalElement (goalId gen90) (goalElementName "chair") (goalSubjectObject
"article") (goalVerb "reviewed") (goalSubject "reviewer") (goalBhId gen17))
(behavioralResponse (bhId gen16) (bhLelElementName "chair") (bhDescription
"sends the review proposal for each of the reviewers,informing deadline for
acceptance of the proposal and the deadline of review of the articles") (bhType
softgoal) (bhProposedType null) (bhCertainFactor 0.0) (bhActionId null))
(goalElement (goalId gen92) (goalElementName "chair") (goalSubjectObject
"proposal") (goalVerb "accepted") (goalSubject "reviewer") (goalBhId gen15))
(goalElement (goalId gen93) (goalElementName "reviewer") (goalSubjectObject
"proposal") (goalVerb "forwarded") (goalSubject "chair") (goalBhId nil))
(goalElement (goalId gen94) (goalElementName "chair") (goalSubjectObject
"article") (goalVerb "submitted") (goalSubject "author") (goalBhId gen14))
(behavioralResponse (bhId gen12) (bhLelElementName "camera-ready")
(bhDescription "must respect the deadline") (bhType softgoal) (bhProposedType
null) (bhCertainFactor 0.0) (bhActionId null))
(goalElement (goalId gen99) (goalElementName "author") (goalSubjectObject
"article") (goalVerb "published") (goalSubject "chair") (goalBhId gen6))

(goalElement (goalId gen100) (goalElementName "chair") (goalSubjectObject
"camera-ready") (goalVerb "sent") (goalSubject "author") (goalBhId nil))
(goalElement (goalId gen101) (goalElementName "author") (goalSubjectObject
"article") (goalVerb "accepted") (goalSubject "null") (goalBhId gen5))
(goalElement (goalId gen102) (goalElementName "author") (goalSubjectObject
"article") (goalVerb "reviewed") (goalSubject "reviewer") (goalBhId gen4))
(goalElement (goalId gen103) (goalElementName "reviewer") (goalSubjectObject
"article") (goalVerb "submitted") (goalSubject "author") (goalBhId nil))
(behavioralResponse (bhId gen2) (bhLelElementName "area of the article")
(bhDescription "must be in the same area of interest of the reviewer") (bhType
softgoal) (bhProposedType null) (bhCertainFactor 0.0) (bhActionId null))
(softgoalElement (sgId gen104) (sgQualityAttribute "quality") (sgSubjectObject
"review") (sgGoalId gen98) (sgActorName "reviewer") (sgBhId gen2))
(softgoalElement (sgId gen105) (sgQualityAttribute "punctuality") (sgSubjectObject
"publication") (sgGoalId gen96) (sgActorName "chair") (sgBhId gen12))
(softgoalElement (sgId gen107) (sgQualityAttribute "quality") (sgSubjectObject
"review") (sgGoalId gen96) (sgActorName "chair") (sgBhId gen20))
(softgoalElement (sgId gen108) (sgQualityAttribute "acknowledge") (sgSubjectObject
"committee") (sgGoalId gen102) (sgActorName "reviewer") (sgBhId gen23))
(softgoalElement (sgId gen109) (sgQualityAttribute "quality") (sgSubjectObject
"review") (sgGoalId gen97) (sgActorName "committe") (sgBhId gen26))
(softgoalElement (sgId gen110) (sgQualityAttribute "quality") (sgSubjectObject
"review") (sgGoalId gen98) (sgActorName "reviewer") (sgBhId gen34))
(softgoalElement (sgId gen111) (sgQualityAttribute "honest") (sgSubjectObject
"review") (sgGoalId gen98) (sgActorName "reviewer") (sgBhId gen39))
(softgoalElement (sgId gen113) (sgQualityAttribute "quality") (sgSubjectObject
"article") (sgGoalId gen96) (sgActorName "chair") (sgBhId gen50))
(softgoalElement (sgId gen114) (sgQualityAttribute "acknowledge") (sgSubjectObject
"committee") (sgGoalId gen84) (sgActorName "chair") (sgBhId gen55))
(softgoalElement (sgId gen115) (sgQualityAttribute "no delay") (sgSubjectObject
"review") (sgGoalId gen102) (sgActorName "reviewer") (sgBhId gen58))
(softgoalElement (sgId gen116) (sgQualityAttribute "fair") (sgSubjectObject
"review") (sgGoalId gen97) (sgActorName "committee") (sgBhId gen61))
(softgoalElement (sgId gen112) (sgQualityAttribute "correct") (sgSubjectObject
"evaluation") (sgGoalId gen101) (sgActorName "chair") (sgBhId gen46))
(goalElement (goalId gen98) (goalElementName "reviewer") (goalSubjectObject
"article") (goalVerb "reviewed") (goalSubject "null") (goalBhId gen8))
(goalElement (goalId gen97) (goalElementName "committee member")
(goalSubjectObject "conflict") (goalVerb "decided") (goalSubject "null") (goalBhId
gen9))
(goalElement (goalId gen96) (goalElementName "chair") (goalSubjectObject
"article") (goalVerb "published") (goalSubject "null") (goalBhId gen11))
(goalElement (goalId gen87) (goalElementName "reviewer") (goalSubjectObject
"articles") (goalVerb "reviewed") (goalSubject "null") (goalBhId gen24))
(goalElement (goalId gen86) (goalElementName "reviewer") (goalSubjectObject
"articles") (goalVerb "received") (goalSubject "null") (goalBhId gen30))
(goalElement (goalId gen80) (goalElementName "chair") (goalSubjectObject
"proposals") (goalVerb "fowarded") (goalSubject "null") (goalBhId gen41))
(goalElement (goalId gen78) (goalElementName "reviewer") (goalSubjectObject
"proposal") (goalVerb "approved") (goalSubject "null") (goalBhId gen45))

(goalElement (goalId gen77) (goalElementName "reviewer") (goalSubjectObject
"proposal") (goalVerb "accepted") (goalSubject "null") (goalBhId gen47))
(goalElement (goalId gen76) (goalElementName "reviewer") (goalSubjectObject
"proposal") (goalVerb "accpeted") (goalSubject "null") (goalBhId gen48))
(goalElement (goalId gen75) (goalElementName "chair") (goalSubjectObject
"proposal") (goalVerb "fowarded") (goalSubject "null") (goalBhId gen52))
(action (actionId gen33) (actionStem "submit") (actionVerb "submitted")
(actionType hardgoal) (actionSuccess 1) (actionOccurrence 2))
(action (actionId gen32) (actionStem "execut") (actionVerb "executed") (actionType
hardgoal) (actionSuccess 1) (actionOccurrence 2))
(action (actionId gen30) (actionStem "deliv") (actionVerb "delivered") (actionType
hardgoal) (actionSuccess 1) (actionOccurrence 2))
(action (actionId gen29) (actionStem "decid") (actionVerb "decides") (actionType
hardgoal) (actionSuccess 1) (actionOccurrence 2))
(action (actionId gen28) (actionStem "invit") (actionVerb "invites") (actionType
hardgoal) (actionSuccess 1) (actionOccurrence 2))
(action (actionId gen27) (actionStem "vote") (actionVerb "votes") (actionType
hardgoal) (actionSuccess 1) (actionOccurrence 2))
(action (actionId gen26) (actionStem "send") (actionVerb "sends") (actionType
hardgoal) (actionSuccess 1) (actionOccurrence 2))
(action (actionId gen23) (actionStem "receiv") (actionVerb "receives") (actionType
hardgoal) (actionSuccess 1) (actionOccurrence 2))
(action (actionId gen22) (actionStem "prepar") (actionVerb "prepares") (actionType
hardgoal) (actionSuccess 1) (actionOccurrence 2))
(action (actionId gen19) (actionStem "respect") (actionVerb "respect") (actionType
softgoal) (actionSuccess 1) (actionOccurrence 2))
(action (actionId gen18) (actionStem "evalu") (actionVerb "evaluates") (actionType
softgoal) (actionSuccess 1) (actionOccurrence 2))
(action (actionId gen17) (actionStem "particip") (actionVerb "participate")
(actionType softgoal) (actionSuccess 1) (actionOccurrence 2))
(action (actionId gen16) (actionStem "need") (actionVerb "need") (actionType
softgoal) (actionSuccess 1) (actionOccurrence 2))
(action (actionId gen13) (actionStem "obei") (actionVerb "obey") (actionType
softgoal) (actionSuccess 1) (actionOccurrence 2))
(action (actionId gen10) (actionStem "review") (actionVerb "reviewed") (actionType
hardgoal) (actionSuccess 1) (actionOccurrence 2))
(action (actionId gen6) (actionStem "respond") (actionVerb "respond") (actionType
hardgoal) (actionSuccess 1) (actionOccurrence 2))
(action (actionId gen4) (actionStem "sent") (actionVerb "sent") (actionType
hardgoal) (actionSuccess 1) (actionOccurrence 2))
(action (actionId gen3) (actionStem "compos") (actionVerb "composed")
(actionType hardgoal) (actionSuccess 1) (actionOccurrence 2))
(action (actionId gen2) (actionStem "accept") (actionVerb "accepted") (actionType
hardgoal) (actionSuccess 1) (actionOccurrence 2))
)

10 APÊNDICE C: SOFTGOALFILE

Type: quality
Type: punctuality
Type: acknowledge
Type: honest
Type: quality
Type: no delay
Type: fair
Type: correct
Topic: review
Topic: publication
Topic: committee
Topic: review
Topic: article
Topic: review
Topic: review
Topic: evaluation

11 APÊNDICE D: RESULTADO DA BUSCA DE SINÔNIMOS

(deffacts synonymtype
 (synonymoustype (type "quality") (syntype "caliber"))
 (synonymoustype (type "quality") (syntype "calibre"))
 (synonymoustype (type "quality") (syntype "character"))
 (synonymoustype (type "quality") (syntype "lineament"))
 (synonymoustype (type "quality") (syntype "timbre"))
 (synonymoustype (type "quality") (syntype "timber"))
 (synonymoustype (type "quality") (syntype "tone"))
 (synonymoustype (type "quality") (syntype "choice"))
 (synonymoustype (type "quality") (syntype "prime"))
 (synonymoustype (type "quality") (syntype "prize"))
 (synonymoustype (type "quality") (syntype "select"))
 (synonymoustype (type "punctuality") (syntype "promptness"))
 (synonymoustype (type "acknowledge") (syntype "admit"))
 (synonymoustype (type "acknowledge") (syntype "receipt"))
 (synonymoustype (type "acknowledge") (syntype "notice"))
 (synonymoustype (type "acknowledge") (syntype "recognize"))
 (synonymoustype (type "acknowledge") (syntype "recognise"))
 (synonymoustype (type "acknowledge") (syntype "recognize"))
 (synonymoustype (type "acknowledge") (syntype "recognise"))
 (synonymoustype (type "acknowledge") (syntype "know"))
 (synonymoustype (type "honest") (syntype "honorable"))
 (synonymoustype (type "honest") (syntype "dependable"))
 (synonymoustype (type "honest") (syntype "reliable"))
 (synonymoustype (type "honest") (syntype "true"))
 (synonymoustype (type "honest") (syntype "good"))
 (synonymoustype (type "honest") (syntype "fair"))
 (synonymoustype (type "quality") (syntype "caliber"))
 (synonymoustype (type "quality") (syntype "calibre"))
 (synonymoustype (type "quality") (syntype "character"))
 (synonymoustype (type "quality") (syntype "lineament"))
 (synonymoustype (type "quality") (syntype "timbre"))
 (synonymoustype (type "quality") (syntype "timber"))
 (synonymoustype (type "quality") (syntype "tone"))
 (synonymoustype (type "quality") (syntype "choice"))
 (synonymoustype (type "quality") (syntype "prime"))
 (synonymoustype (type "quality") (syntype "prize"))
 (synonymoustype (type "quality") (syntype "select"))
 (synonymoustype (type "no delay") (syntype "nobelium"))
 (synonymoustype (type "no delay") (syntype "atomic number 102"))
 (synonymoustype (type "no delay") (syntype "no more"))
 (synonymoustype (type "fair") (syntype "carnival"))
 (synonymoustype (type "fair") (syntype "funfair"))
 (synonymoustype (type "fair") (syntype "bazaar"))
 (synonymoustype (type "fair") (syntype "just"))
 (synonymoustype (type "fair") (syntype "fairish"))

(synonymoustype (type "fair") (syntype "reasonable"))
 (synonymoustype (type "fair") (syntype "bonny"))
 (synonymoustype (type "fair") (syntype "bonnie"))
 (synonymoustype (type "fair") (syntype "comely"))
 (synonymoustype (type "fair") (syntype "sightly"))
 (synonymoustype (type "fair") (syntype "average"))
 (synonymoustype (type "fair") (syntype "mediocre"))
 (synonymoustype (type "fair") (syntype "middling"))
 (synonymoustype (type "fair") (syntype "clean"))
 (synonymoustype (type "fair") (syntype "honest"))
 (synonymoustype (type "fair") (syntype "fairish"))
 (synonymoustype (type "fair") (syntype "fairly"))
 (synonymoustype (type "fair") (syntype "clean"))
 (synonymoustype (type "fair") (syntype "fairly"))
 (synonymoustype (type "fair") (syntype "evenhandedly"))
 (synonymoustype (type "correct") (syntype "rectify"))
 (synonymoustype (type "correct") (syntype "right"))
 (synonymoustype (type "correct") (syntype "right"))
 (synonymoustype (type "correct") (syntype "compensate"))
 (synonymoustype (type "correct") (syntype "redress"))
 (synonymoustype (type "correct") (syntype "chastise"))
 (synonymoustype (type "correct") (syntype "castigate"))
 (synonymoustype (type "correct") (syntype "objurgate"))
 (synonymoustype (type "correct") (syntype "chasten"))
 (synonymoustype (type "correct") (syntype "compensate"))
 (synonymoustype (type "correct") (syntype "counterbalance"))
 (synonymoustype (type "correct") (syntype "make up"))
 (synonymoustype (type "correct") (syntype "even out"))
 (synonymoustype (type "correct") (syntype "even off"))
 (synonymoustype (type "correct") (syntype "even up"))
 (synonymoustype (type "correct") (syntype "discipline"))
 (synonymoustype (type "correct") (syntype "sort out"))
 (synonymoustype (type "correct") (syntype "decline"))
 (synonymoustype (type "correct") (syntype "slump"))
 (synonymoustype (type "correct") (syntype "adjust"))
 (synonymoustype (type "correct") (syntype "set"))
 (synonymoustype (type "correct") (syntype "right"))
 (synonymoustype (type "correct") (syntype "right"))
 (synonymoustype (type "correct") (syntype "right"))
 (synonymoustype (type "correct") (syntype "right"))
)

(deffacts synonymtopic

(synonymoustopic (topic "review") (syntopic "reappraisal"))
 (synonymoustopic (topic "review") (syntopic "revaluation"))
 (synonymoustopic (topic "review") (syntopic "reassessment"))
 (synonymoustopic (topic "review") (syntopic "critique"))
 (synonymoustopic (topic "review") (syntopic "critical review"))
 (synonymoustopic (topic "review") (syntopic "review article"))

(synonymoustopic (topic "review") (syntopic "follow-up")))
 (synonymoustopic (topic "review") (syntopic "followup")))
 (synonymoustopic (topic "review") (syntopic "reexamination")))
 (synonymoustopic (topic "review") (syntopic "limited review")))
 (synonymoustopic (topic "review") (syntopic "revue")))
 (synonymoustopic (topic "review") (syntopic "recapitulation")))
 (synonymoustopic (topic "review") (syntopic "recap")))
 (synonymoustopic (topic "review") (syntopic "brushup")))
 (synonymoustopic (topic "review") (syntopic "inspection")))
 (synonymoustopic (topic "review") (syntopic "reexamine")))
 (synonymoustopic (topic "review") (syntopic "critique")))
 (synonymoustopic (topic "review") (syntopic "go over")))
 (synonymoustopic (topic "review") (syntopic "survey")))
 (synonymoustopic (topic "review") (syntopic "brush up")))
 (synonymoustopic (topic "review") (syntopic "refresh")))
 (synonymoustopic (topic "review") (syntopic "look back")))
 (synonymoustopic (topic "review") (syntopic "retrospect")))
 (synonymoustopic (topic "publication") (syntopic "issue")))
 (synonymoustopic (topic "publication") (syntopic "publishing")))
 (synonymoustopic (topic "committee") (syntopic "admit")))
 (synonymoustopic (topic "committee") (syntopic "receipt")))
 (synonymoustopic (topic "committee") (syntopic "notice")))
 (synonymoustopic (topic "committee") (syntopic "recognize")))
 (synonymoustopic (topic "committee") (syntopic "recognise")))
 (synonymoustopic (topic "committee") (syntopic "recognize")))
 (synonymoustopic (topic "committee") (syntopic "recognise")))
 (synonymoustopic (topic "committee") (syntopic "know")))
 (synonymoustopic (topic "review") (syntopic "reappraisal")))
 (synonymoustopic (topic "review") (syntopic "revaluation")))
 (synonymoustopic (topic "review") (syntopic "reassessment")))
 (synonymoustopic (topic "review") (syntopic "critique")))
 (synonymoustopic (topic "review") (syntopic "critical review")))
 (synonymoustopic (topic "review") (syntopic "review article")))
 (synonymoustopic (topic "review") (syntopic "follow-up")))
 (synonymoustopic (topic "review") (syntopic "followup")))
 (synonymoustopic (topic "review") (syntopic "reexamination")))
 (synonymoustopic (topic "review") (syntopic "limited review")))
 (synonymoustopic (topic "review") (syntopic "revue")))
 (synonymoustopic (topic "review") (syntopic "recapitulation")))
 (synonymoustopic (topic "review") (syntopic "recap")))
 (synonymoustopic (topic "review") (syntopic "brushup")))
 (synonymoustopic (topic "review") (syntopic "inspection")))
 (synonymoustopic (topic "review") (syntopic "reexamine")))
 (synonymoustopic (topic "review") (syntopic "critique")))
 (synonymoustopic (topic "review") (syntopic "go over")))
 (synonymoustopic (topic "review") (syntopic "survey")))
 (synonymoustopic (topic "review") (syntopic "brush up")))
 (synonymoustopic (topic "review") (syntopic "refresh")))
 (synonymoustopic (topic "review") (syntopic "look back")))
 (synonymoustopic (topic "review") (syntopic "retrospect")))

(synonymoustopic (topic "article") (syntopic "clause"))
 (synonymoustopic (topic "review") (syntopic "reappraisal"))
 (synonymoustopic (topic "review") (syntopic "revaluation"))
 (synonymoustopic (topic "review") (syntopic "reassessment"))
 (synonymoustopic (topic "review") (syntopic "critique"))
 (synonymoustopic (topic "review") (syntopic "critical review"))
 (synonymoustopic (topic "review") (syntopic "review article"))
 (synonymoustopic (topic "review") (syntopic "follow-up"))
 (synonymoustopic (topic "review") (syntopic "followup"))
 (synonymoustopic (topic "review") (syntopic "reexamination"))
 (synonymoustopic (topic "review") (syntopic "limited review"))
 (synonymoustopic (topic "review") (syntopic "revue"))
 (synonymoustopic (topic "review") (syntopic "recapitulation"))
 (synonymoustopic (topic "review") (syntopic "recap"))
 (synonymoustopic (topic "review") (syntopic "brushup"))
 (synonymoustopic (topic "review") (syntopic "inspection"))
 (synonymoustopic (topic "review") (syntopic "reexamine"))
 (synonymoustopic (topic "review") (syntopic "critique"))
 (synonymoustopic (topic "review") (syntopic "go over"))
 (synonymoustopic (topic "review") (syntopic "survey"))
 (synonymoustopic (topic "review") (syntopic "brush up"))
 (synonymoustopic (topic "review") (syntopic "refresh"))
 (synonymoustopic (topic "review") (syntopic "look back"))
 (synonymoustopic (topic "review") (syntopic "retrospect"))
 (synonymoustopic (topic "review") (syntopic "reappraisal"))
 (synonymoustopic (topic "review") (syntopic "revaluation"))
 (synonymoustopic (topic "review") (syntopic "reassessment"))
 (synonymoustopic (topic "review") (syntopic "critique"))
 (synonymoustopic (topic "review") (syntopic "critical review"))
 (synonymoustopic (topic "review") (syntopic "review article"))
 (synonymoustopic (topic "review") (syntopic "follow-up"))
 (synonymoustopic (topic "review") (syntopic "followup"))
 (synonymoustopic (topic "review") (syntopic "reexamination"))
 (synonymoustopic (topic "review") (syntopic "limited review"))
 (synonymoustopic (topic "review") (syntopic "revue"))
 (synonymoustopic (topic "review") (syntopic "recapitulation"))
 (synonymoustopic (topic "review") (syntopic "recap"))
 (synonymoustopic (topic "review") (syntopic "brushup"))
 (synonymoustopic (topic "review") (syntopic "inspection"))
 (synonymoustopic (topic "review") (syntopic "reexamine"))
 (synonymoustopic (topic "review") (syntopic "critique"))
 (synonymoustopic (topic "review") (syntopic "go over"))
 (synonymoustopic (topic "review") (syntopic "survey"))
 (synonymoustopic (topic "review") (syntopic "brush up"))
 (synonymoustopic (topic "review") (syntopic "refresh"))
 (synonymoustopic (topic "review") (syntopic "look back"))
 (synonymoustopic (topic "review") (syntopic "retrospect"))
 (synonymoustopic (topic "evaluation") (syntopic "rating"))
 (synonymoustopic (topic "evaluation") (syntopic "valuation"))
 (synonymoustopic (topic "evaluation") (syntopic "rating"))

12 APÊNDICE E: BASE WORDNET COM SINÔNIMOS VÁLIDOS

(deffacts synonymtype
 (synonymoustype (type "correct") (syntype "right"))
 (synonymoustype (type "fair") (syntype "just"))
 (synonymoustype (type "fair") (syntype "reasonable"))
 (synonymoustype (type "fair") (syntype "clean"))
 (synonymoustype (type "fair") (syntype "honest"))
 (synonymoustype (type "fair") (syntype "fairly"))
 (synonymoustype (type "fair") (syntype "evenhandedly"))
 (synonymoustype (type "quality") (syntype "choice"))
 (synonymoustype (type "quality") (syntype "select"))
 (synonymoustype (type "honest") (syntype "honorable"))
 (synonymoustype (type "honest") (syntype "dependable"))
 (synonymoustype (type "honest") (syntype "reliable"))
 (synonymoustype (type "honest") (syntype "true"))
 (synonymoustype (type "honest") (syntype "fair"))
 (synonymoustype (type "acknowledge") (syntype "admit"))
 (synonymoustype (type "acknowledge") (syntype "recognize"))
 (synonymoustype (type "acknowledge") (syntype "recognise"))
 (synonymoustype (type "punctuality") (syntype "promptness"))
)

(deffacts synonymtopic
 (synonymoustopic (topic "evaluation") (syntopic "rating"))
 (synonymoustopic (topic "evaluation") (syntopic "valuation"))
 (synonymoustopic (topic "review") (syntopic "reappraisal"))
 (synonymoustopic (topic "review") (syntopic "revaluation"))
 (synonymoustopic (topic "review") (syntopic "reassessment"))
 (synonymoustopic (topic "review") (syntopic "critique"))
 (synonymoustopic (topic "review") (syntopic "critical review"))
 (synonymoustopic (topic "review") (syntopic "review article"))
 (synonymoustopic (topic "review") (syntopic "reexamination"))
 (synonymoustopic (topic "review") (syntopic "inspection"))
 (synonymoustopic (topic "review") (syntopic "reexamine"))
 (synonymoustopic (topic "review") (syntopic "go over"))
 (synonymoustopic (topic "review") (syntopic "survey"))
 (synonymoustopic (topic "article") (syntopic "clause"))
 (synonymoustopic (topic "committee") (syntopic "commission"))
 (synonymoustopic (topic "publication") (syntopic "issue"))
 (synonymoustopic (topic "publication") (syntopic "publishing"))
)

13 APÊNDICE F: BASE MACMILLAN COM SINÔNIMOS VÁLIDOS

(deffacts synonymtype
 (synonymoustype (type "correct") (syntype "accurate"))
 (synonymoustype (type "correct") (syntype "perfect"))
 (synonymoustype (type "correct") (syntype "precise"))
 (synonymoustype (type "correct") (syntype "correct"))
 (synonymoustype (type "correct") (syntype "specific"))
 (synonymoustype (type "correct") (syntype "reliable"))
 (synonymoustype (type "fair") (syntype "fair"))
 (synonymoustype (type "fair") (syntype "reasonable"))
 (synonymoustype (type "fair") (syntype "legitimate"))
 (synonymoustype (type "fair") (syntype "just"))
 (synonymoustype (type "quality") (syntype "quality"))
 (synonymoustype (type "quality") (syntype "nature"))
 (synonymoustype (type "quality") (syntype "peculiarity"))
 (synonymoustype (type "honest") (syntype "honest"))
 (synonymoustype (type "honest") (syntype "sincere"))
 (synonymoustype (type "honest") (syntype "genuine"))
 (synonymoustype (type "honest") (syntype "decent"))
 (synonymoustype (type "honest") (syntype "truthful"))
 (synonymoustype (type "honest") (syntype "irreproachable"))
 (synonymoustype (type "acknowledge") (syntype "value"))
 (synonymoustype (type "acknowledge") (syntype "attach importance"))
 (synonymoustype (type "acknowledge") (syntype "acknowledge"))
 (synonymoustype (type "punctuality") (syntype "punctual"))
 (synonymoustype (type "punctuality") (syntype "promptly"))
 (synonymoustype (type "punctuality") (syntype "in time"))
 (synonymoustype (type "punctuality") (syntype "on time"))
 (synonymoustype (type "punctuality") (syntype "prompt"))
 (synonymoustype (type "punctuality") (syntype "on the dot"))
)

(deffacts synonymtopic
 (synonymoustopic (topic "evaluation") (syntopic "consider"))
 (synonymoustopic (topic "evaluation") (syntopic "plan"))
 (synonymoustopic (topic "evaluation") (syntopic "assess"))
 (synonymoustopic (topic "evaluation") (syntopic "evaluate"))
 (synonymoustopic (topic "review") (syntopic "examination"))
 (synonymoustopic (topic "review") (syntopic "analysis"))
 (synonymoustopic (topic "review") (syntopic "review"))
 (synonymoustopic (topic "review") (syntopic "survey"))
 (synonymoustopic (topic "review") (syntopic "inspection"))
 (synonymoustopic (topic "review") (syntopic "scrutiny"))
 (synonymoustopic (topic "review") (syntopic "check"))
 (synonymoustopic (topic "review") (syntopic "search"))
 (synonymoustopic (topic "article") (syntopic "article"))
 (synonymoustopic (topic "committee") (syntopic "committee"))

(synonymoustopic (topic "committee") (syntopic "crew"))
(synonymoustopic (topic "committee") (syntopic "interest group"))
(synonymoustopic (topic "committee") (syntopic "steering committee"))
)

14 APÊNDICE G: BASE THESAURUS COM SINÔNIMOS VÁLIDOS

(deffacts synonymytype
 (synonymoustype (type "correct") (syntype "appropriate"))
 (synonymoustype (type "correct") (syntype "equitable"))
 (synonymoustype (type "correct") (syntype "legitimate"))
 (synonymoustype (type "correct") (syntype "perfect"))
 (synonymoustype (type "correct") (syntype "precise"))
 (synonymoustype (type "correct") (syntype "proper"))
 (synonymoustype (type "correct") (syntype "true"))
 (synonymoustype (type "fair") (syntype "candid"))
 (synonymoustype (type "fair") (syntype "equal"))
 (synonymoustype (type "fair") (syntype "equitable"))
 (synonymoustype (type "fair") (syntype "honest"))
 (synonymoustype (type "fair") (syntype "impartial"))
 (synonymoustype (type "fair") (syntype "lawful"))
 (synonymoustype (type "fair") (syntype "legitimate"))
 (synonymoustype (type "fair") (syntype "reasonable"))
 (synonymoustype (type "fair") (syntype "sincere"))
 (synonymoustype (type "fair") (syntype "straightforward"))
 (synonymoustype (type "fair") (syntype "trustworthy"))
 (synonymoustype (type "fair") (syntype "unbiased"))
 (synonymoustype (type "quality") (syntype "aspect"))
 (synonymoustype (type "quality") (syntype "character"))
 (synonymoustype (type "quality") (syntype "condition"))
 (synonymoustype (type "quality") (syntype "nature"))
 (synonymoustype (type "quality") (syntype "trait"))
 (synonymoustype (type "honest") (syntype "authentic"))
 (synonymoustype (type "honest") (syntype "decent"))
 (synonymoustype (type "honest") (syntype "fair"))
 (synonymoustype (type "honest") (syntype "genuine"))
 (synonymoustype (type "honest") (syntype "honorable"))
 (synonymoustype (type "honest") (syntype "reliable"))
 (synonymoustype (type "honest") (syntype "sincere"))
 (synonymoustype (type "honest") (syntype "straightforward"))
 (synonymoustype (type "honest") (syntype "true"))
 (synonymoustype (type "honest") (syntype "trustworthy"))
 (synonymoustype (type "acknowledge") (syntype "accept"))
 (synonymoustype (type "acknowledge") (syntype "recognize"))

)

(deffacts synonymytopic
 (synonymoustopic (topic "evaluation") (syntopic "appraisal"))
 (synonymoustopic (topic "evaluation") (syntopic "assessment"))
 (synonymoustopic (topic "evaluation") (syntopic "opinion"))
 (synonymoustopic (topic "review") (syntopic "analysis"))

(synonymoustopic (topic "review") (syntopic "check"))
(synonymoustopic (topic "review") (syntopic "inspection"))
(synonymoustopic (topic "review") (syntopic "revision"))
(synonymoustopic (topic "review") (syntopic "survey"))
(synonymoustopic (topic "article") (syntopic "essay"))
(synonymoustopic (topic "article") (syntopic "paper"))
(synonymoustopic (topic "committee") (syntopic "commission"))
(synonymoustopic (topic "publication") (syntopic "disclosure"))
(synonymoustopic (topic "publication") (syntopic "publishing"))
(synonymoustopic (topic "publication") (syntopic "reporting"))

)