

3. O Método ERi*c - Engenharia de Requisitos Intencional

3.1. Visão Geral do Método

O objetivo deste capítulo é apresentar o método Engenharia de Requisitos Intencional - **ERi*c**, o qual é uma contribuição à Engenharia de Requisitos de Sistemas Multi-Agentes [Wooldridge 02]. O Método ERi*c se baseia no conceito de intencionalidade como aplicado pelo Framework i* [Yu 95]. Intencionalidade reflete as motivações e os interesses dos atores de uma organização e é representada nos modelos através das metas (concretas e flexíveis). Nesta seção mostramos primeiramente uma visão geral do Método ERi*c e, nas seções seguintes, apresentamos as etapas com os objetivos, as atividades e os artefatos (documentos e diagramas) preparados para alcançarmos os objetivos desejados.

A Figura 3.1 apresenta um esquema simplificado do método, que está dividido em seis etapas. Na primeira etapa o método ERi*c faz a identificação dos atores e a elicitação das metas, em seguida trabalha na identificação das situações de dependência estratégica e depois se ocupa com a modelagem das metas dos agentes. Na quarta etapa se preocupa com a modelagem da racionalização das metas, em seguida faz a especificação das situações de dependência estratégica e, concluindo, na sexta etapa executa a análise dos modelos SD e SR.

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

Na segunda etapa a contribuição é a idéia de “Strategic Dependency Situations” - SDSituations ou Situações de Dependência Estratégica, definida por Oliveira e Cysneiros [Oliveira 06b]. As SDSituations trabalham para que, no processo de modelagem do Framework i*, a variável tempo e as ligações tipo ponte (ou “bridges”) sejam levadas em consideração através de uma abordagem situacional.

Na etapa três, a contribuição do método é a representação de metas concretas e metas flexíveis no “Intentionality Panel” - IP Diagram [Oliveira 07], ou Painel de

Intencionalidade. O Painel de Intencionalidade traz três benefícios: (a) primeiro: a representação exclusiva das metas, concretas e flexíveis, e das relações entre elas através de dependências, correlações, contribuições e equivalências em um único diagrama considerando “níveis de relevância” e o agrupamento das metas por “SDsituations”; (b) segundo: a avaliação do primeiro impacto (por não ser uma análise aprofundada) do efeito da propagação do não-alcance de metas concretas e do alcance parcial de metas flexíveis; (c) terceiro: a facilidade da avaliação da complexidade do diagrama por meio do número ciclomático [McCabe 94].

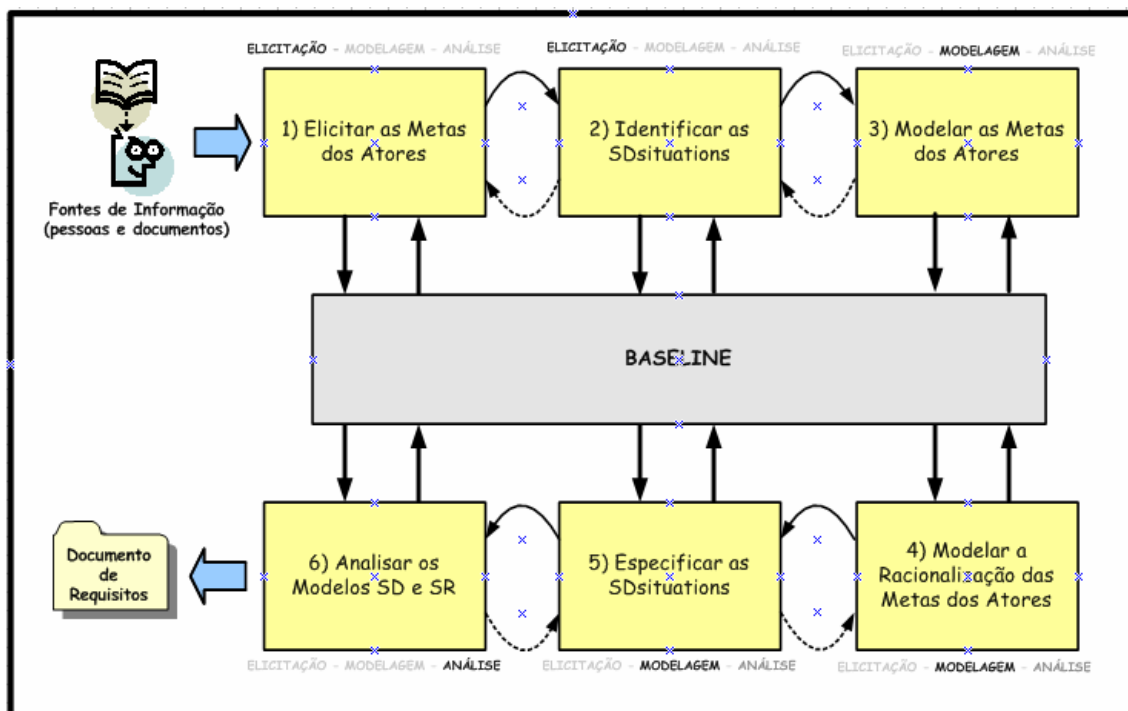


Figura 3.1 – Visão geral do método ilustrando o encadeamento das etapas

Na etapa quatro, duas contribuições do método são feitas: a idéia do detalhamento de metas concretas e de metas flexíveis em “minimodelos” SR, chamados de SRconstructs [Oliveira 07] e a orientação, ou guia, para a derivação de modelos SR a partir do modelo SD [Oliveira 08a]. Nos SRconstructs – Construtos de Razão Estratégica desenvolvem-se as estruturas *meios-fim* e *decomposição de tarefa* usadas por modelos SR do Framework i^* [Yu 95].

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

atividade, sociabilidade, adaptação, interação, colaboração, aprendizado e mobilidade [Wooldridge 02] [Silva 04] [Garcia 04].

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

Na parte central do diagrama mostramos o termo baseline de Leite e Oliveira [Leite 95] para indicar que o Método ERi*c deve ser apoiado por ferramentas de software. Essas ferramentas têm a missão de facilitar o trabalho do engenheiro de requisitos no processo e também garantir, de modo automatizado, o rastreamento dos requisitos [Ramesh 01]. As recomendações de ferramentas de software para apoio ao Método ERi*c estão definidas ao longo da especificação do método.

Propõe-se que o Método ERi*c venha a contribuir para melhorar a qualidade final dos requisitos. Por melhor qualidade final dos requisitos deve-se entender: modelos de requisitos mais completos, maior facilidade na assimilação dos requisitos, melhor organização na representação dos requisitos e, conseqüentemente, menor sobrecarga sobre os modelos.

3.2. Etapas do Método Engenharia de Requisitos Intencional – ERi*c

1) Elicitar as Metas dos Atores

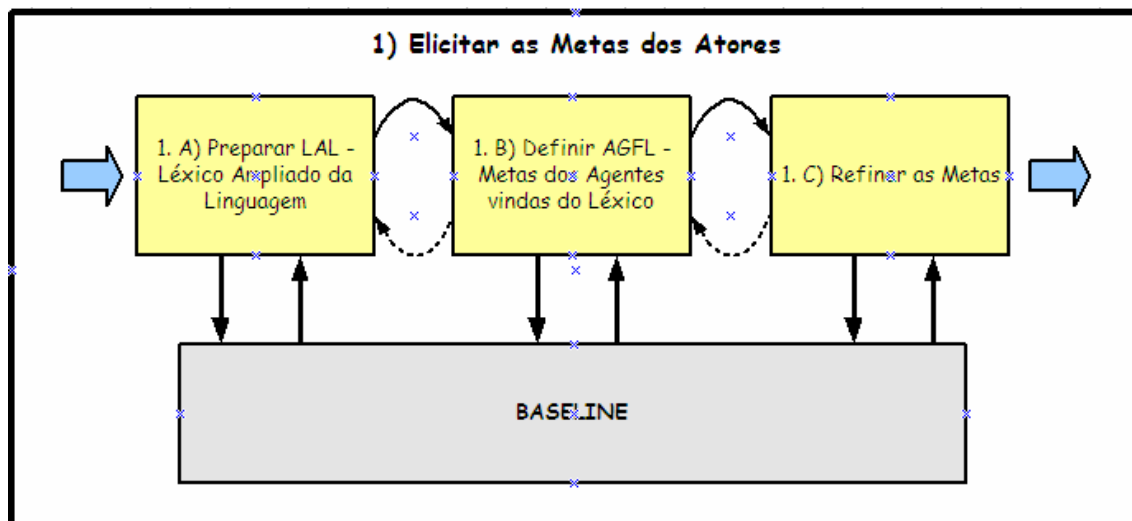


Figura 3.2 – Detalhamento da etapa 1 do Método ERi*c

Objetivo: “Elicitação e refinamento das metas para Sistemas Multi-Agentes”.

A primeira etapa é composta por três atividades: A) Preparar LAL - Léxico Ampliado da Linguagem, B) Definir AGFL - Metas dos Agentes vindas do Léxico e C) Refinar as metas, veja Figura 3.2.

1.A) Preparar LAL – Léxico Ampliado da Linguagem

Para a construção do LAL o engenheiro de requisitos primeiramente identifica as fontes de informação que podem ser utilizadas. Devem ser considerados todos os documentos disponíveis, além de todas as pessoas que podem fornecer informações sobre o UDI – Universo de Informação. O engenheiro de requisitos deve seguir os seguintes passos recomendados por Leite et al. [Leite 00]: (i) identificar a lista de símbolos relevantes que são as palavras ou frases peculiares e mais usadas; (ii) classificar os símbolos como: sujeito, objeto, verbo, e estado; (iii) descrever os símbolos através da noção e do impacto; (iv) verificar o LAL através de inspeção; e (v) validar o LAL com os atores do UDI.

Léxico Ampliado da Linguagem

(Regras Gerais)

	Noção	Impacto
Sujeito	Quem é o sujeito.	Quais ações executa.
Verbo	Quem realiza quando acontece e quais os procedimentos envolvidos.	Quais os reflexos da ação no ambiente (outras ações que devem ocorrer) e quais os novos estados decorrentes.
Objeto	Definir o objeto e identificar outros objetos com os quais se relaciona.	Ações que podem ser aplicadas ao objeto.
Estado	O que significa e quais ações levaram a esse estado.	Identificar outros estados e ações que podem ocorrer a partir do estado que se descreve.

Figura 3.3 – Regras gerais para definição de símbolos [C&L – PUC-Rio]

Durante os passos da atividade, siga as instruções que aparecem nas regras gerais fornecidas pela Figura 3.3 para a definição do nome, da noção e do impacto e obedeça aos princípios da circularidade e do vocabulário mínimo.

Para identificar a qual classe o símbolo pertence, em [Leite 00] são sugeridas as seguintes heurísticas: se o símbolo pratica uma ação, ele é classificado como sujeito; se é quem sofre a ação, é classificado como objeto; se o símbolo é uma situação em um dado momento, ele é classificado como estado; se representa uma ação, é classificado como verbo.

Resumindo:

- Os **sujeitos** praticam as ações.
- Os **objetos** sofrem as ações.
- Os **verbos** representam as ações.
- Os **estados** são situações.

1.B) Definir AGFL - Metas dos Agentes vindas do Léxico

No LAL os atores são descritos como sujeitos. Atores também aparecem nas noções e nos impactos que descrevem os símbolos. Os impactos mencionam as ações que acontecem na organização. Uma *ação concreta* muda um estado; muda o estado para outro diferente e uma *ação flexível* adiciona um atributo de qualidade ao estado.

Ações modificam estados. Então, o ponto-chave da atividade é identificar a motivação (o porquê) por trás de cada ação. (1) Quando a ação muda um estado → a ação vai definir uma meta concreta. (2) Quando a ação fornece uma “qualidade” a um estado → a ação vai definir uma meta flexível.

Três subatividades são requeridas neste ponto do método: Identificar os atores, Extrair as metas dos atores a partir dos impactos dos símbolos e Refinar as metas..

1.B.1) Identificar os atores

O objetivo da atividade é identificar os atores, sejam eles especializados em agentes, desempenhando papéis ou mesmo cobrindo posições [Leite 07].

Os atores são classificados como **sujeitos** no LAL porque praticam ações. Os símbolos classificados como **objetos** sofrem ações praticadas por atores. Os símbolos

do tipo sujeito e os praticantes das ações sobre os objetos são candidatos naturais a atores.

1.B.2) Extrair as metas dos atores a partir dos símbolos

O propósito da atividade é capturar as metas (concretas e flexíveis).

Para capturar as metas deve-se usar o “template” específico para cada tipo de ação definida no impacto do símbolo do LAL. Veja os exemplos: **Template 3.A** (caso de símbolo tipo sujeito em que o impacto menciona uma ação concreta), **Template 3.B** (caso de símbolo tipo objeto que sofre ação concreta) e **Template 3.C** (casos de ação flexível). Cada impacto irá gerar uma meta em um formato pré-estabelecido pelos templates. A Tabela 3.1 fornece o resumo da aplicação dos templates. Para definir a meta, o engenheiro de requisitos deve responder uma pergunta simples: POR QUÊ?.

Os impactos mencionam as ações que acontecem em uma organização e as ações mudam estados. As mudanças de estado são de duas modalidades:

- Ações mudam de um estado para outro estado, ou
- Ações adicionam um atributo de qualidade a um estado.

Essas heurísticas se baseiam na definição de Eric Yu [Yu 95], que diz que a intencionalidade dos atores pode ser modelada através de metas, e define meta como:

“Uma meta é uma condição ou estado desejado no mundo que um ator deseja alcançar”. [Yu 95]

Nossa idéia é simples: AÇÕES MUDAM ESTADOS E ESTADOS SÃO METAS.

A Tabela 3.1 apresenta a combinação das possíveis situações para a definição das metas. Na tabela, o tipo do símbolo combinado com os dois tipos de ações indica qual template é recomendado para a definição da meta.

Nas subseções a seguir ilustramos a definição das metas para todos os tipos de símbolos (sujeito, objeto, verbo e estado). Estes exemplos foram extraídos do estudo de caso EC – Expert Committee, apresentado no capítulo 4.

Tabela 3.1 – Aplicação dos templates na definição das metas

Símbolo	Impacto	Template - Meta
sujeito	ação concreta ação flexível	Template 3.A – meta concreta Template 3.C – meta flexível
objeto	ação concreta ação flexível	Template 3.B – meta concreta Template 3.C – meta flexível
verbo	ação concreta ação flexível	Template 3.C – meta concreta Template 3.C – meta flexível
estado	ação concreta ação flexível	Template 3.C – meta concreta Template 3.C – meta flexível

1.B.2.a) Definir metas a partir de símbolos tipo sujeito

As ações concretas provenientes de símbolos do tipo sujeito podem refletir duas possibilidades da motivação de um ator:

- um estado que o ator deseja e não depende de outro ator;
- um estado que o ator deseja e depende de outro ator.

Se a resposta ao “por quê” definir uma meta que deve ser alcançada com uma dependência para outro ator então, devido a essa dependência, mais uma meta deve ser definida, a meta do outro ator (caso da meta reflexiva). Nesse caso a resposta da análise do impacto indica que o primeiro ator (“o depender” – aquele que depende) depende ou necessita ajuda de outro ator (“o dependee” – daquele de que se depende) para que a meta seja alcançada. Esse tipo de ocorrência indica que uma “meta reflexiva” precisa ser definida. Essa “corrente” de metas concretas deve ser elicitada até aparecer uma meta sem a dependência de um segundo ator.

- **Instruções: Para cada símbolo do tipo sujeito:**
 - ♦ Para cada impacto que mencionar uma ação flexível: definir uma meta flexível com o **Template 3.C(2)**. Já que uma meta flexível está associada, por contribuição, a pelo menos uma meta concreta, nossa recomendação nesse caso é que a meta seja definida posteriormente, quando o engenheiro de requisitos for definir as metas provenientes de símbolos do tipo verbo.
 - ♦ Para cada impacto que mencionar uma ação concreta: definir uma meta concreta (**Template 3.A**) com quatro ou cinco componentes (ator, objeto, auxiliar, verbo e [ator]).

1. O primeiro ator já fica estabelecido pelo símbolo.
2. O item objeto será escolhido a partir de um símbolo (tipo: objeto ou sujeito) já existente no LAL.
3. O auxiliar que melhor se aplicar (seja(m) ou esteja(m)) será escolhido.
4. O engenheiro deve fornecer um verbo na voz passiva ou repetir um verbo dentre os já fornecidos.
5. Caso a meta dependa de um segundo ator, esse ator será escolhido a partir de um símbolo (tipo: sujeito) no LAL.

Heurísticas gerais: Sugestões para a definição das metas:

(1) use sempre o nome principal do símbolo na sentença de definição da meta, não utilize os seus sinônimos.

(2) crie metas com sentenças simples e diretas. Por exemplo: em vez de “processo de revisão seja executado” deve-se escrever “artigos sejam revisados”, porque a comunicação direta é mais fácil de ser compreendida.

Nome:	pesquisador
Noção:	Pessoas que estudam e propoem trabalhos ou artigos sobre uma area de interesse da conferencia .
Classificação:	sujeito
Impacto(s):	- submetem artigos . - participam da conferencia .
Sinônimo(s):	pesquisadores.

Figura 3.4 – Exemplo de um símbolo do LAL do tipo sujeito

Template 3.A – Usando as ações concretas de símbolos do tipo sujeito

TIPO: SUJEITO		<meta concreta>			ATOR	
-- impacto	resposta ao por quê?	<sujeito / objeto LAL>	seja / esteja	<verbo>		<sujeito LAL>
PESQUISADORES						
-- submetem	artigos.					
	Porque pesquisador deseja que	artigo	seja	publicado	por	chair
	Porque chair deseja que	artigo	seja	submetido	por	pesquisador
-- participam da	conferência.	ação flexível				

No exemplo da Figura 3.4, para o símbolo pesquisador, a ação (submetem) no primeiro impacto foi considerada concreta e a ação (participam) no segundo impacto foi considerada flexível; veja no Template 3.A. O primeiro impacto nos levou a identificar duas metas concretas enquanto o segundo impacto será trabalhado mais adiante no método.

No primeiro impacto: Pesquisador submete artigo pois deseja que artigo seja publicado por chair (pela conferência). Chair representa a conferência nesse relacionamento de cooperação. A segunda meta se deve ao relacionamento “reflexivo” entre os dois atores e está explicada no parágrafo a seguir.

1.B.2.b) Definir metas reflexivas (o outro lado da dependência)

As metas reflexivas refletem a interação e a colaboração entre dois atores. Na etapa de modelagem, o método resolverá a dependência entre os atores com as propriedades de SMA (propriedades de agência).

- **Instruções: Para cada meta concreta em que um segundo ator foi indicado:**
- ♦ Definir uma nova meta concreta que complemente a dependência entre os atores.
 1. Continuar até que na definição da meta não mais exista um segundo ator ou até que surja uma dependência mútua.
 2. Criar a meta reflexiva com base nos passos anteriores de 1 até 5, como são elaboradas para símbolos do tipo sujeito. No exemplo anterior: pesquisador deseja que artigo seja publicado por chair e chair deseja que artigo seja submetido por pesquisador.

1.B.2.c) Definir metas a partir de símbolos tipo objeto

Cada impacto que menciona uma ação sofrida por um símbolo do tipo objeto indica um estado que algum ator deseja atingir.

- **Instruções: Para cada símbolo do tipo objeto:**
- ♦ Para cada impacto que mencionar uma ação flexível: definir uma meta flexível com o **Template 3.C(2)**. Pois uma meta flexível está associada (por contribuição) a pelo menos uma meta concreta. Nossa recomendação nesse caso é que a meta seja definida posteriormente, quando o engenheiro for definir as metas provenientes de símbolos do tipo verbo.
- ♦ Para cada impacto que mencionar uma ação concreta: definir uma meta concreta (**Template 3.B**) com quatro componentes (objeto, auxiliar, verbo e ator). O template usado para símbolos do tipo objeto é diferente daquele usado para símbolos do tipo sujeito porque objetos sofrem as ações.

1. O item objeto deverá ser um símbolo (tipo: objeto ou sujeito) já existente. Pode ser o próprio objeto da ação.
 2. Engenheiro escolhe o auxiliar que melhor se aplicar (seja ou esteja).
 3. Engenheiro deve fornecer um novo verbo na voz passiva ou repetir um verbo dentre os já escolhidos.
 4. O item ator será escolhido a partir de um símbolo tipo sujeito já existente no LAL.
- ♦ O objeto mencionado na heurística (1) deve ser aquele que possui “maior proximidade” do objeto que está gerando a meta. Ou seja, evite escolher um objeto que indique uma meta de longo prazo. A heurística orienta para que se evite a tendência de definir metas de longo prazo. As metas de longo prazo, por serem mais genéricas e mais abstratas, são menos questionadas porque sempre são aplicáveis. As metas de curto prazo elicítadas através do LAL darão a consistência necessária e “corpo situacional” para que as situações de dependência estratégica sejam identificadas na etapa seguinte.

Nome:	conferencia
Noção:	Encontro periódico de pesquisadores experts em uma área de pesquisa.
Classificação:	objeto
Impacto(s):	- pesquisadores participam da conferência. - autores submetem artigos .
Sinônimo(s):	simpósio, congresso.

Figura 3.5 – Exemplo de um símbolo do LAL do tipo objeto

Template 3.B – Usando as ações concretas de símbolos do tipo objeto

TIPO: OBJETO	<meta concreta>			ATOR
	<sujeito / objeto LAL>	seja/ esteja	<verbo>	
-- impacto resposta ao por quê?				
CONFERÊNCIA				
-- pesquisadores participam da conferência.	ação-flexível			
-- autores submetem artigos.				
Para que	artigos	sejam	revistos	por revisor

No exemplo da Figura 3.5, para o símbolo conferência, a ação (participam) no primeiro impacto foi considerada flexível e a ação (submetem) no segundo impacto foi considerada concreta. O segundo impacto nos levou a identificar uma meta concreta enquanto o primeiro impacto será trabalhado mais adiante no método.

1.B.2.d) Definir metas a partir de símbolos tipo verbo

Cada impacto que mencionar uma ação referenciada por um símbolo do tipo verbo indica um estado que algum ator deseja atingir.

- **Instruções: Para cada símbolo do tipo verbo:**

- ♦ Para cada impacto que mencionar uma ação concreta: definir uma meta (**Template 3.C(1)**) com quatro componentes (objeto, auxiliar, verbo e ator).
- ♦ Para cada impacto que mencionar uma ação flexível: gerar uma meta flexível (**Template 3.C(1)**), que deve ser definida com três atributos (tipo, tópico e meta concreta associada).
 1. Fornecer o tipo (atributo de qualidade) ou repetir um entre os fornecidos.
 2. O tópico será um dos símbolos (tipo: objeto ou sujeito) existentes no léxico.
 3. Escolher uma meta concreta entre uma das metas já definidas.
- ♦ A meta concreta mencionada na heurística (3) deve ser aquela que possuir “menor proximidade” do objeto que está gerando o tópico da meta flexível, ou seja, a melhor escolha será pela meta de mais longo prazo que se aplicar. De modo inverso ao da heurística (4) do item anterior, a orientação é fazer uma definição de metas flexíveis muito pouco específicas e por isso orientamos que seja escolhida uma meta concreta de mais longo prazo para ser associada à meta flexível, de modo que a meta flexível possa qualificar um número maior de metas concretas.

Nome:	rever artigo
Noção:	Trabalho de identificar erros e sugerir correções em artigos submetidos a conferencia .
Classificação:	verbo
Impacto(s):	- é executada por revisores . - deve ser obedecido o deadline de revisão .
Sinônimo(s):	revisões, revisão.

Figura 3.6 – Exemplo de um símbolo do LAL do tipo verbo

Template 3.C(1) – Usando símbolos do tipo verbo

TIPO: VERBO	<meta flexível>		<meta concreta associada>	<ator>
-- impacto	<TIPO> atributo de qualidade	[TOPICO] sujeito/objeto LAL		
resposta ao por quê?				
REVER ARTIGO				
-- é executada por revisores.	ação concreta		artigo seja revisado	revisor
-- deve ser obedecido o deadline de revisão.				
Porque	sem atraso	[revisão]	artigo seja revisado	revisor

No exemplo (Figura 3.6 e **Template 3.C(1)**) para o símbolo “rever artigo” a ação (executar) no primeiro impacto foi considerada concreta e a ação (deve obedecer) no segundo impacto foi considerada flexível. O primeiro impacto nos levou a identificar uma meta concreta enquanto o segundo impacto, uma meta flexível. A revisão dos artigos deve obedecer ao prazo para que não haja atraso na apuração do resultado das revisões. A restrição gerou a meta flexível **sem atraso [revisão]**. Essa meta flexível foi associada à meta concreta do revisor “*artigo seja revisado*”.

Template 3.C(2) – Ações flexíveis de símbolos tipo sujeito e tipo objeto

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

Como pode ser observado no **Template 3.C(2)**, depois que a maioria das metas concretas foi elicitada, as metas flexíveis provenientes de símbolos tipo sujeito, tipo objeto, tipo verbo e tipo estado podem ser elicitadas. Este ponto é considerado mais adequado porque as metas flexíveis precisam ser associadas a metas concretas. Então, metas concretas devem ser descobertas antes.

1.B.2.e) Definir metas a partir de símbolos tipo estado

Cada impacto que mencionar uma ação provocada por um símbolo do tipo estado indica um estado que algum ator deseja atingir.

- **Instruções: Para cada símbolo do tipo estado:**
 - ♦ Para cada impacto que gerar uma meta concreta ou uma meta flexível, o engenheiro de requisitos escolhe a melhor dentre as duas alternativas:
 - Aplicar as heurísticas do passo (1.B.2.c) para o caso de meta concreta.
 - Aplicar as heurísticas do passo (1.B.2.d) para o caso de meta flexível.

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 3.7 – Exemplo de um símbolo do LAL do tipo estado

Template 3.C(3) –Usando símbolos do tipo estado

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

O exemplo composto pela Figura 3.7 e pelo **Template 3.C(3)** mostra que um símbolo tipo estado pode gerar tanto uma meta flexível quanto uma meta concreta, dependendo da natureza da ação tratada no impacto do símbolo. No caso, a ação (precisa receber outra avaliação) foi considerada flexível devido à necessidade de *melhor qualidade* na revisão de artigos, quando existe um conflito entre duas revisões.

1.C) Refinar as metas

O objetivo da atividade é agrupar as metas (concretas e flexíveis) por ator e organizá-las em ordem cronológica.

- **Instruções:**

1. Converter as metas tipo objeto em metas tipo sujeito.
 - Como o ator de cada meta oriunda de um símbolo do tipo objeto ficou no final do template, a tarefa é apenas remanejar o ator para o início da meta. A Figura 3.8 mostra o resultado da operação sobre a meta apresentada no Template 3.B.

DEPENDER					DEPENDEE
revisor					
Porque revisor deseja que	artigo	seja	revisto		

Figura 3.8 –Resultado da conversão de meta tipo objeto em meta tipo sujeito

2. Agrupar as metas por ator.
 - Colocar juntas as metas que pertençam ao mesmo ator.
 - Unir também as metas de agentes, posições e papéis que sejam referentes ao mesmo ator.
3. Ordenar temporalmente as metas.

- Iniciar com as metas que representem os primeiros estados alcançados ou modificados e depois colocar as metas em ordem cronológica.
 - Ordenar as metas a partir da identificação da seqüência temporal dos estados atingidos. Colocar as metas flexíveis associadas próximas às metas concretas.
 - Colocar as metas de longo prazo no final. Metas de longo prazo apontam para resultados gerais ou resultados de sobrevivência do negócio.
4. Excluir as metas redundantes e/ou repetidas.
- Apagar as metas de agentes, posições e papéis que estejam repetidas para o mesmo ator.
 - Atenção, não apagar uma meta quando esta contiver um verbo sinônimo de outra meta, pois pode estar indicando um ponto no tempo de alcance diferente.

DEPENDER					DEPENDEE
autor					
	artigo	seja	submetido		
justa [revisão]	artigo	seja	revisado	por	revisor
	artigo	seja	aceito		chair
	camera-ready	seja	enviada		
	artigo	seja	publicado	por	chair

Figura 3.9 – Exemplo com metas concretas, metas flexíveis e dependências

A Figura 3.9 traz o resultado final da atividade de refinar as metas para o ator autor como exemplo. Há cinco metas concretas e uma meta flexível, e também está explicitado que duas metas do autor dependem de outros atores.

- O Diagrama de Classes da AGFL

O diagrama de classes da Figura 3.10 mostra as classes de objetos com os relacionamentos trabalhados para a definição de metas AGFL. As classes em tom cinza são provenientes do Léxico e as demais são trabalhadas pelo AGFL.

Com o objetivo de potencializar o rastreamento das metas, o diagrama de classes da AGFL contempla os atores e as metas (concretas e flexíveis), além do rastro do histórico que gerou cada meta. Significa dizer que: (a) para cada meta podem ser identificados os impactos que a gerou e os símbolos de origem. Além do rastro, o modelo permite que o “rationale” seja anotado na classe de relacionamento “Rationale”,

onde o engenheiro de requisitos pode documentar os motivos para determinada decisão. O diagrama da Figura 3.10 faz parte da “baseline” que aparece na parte central do diagrama simplificado do Método ERi*c (Figura 3.1), pois é um elemento de apoio à rastreabilidade da elicitação das metas.

Descrição dos elementos do diagrama:

Cada símbolo do léxico (SimboloLAL) corresponde a um e somente um símbolo AGFL e vice-versa. Cada símbolo AGFL (AGFLsymbol) é composto por um ou mais (1..*) impactos (AGFLbr) e um impacto constrói uma ou mais metas.

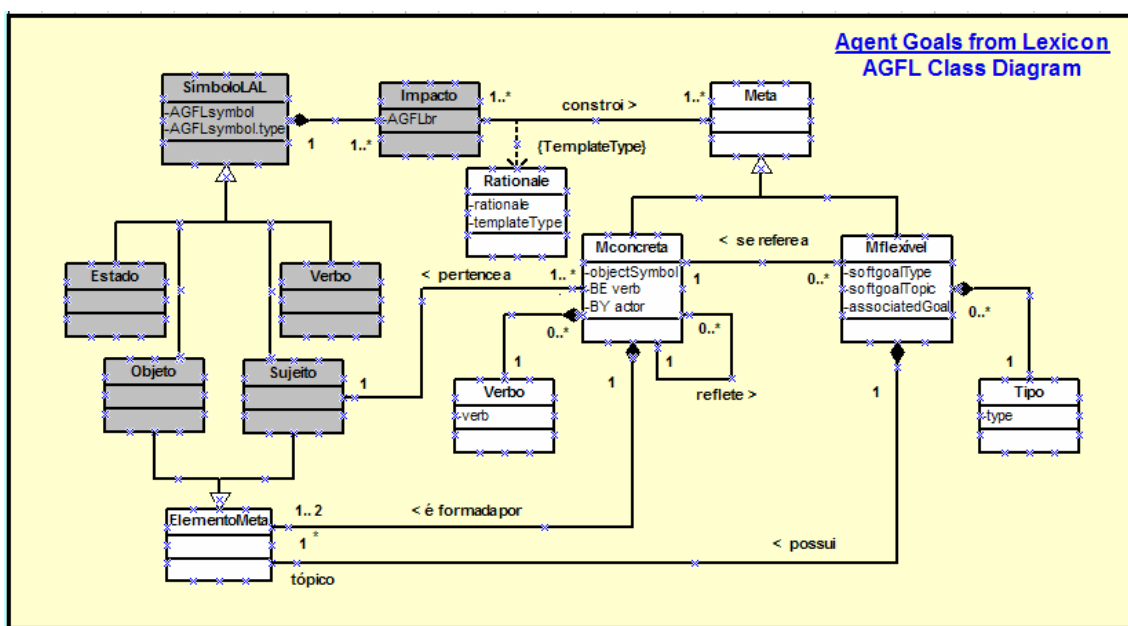


Figura 3.10 - Diagrama de classes da AGFL

Uma meta pode ser de dois (2) tipos: meta concreta (Mconcreta) ou meta flexível (Mflexível). Quando a ação contida no impacto do símbolo for uma ação concreta, esta vai construir uma meta concreta, em caso contrário construirá uma meta flexível. A escolha pode ser registrada através da classe Rationale. Uma meta flexível possui dois componentes: o tipo, que é um atributo de qualidade, e o tópico, que é o item a que a qualidade se aplica. Uma meta flexível está sempre associada a uma meta concreta e uma meta concreta pode estar associada a zero ou mais metas flexíveis. Uma meta concreta pertence a um ator, que é um sujeito do Léxico. Ela possui dois componentes obrigatórios e um componente alternativo. Obrigatoriamente uma meta concreta é definida por um verbo (conjugado na voz passiva) e por um objeto ou um sujeito do Léxico. Como componente alternativo, a meta concreta pode ter outro ator, o qual é um

sujeito do Léxico, no caso de a meta ser alcançada por esse ator para o proprietário da meta.

2) Identificar as Situações de Dependência Estratégica

Objetivo: “Refinamento e organização das metas de SMAs”.

A segunda etapa, Identificar as Situações de Dependência Estratégica, é composta por três atividades: A) Distinguir SDSituations, B) Reconhecer as interdependências entre as SDSituations e C) Construir diagrama de SDSituations, veja Figura 3.11.

Para o entendimento das atividades foi apresentada, no capítulo 2, a conceituação de Situações de Dependência Estratégica – SDSituations [Oliveira 06b], e os detalhes das atividades são apresentados posteriormente.

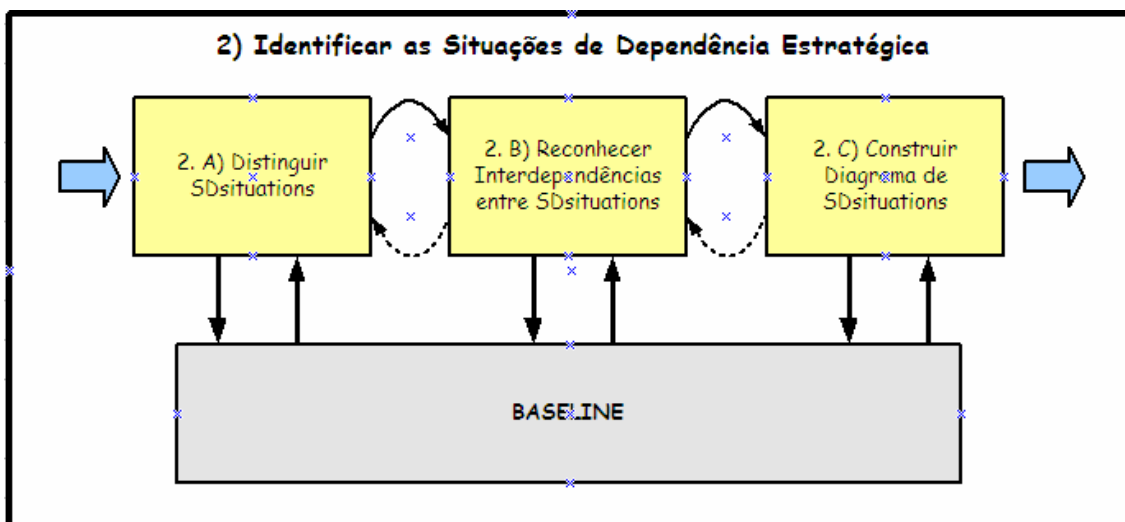


Figura 3.11 – Detalhamento da etapa 2 do Método ERi*c

2.A) Distinguir SDSituations

Arranjar as metas por situações de dependência estratégica. Os atores participam através das metas próprias de uma situação de colaboração bem definida chamada “Situação de Dependência Estratégica”.

Na atividade trabalha-se com o objetivo de perceber como as metas compõem conjuntos para formar situações de negócio. Sabemos que atores possuem metas, mas elas são fortemente ligadas ou associadas entre si de modo a concluir uma situação de negócio, uma situação estratégica de cooperação entre os atores, uma SDSituation. O

engenheiro de requisitos deve perceber que dentro de cada SDSituation não existe meta com fraca conectividade às demais metas. Em uma mesma SDSituation a conectividade entre as metas não deve ser temporal, mas elas precisam de uma finalidade comum para que permaneçam juntas.

Nas SDSituations percebe-se com mais clareza que os atores podem ter especializações como agentes, papéis e posições [Cunha 07] e [Leite 07].

A Tabela 3.2 mostra as sintaxes para a descrição dos elementos da SDSituation e as funções dos atributos na SDSituation. O nome de uma SDSituation deve ser escrito com a sintaxe: < **substantivo da situação + objeto** >, por exemplo: “Indicação de Revisores”.

Adotamos como guia a mesma sintaxe usada por [Zheng 04] para elementos do i*.

Tabela 3.2 – Descrevendo os elementos das SDSituations

	META CONCRETA	META FLEXÍVEL	TAREFA	RECURSO
NOME <sintaxe>	Substantivo + seja + verbo na passiva	Atributo de qualidade [tópico]	Verbo + objeto	Substantivo
DEFINIÇÃO	Sentença simples descrevendo a meta concreta.	Sentença simples descrevendo a meta-flexível.	Sentença simples descrevendo a tarefa.	Sentença simples descrevendo o recurso.
CONSEQÜÊNCIAS	Sentença simples	Sentença simples	Sentença simples	Sentença simples

2.B) Reconhecer Interdependências entre SDSituations

Observe cada SDSituation e reconheça situações em que há dependência lógica, temporal ou seqüencial com outra situação. Em outras palavras, você deve identificar situações distintas, de maneira que uma dependa da outra no processo do negócio. Reconheça também se existe algum paralelismo entre as SDSituations.

2.C) Construir Diagrama de SDSituations

Represente todas as SDSituations em um único diagrama [Oliveira 06b]. Mostre as interdependências entre elas e represente o encadeamento das SDSituations. Coloque a representação do fator tempo no diagrama quando for importante. A Figura 3.12 ilustra como deve ser o Diagrama de SDSituations.

Represente todas as SDSituations em um único diagrama [Oliveira 06b]. Mostre as interdependências entre as SDSituations e represente o encadeamento das SDSituations.

Coloque a representação do fator tempo no diagrama quando for importante. A Figura 3.12 ilustra como deve ser o Diagrama de SDSituations.

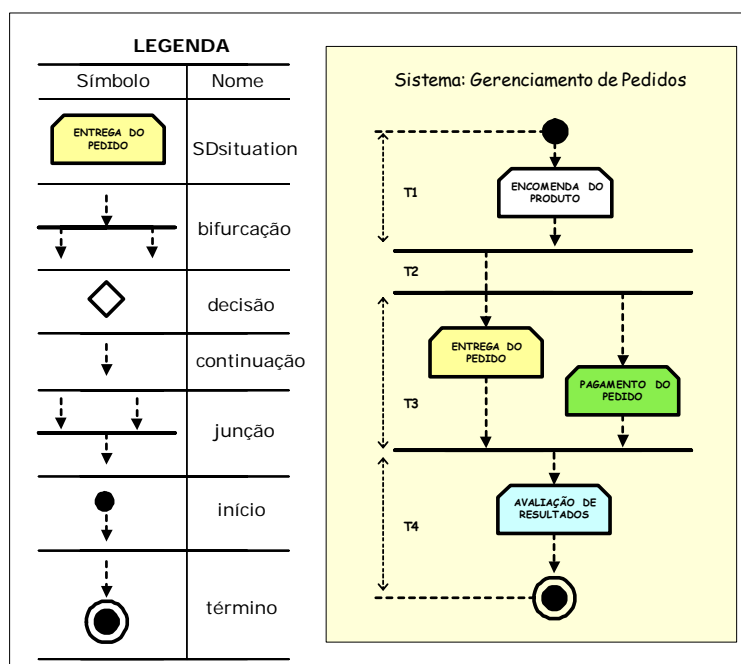


Figura 3.12 – Ilustração: Exemplo do Diagrama de SDSituations

No exemplo da Figura 3.12, do Sistema de Gerenciamento de Pedidos, o encadeamento das SDSituations é iniciado com a ENCOMENDA DO PRODUTO. A SDSituation ENTREGA DO PEDIDO começa logo depois, e o PAGAMENTO DO PEDIDO tem início após um tempo T2 e há um tempo T3 para o encerramento. A última SDSituation, AVALIAÇÃO DE RESULTADOS, só é iniciada depois da conclusão da entrega do pedido e de seu pagamento, e possui o limite de tempo T4.

3) Modelar as Metas dos Atores

Objetivo: “Modelagem e avaliação das metas”

A terceira etapa é composta por duas atividades: A) Identificar Agentes, Posições e Papéis e B) Criar os Painéis de Intencionalidade, veja Figura 3.13.

Para o entendimento das atividades, como fizemos na seção anterior, foi apresentada, no capítulo 2, a conceituação necessária para a elaboração dos Painéis de Intencionalidade [Oliveira 07] com a representação das relações entre as metas.

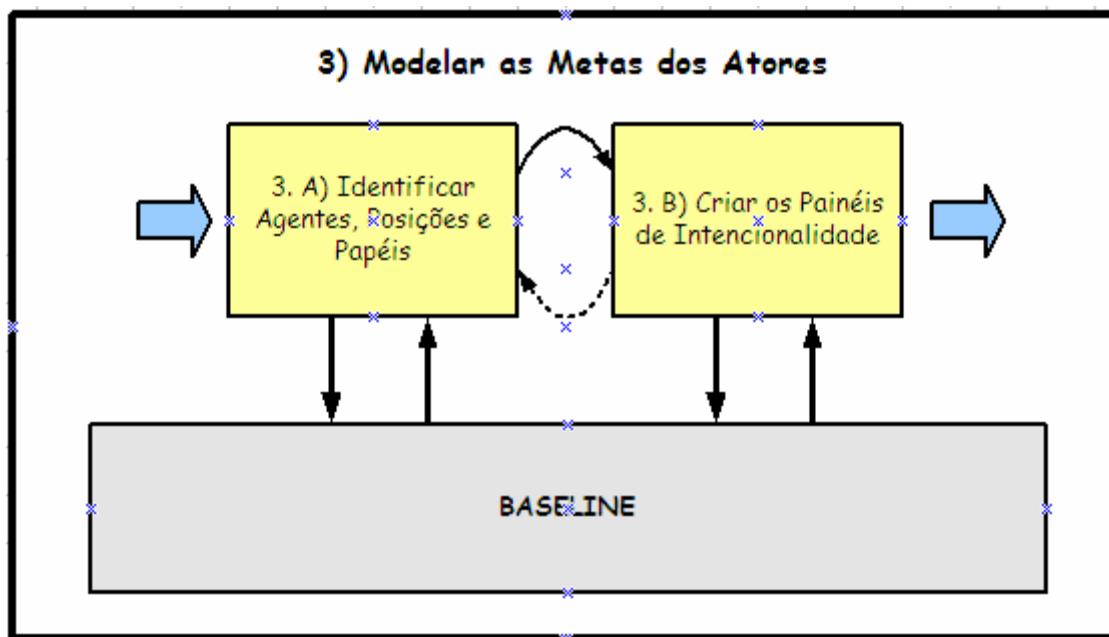


Figura 3.13 – Detalhamento da etapa 3 do Método ERi*c

A principal motivação para a criação do Diagrama IP é a representação da intencionalidade em um único e homogêneo diagrama. A intencionalidade elicitada através das metas dá origem aos requisitos do sistema. O Diagrama IP retrata para os atores apenas as metas e as relações de transição entre as mesmas, podendo ser considerado uma redução do Modelo SR. O Diagrama IP, além de não representar as tarefas e os recursos que aparecem no Modelo SR, adota também uma simplificação em relação ao diagrama de estados tradicional. Essa simplificação ocorre porque a notação da transição de estados foi suprimida pela adoção de uma convenção que retrata apenas três tipos de transição (a correlação, a contribuição e a dependência). As relações entre as metas são semanticamente as mesmas adotadas pelo Modelo SR [Yu 95]; veja a contextualização na seção a seguir.

3.A) Identificar Agentes, Posições e Papéis

Como em cada SDSituation os atores trabalham e colaboram mutuamente para a resolução da meta da situação identificada, eles podem assumir posições e desempenhar papéis pertinentes a cada SDSituation. Por serem situações guiadas pela intencionalidade comum, as SDSituations agregam as metas dos atores dedicadas à resolução da meta comum aos envolvidos na situação de dependência estratégica. Como, com a abordagem de SDSituations, o Engenheiro de Requisitos lida com um

número reduzido de metas por vez e, além disso, elas são restritas à situação, ele pode perceber com mais clareza quando os atores têm algum comportamento que pode ser classificado como especialização do ator, como agente, papel ou posição.

Utilizando as metas concretas elicitadas para cada ator, observe e tente perceber se existe algum papel ou posição que pode ser identificada. Metas exclusivas de um ator e inalcançáveis por alguém mais podem indicar um papel desempenhado pelo ator. Um ou mais papéis podem estar associados a um cargo coberto (posição) pelo agente na organização.

Enquanto os papéis geralmente são restritos a uma única SDsituation, uma posição coberta por um agente pode aparecer ou se repetir em mais de uma SDsituation.

3.B) Criar os Painéis de Intencionalidade

Prepare um diagrama para cada SDsituation. Dê preferência a colocar como vizinhos no diagrama, ou alocar o mais próximo possível quando existirem mais de dois atores, atores ou agentes que tenham o maior número de relações de dependência. Decida sobre a ordenação das metas no eixo do ator, sendo que as metas iniciais devem ficar na parte inferior do diagrama. Metas concretas e metas flexíveis que necessitem aguardar o alcance de outro estado devem ser colocadas na parte superior do eixo do ator. Primeiramente, coloque apenas as metas concretas e em segundo lugar, as metas flexíveis. O exemplo de Diagrama IP na Figura 3.14 mostra como é feito o detalhamento de uma SDsituation.

Trabalhe com um ator de cada vez. Primeiro, represente as relações entre as metas flexíveis (a contribuição de uma meta flexível para outra meta flexível), depois as relações entre as metas flexíveis e as metas concretas (a correlação de uma meta flexível para uma meta concreta e a correlação de uma meta concreta para uma meta flexível). Por último, represente as relações de dependência entre as metas concretas dos atores, iniciando a representação pela parte inferior do diagrama. Faça uma revisão, e se metas de atores diferentes forem equivalentes e atingidas simultaneamente, represente a relação de equivalência entre elas.

O Diagrama IP é formado pelos atores e pelas metas desses atores. Cada meta de um ator é representada em apenas um diagrama enquanto os atores podem aparecer em diversos diagramas.

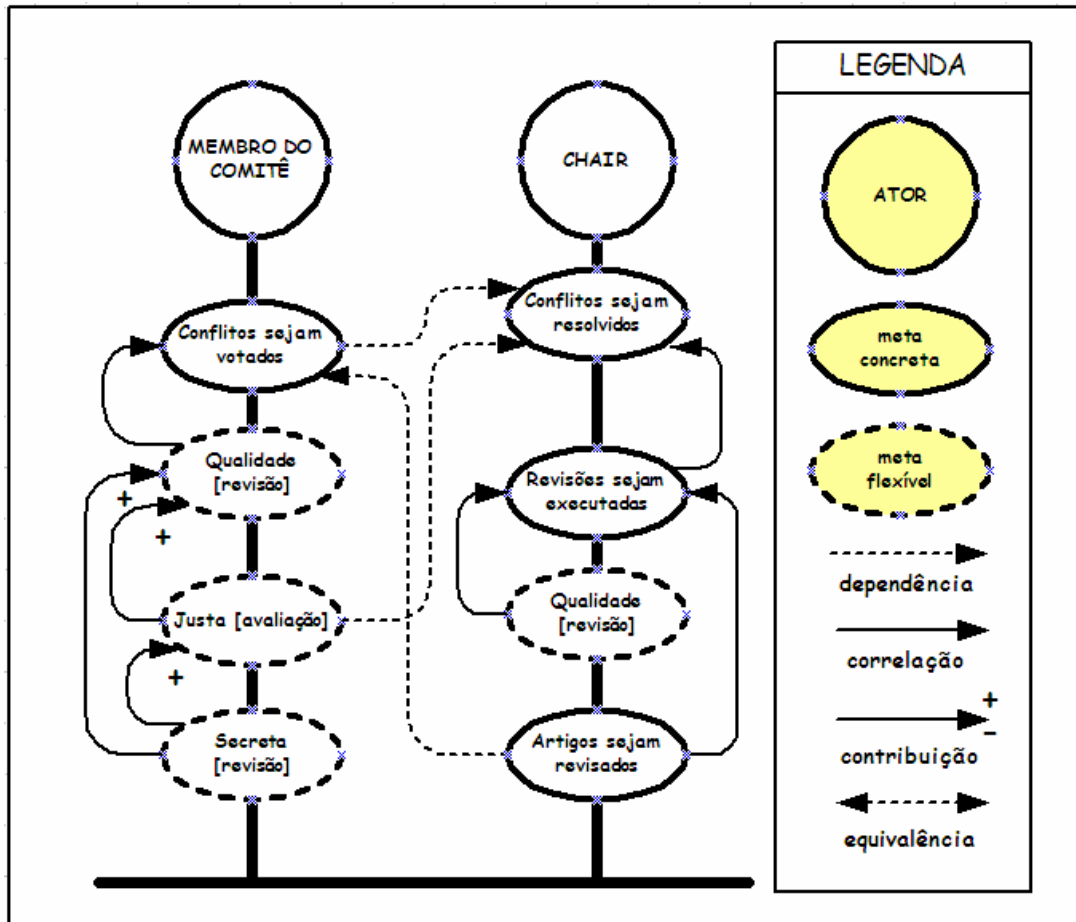


Figura 3.14 – Exemplo de Diagrama IP – SDsituation: VOTAÇÃO DE CONFLITOS

No diagrama da Figura 3.14 o ator MEMBRO DO COMITÊ possui quatro metas, que estão temporalmente ordenadas sobre o eixo do ator. A meta concreta “Conflitos sejam votados” recebe a correlação da meta flexível “Qualidade [revisão]” e esta recebe contribuições de outras duas metas flexíveis. A meta concreta “Conflitos sejam resolvidos” do ator CHAIR possui duas dependências com o ator MEMBRO DO COMITÊ e este depende do CHAIR para que “Artigos sejam revisados”. As oito metas dos atores e as relações entre as mesmas fazem parte da SDsituation Votação de Conflitos.

No fim da etapa, após identificar e modelar as metas e as SDsituations, desejamos suprir uma questão que frequentemente aparece em projetos de Engenharia de Software: Existe alguma medida para o método que possa avaliar a complexidade do sistema que estamos modelando? Logo no início, na fase “early”?

Quando concluímos a elicitação das metas poderíamos simplesmente contar todas as metas elicidadas e assim chegaríamos a uma medida muito simples: o número de

metas (concretas e flexíveis). Porém essa medida é insuficiente e incompleta porque considera somente as metas individuais e despreza a interdependência entre as mesmas.

Devido à natureza do problema representado, que as metas são estados interconectados e interdependentes formando uma rede de nós e arcos, acreditamos que a medida da Complexidade Ciclômática de McCabe [McCabe 76] é adequada para avaliar a complexidade dos modelos com que estamos lidando. Avaliamos dessa maneira porque é possível observar que a fórmula da Complexidade Ciclômática de McCabe considera os nós e subtrai os arcos. Essa visão é apropriada para o esquema de dependências estratégicas do Framework i^* porque a meta de um ator pode ser alcançada por outro ator. Além dessa perspectiva, a medida considera apropriadamente os atores envolvidos nas SDsituations, ou melhor, considera e aplica um peso positivo para o aumento da medida da complexidade devido ao envolvimento de atores. Veja o exemplo da aplicação da medida na seção a seguir.

A Aplicação da Complexidade Ciclômática

A métrica conhecida como Complexidade Ciclômática ou indicador de McCabe avalia a complexidade de um grafo através de uma equação que algebricamente contabiliza o número de arestas e nós do grafo avaliado [McCabe 94]. Esse indicador é referenciado e utilizado na subárea de testes de software quando se avalia a cobertura de casos de teste em módulos de software após a confecção de um grafo de complexidade equivalente. McCabe [McCabe 76] define o número ciclômático $V(G) = a - n + p$ para um grafo G com n nós (ou vértices), a arestas (ou arcos) e p componentes conexos.

Aplicando para o exemplo da Figura 3.14:

$n \rightarrow$ corresponde ao número de metas = 8

$a \rightarrow$ corresponde ao número de relações = 9

$p \rightarrow$ número de atores = 2

$V(G) = 9 - 8 + 2 = 3$.

4) - Modelar a Racionalização das Metas dos Atores

Objetivo: Modelagem do “rationale” dos atores (agentes, posições e papéis).

A quarta etapa é composta por duas atividades: a) Construir Modelos SD e b) Construir Modelos SR, veja Figura 3.15.

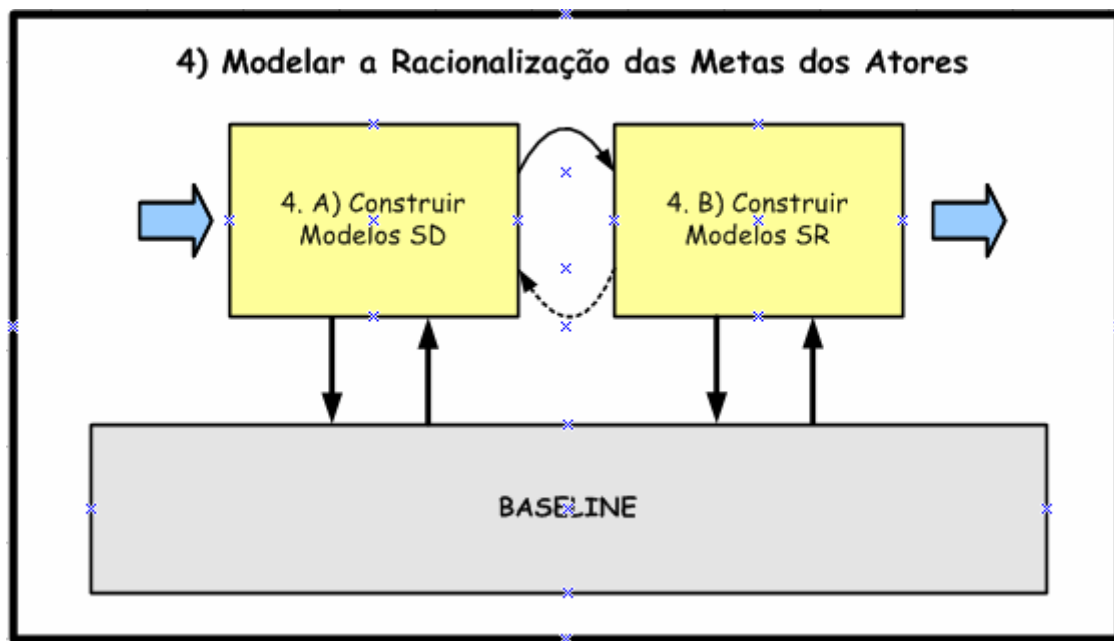


Figura 3.15 – Detalhamento da etapa 4 do Método ERi*c

4.A) Construir Modelos SD

Para a construção dos Modelos SD o engenheiro de requisitos precisa utilizar artefatos preparados nas fases anteriores do Método ERi*c. São necessários o Diagrama de SDsituations e os Diagramas IP (Painéis de Intencionalidade) de cada SDsituation identificada.

4.A.1) Definir as dependências estratégicas

O objetivo da atividade é definir as dependências estratégicas entre os atores usando os mesmos critérios de [Yu 95]. Para cada SDsituation do Diagrama de SDsituations, o engenheiro de requisitos deve observar as relações de dependência assinaladas no Diagrama IP, e para cada uma delas deve definir entre os quatro tipos de dependência possíveis (por meta concreta, por meta flexível, por recurso e por tarefa) qual deles é o mais indicado e vantajoso para o “dependee”.

Cada tipo de dependência tem um grau de liberdade diferente do “dependee” na relação de colaboração entre os atores. Na dependência por recurso o grau de liberdade do “dependee” é nulo, pois ele deve disponibilizar o recurso exatamente como é

esperado pelo “depender”. Nas duas dependências por meta o “depender” fica mais vulnerável, pois não tem influência sobre o cumprimento da meta pelo “dependee”. Na dependência por meta flexível, o “depender” avalia e aceita, se achar conveniente, que a meta flexível foi razoavelmente satisfeita (“satisficed”¹). Na dependência por tarefa, o “dependee” executa uma tarefa seguindo as orientações definidas pelo “depender”. Ainda na dependência por tarefa, o “dependee” deve ter alguma motivação (ou compensação) para executar a tarefa da maneira que o “depender” deseja e essa motivação deve ser mostrada no Modelo SD, na SDsituation atual, como é mais comum, ou em outra SDsituation.

Relação de colaboração entre os atores	Dependência recomendada	Grau de liberdade
O “depender” não exerce qualquer influência sobre o que é feito pelo “dependee”.	meta concreta	pleno
O “dependee” presta o serviço para o “depender” e obedece sempre os critérios por ele definidos.	tarefa	parcial
O “dependee” disponibiliza o recurso (material ou informação) desejado para o “depender”.	recurso	não existe
O “depender” não exerce influência sobre o “dependee” e o “depender” tem a decisão final de aceitar ou não.	meta flexível	pleno mas avaliável

Figura 3.16 – Regras gerais para definição de dependências estratégicas

Para todos os tipos de dependências, deve existir uma motivação para o cumprimento da dependência estratégica pelo “dependee”, pois se trata de um modelo de cooperação guiado pela intencionalidade dos atores. No entanto, a motivação pode não se tornar explícita pelo processo de elicitación.

4.A.2) Preparar um Modelo SD para cada SDsituation

A atividade é simples: fazer um Modelo de Dependências Estratégicas – Modelo SD para cada conjunto de atores participantes da SDsituation, usando as dependências estratégicas definidas na atividade anterior.

HEURÍSTICAS PARA DERIVAR MODELOS SD:

- i) Represente os atores aplicando as especializações percebidas na SDSituation.
 - Represente um ator pelos papéis desempenhados e posições ocupadas, pois assim há maior representatividade do detalhamento do processo de elicitação.
- ii) Modele as dependências entre os atores.
 - Utilize o Modelo SD para expressar os relacionamentos intencionais. Use as “Regras gerais para definição de dependências estratégicas” da Figura 3.16.
 - Represente no Modelo SD as metas (concretas e flexíveis), as tarefas e os recursos importantes para cada SDSituation. Use a padronização de nomes da Tabela 3.2.

4.B) Construir Modelos SR

Para a construção dos Modelos SR o engenheiro de requisitos precisa dos Modelos SD construídos na atividade anterior além dos artefatos preparados nas fases anteriores do Método ERi*c. São necessários o Diagrama de SDSituations e os Diagramas IP (Painéis de Intencionalidade) de cada SDSituation identificada.

HEURÍSTICAS PARA DERIVAR MODELOS SR:

O processo de modelagem MAS deve lidar com três (3) tipos de interações adicionais, veja Figura 3.17. Existem três novas interações (Ator ↔ Agente, Agente ↔ Sistema, e Agente ↔ Agente) em comparação à modelagem tradicional de software (Ator ↔ Sistema) [Jennings 01] [Silva 04]. Jennings [Jennings 01] chama de ambiente (“environment”) o conjunto de todos os agentes e recursos compartilhados entre os agentes e os atores. Para gerenciar e controlar a complexidade, como também explorar o uso adequado das propriedades de sistemas multi-agentes, recomendamos a elaboração de quatro (4) Modelos SR em vez de modelar todas as interações em um único diagrama [Oliveira 06a]. A técnica está apresentada em detalhes nas subseções i), ii) e iii).

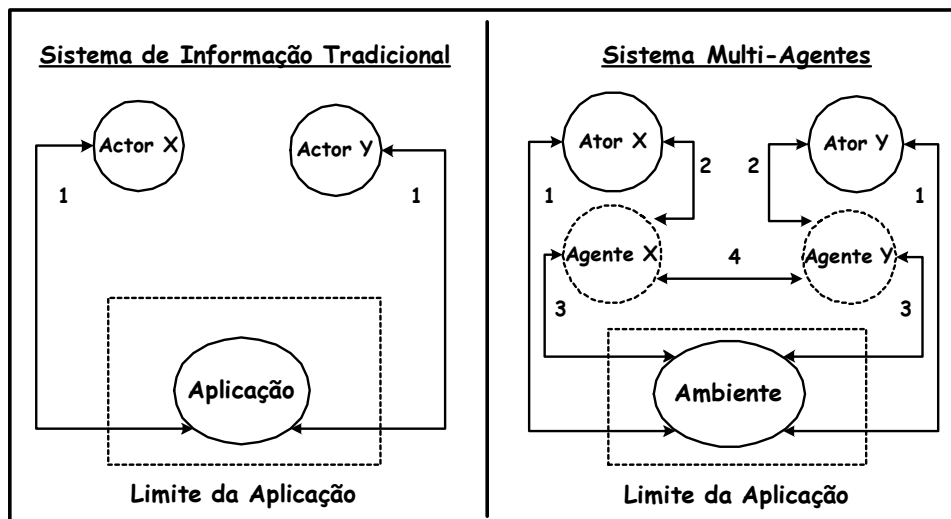


Figura 3.17 – Ilustração comparativa dos tipos de interações que MAS deve lidar

i) MODELAR AS INTERAÇÕES: ATOR – ATOR

(Este modelo é a base para criar os outros modelos – exemplo Figura 3.18).

- Para cada SDsituation crie um Modelo SR para expressar como a Situação de Dependência Estratégica é resolvida por cada ator participante. Represente dentro da linha-limite de cada ator as metas concretas, as metas flexíveis, os recursos e as tarefas como elementos para cumprir a intencionalidade do ator. Conecte esses elementos usando as ligações intencionais (meios-fim, decomposição e contribuição) [Yu 95].
- Represente as dependências entre os atores que aparecem no Modelo SD da SDsituation também no Modelo SR; os elementos da dependência aparecem fora das linhas-limite dos atores, mas no Modelo SR as ligações aparecem unindo elementos que estão dentro da linha-limite de cada ator.
- Identifique as metas principais para cada ator. Elas são as metas que aparecem no Diagrama IP no topo de cada eixo do ator. Atores podem possuir mais de uma meta principal e freqüentemente possuem uma ou mais metas intermediárias. O Diagrama IP fornece a hierarquia das metas e o engenheiro de requisitos deve representar essa hierarquia no Modelo SR.
- Represente as tarefas em ligações meios-fim de modo a cumprir o atingimento das metas concretas. Soluções alternativas devem ser mapeadas, assim como tarefas meio, se existir mais de um meio de alcançar a meta concreta.

- Represente as tarefas adicionais, ou especializadas, tipo subtarefas, que devem ser ligadas às tarefas (meio) usando a ligação de decomposição.
- Identifique os recursos necessários às tarefas assim também os recursos a serem compartilhados por tarefas e subtarefas.
- Mapeie as metas flexíveis conectando-as às tarefas quando adicionarem qualidade à intencionalidade mapeada pela meta concreta, ou conectando-as às outras metas flexíveis, usando a ligação de contribuição.
- Para nomear os elementos do modelo, use a padronização da Tabela 3.2.

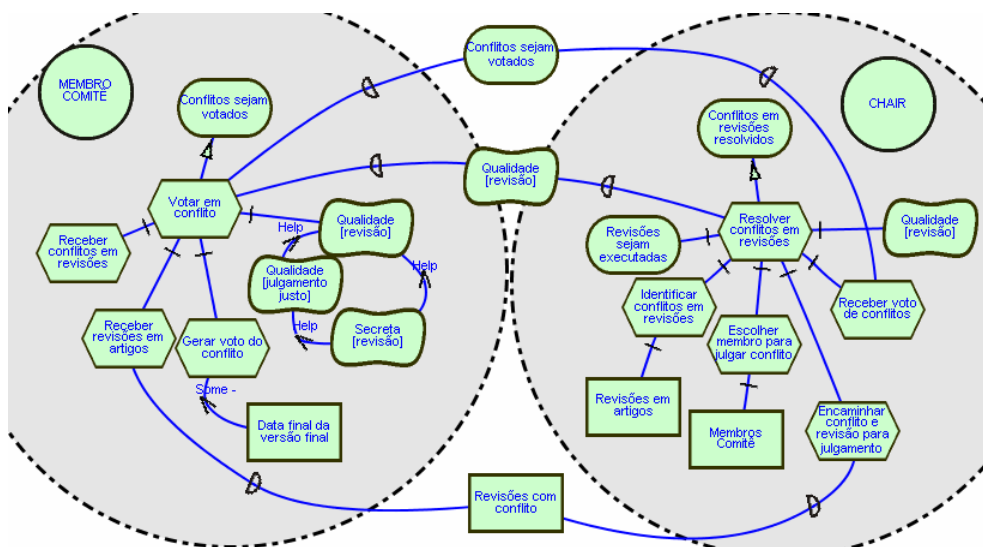


Figura 3.18 – Modelo SR (ator – ator) SDSituation: VOTAÇÃO DE CONFLITOS

ii) MODELAR AS INTERAÇÕES: ATOR – AGENTE

(Estes modelos fornecem a interação de delegação – exemplo Figura 3.19)

- Prepare um Modelo SR para o relacionamento entre o ator e o agente de software que representará o ator no ambiente (“environment”). Esse diagrama é chamado de Modelo da Delegação [Oliveira 06a] (veja a Figura 3.19).
- Modele o relacionamento intencional entre o ator e o agente do ator. Precisam ser decididas quais responsabilidades o ator deseja delegar ao agente de software.
- Adapte as mesmas regras mostradas em (i) para esse caso de interação. Para cada meta concreta (na qual o agente irá apoiar o ator), inclua uma meta concreta similar à do ator com a palavra “apóia” no nome da meta concreta do agente ou inicie com o verbo “apoiar” para as tarefas do agente.

- Crie tarefas para a interação entre eles.

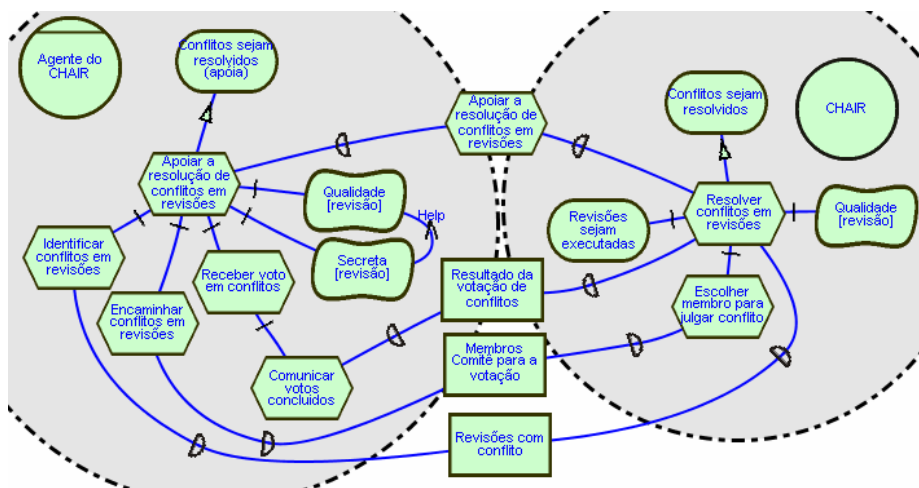


Figura 3.19 – Modelo SR (ator – agente) SDsituation: VOTAÇÃO DE CONFLITOS

iii) MODELAR AS INTERAÇÕES: AGENTE – AGENTE

(Representa as interações entre os agentes no ambiente – veja Figura 3.20).

- Modele o relacionamento intencional entre os agentes. Use o Modelo SR (Figura 3.20) para expressar o relacionamento intencional e prepare um modelo para cada relacionamento entre agentes que representarão os atores no ambiente.
- Modele o relacionamento intencional entre os agentes.
- Adapte as mesmas regras mostradas em (i) a este caso.

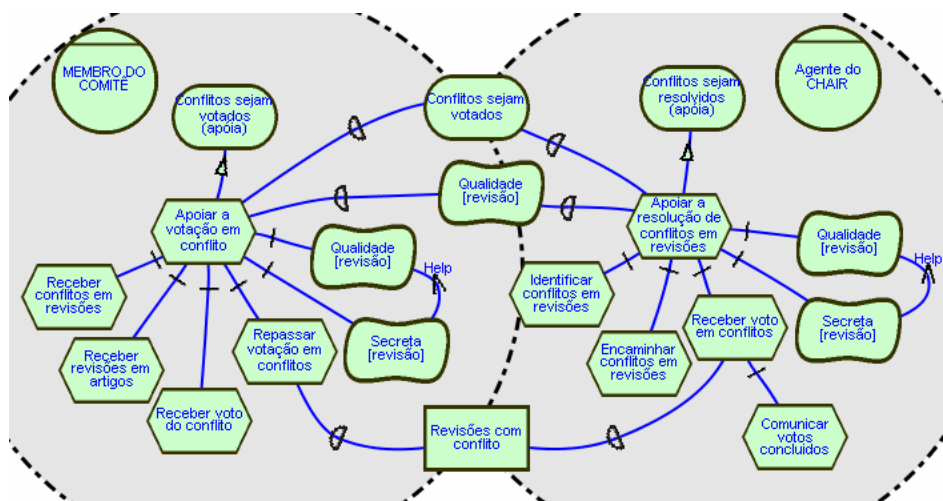


Figura 3.20 – Modelo SR (agente – agente) SDsituation: VOTAÇÃO DE CONFLITOS

5) Especificar as SDsituations

Objetivo: Especificação das Situações de Dependência Estratégica.

A quinta etapa é composta por apenas uma atividade: a) Descrever as Situações de Dependência Estratégica, veja Figura 3.21.

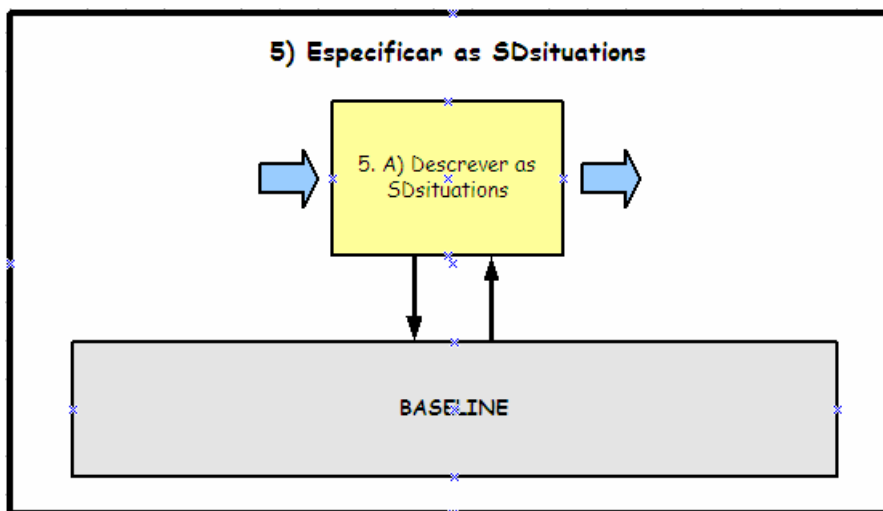


Figura 3.21 – Detalhamento da etapa 5 do Método ERi*c

5.A) Descrever as Situações de Dependência Estratégica

Para a descrição de SDSituations através de Cenários, o engenheiro de requisitos necessita utilizar artefatos preparados nas fases anteriores do Método ERi*c. São necessários o Diagrama de SDSituations e os Modelos SR de cada SDSituation.

A atividade descreve SDSituations aplicando a Técnica de Cenários. A construção de Cenários deve seguir a técnica determinada e dar ênfase aos conceitos de Sistemas Multi-Agentes (propriedades de agência). A ferramenta de software C&L, ferramenta de gerenciamento de léxicos e cenários, apóia essa atividade. O C&L foi desenvolvido pelo grupo de engenharia de requisitos da PUC-Rio e está disponível em [C&L – PUC-Rio].

HEURÍSTICAS PARA DESCREVER SDSITUATIONS:

Para a descrição use o formato pré-definido com os elementos do “template” do cenário.

- Maximize o uso de símbolos do LAL na descrição de cada SDSituation. A Figure 3.22 apresenta como exemplo a SDSituation Aceitação de Propostas. Esse cenário descreve o que está envolvido na situação de dependência para designar artigos a serem revisados por revisores. Observe que as palavras ou sentenças sublinhadas são símbolos do LAL.

- Dê ênfase aos elementos que lidem com as propriedades de agência como autonomia, pró-atividade, sociabilidade, adaptação e interação, além de colaboração, aprendizado e mobilidade. É importante mostrar o que foi considerado na SDsituation ao descrever o cenário: (1) Atores em episódios recebem atenção especial porque participam da propriedade de interação; (2) O nome do objetivo do cenário deve obedecer às regras de descrição de elementos do i^* , veja Tabela 3.2; (3) Os episódios irrelevantes para lidar com propriedades de MAS devem ser agrupados dentro de um episódio de mais alto nível (“*Chair prepara propostas”). (4) Atributos de qualidade devem ser representados com a mesma sintaxe de metas flexíveis, veja Tabela 3.2.
- Coloque os atores envolvidos de maneira explícita, pois isso facilita a visualização das dependências entre os atores.

Título:	ACEITAÇÃO DE PROPOSTAS
Objetivo:	<u>Propostas sejam aceitas</u>
Contexto:	
Localização geográfica:	WEB
Localização temporal:	Logo após o prazo <u>das submissões</u> .
Precondição:	<u>Relação dos revisores</u> e <u>lista de artigos</u> devem estar preparadas.
Recursos:	Computador, Internet, <u>Relação dos revisores</u> , <u>Lista de artigos</u>
Atores:	<u>Chair</u> e <u>revisores</u>
Episódios:	<u>Chair prepara propostas</u> . <u>Chair</u> seleciona <u>revisores</u> que sejam da mesma <u>área do artigo</u> . <u>Chair</u> separa os <u>revisores</u> que sejam da mesma <u>instituição</u> que os <u>autores</u> . <u>Chair</u> conclui as <u>propostas de revisão</u> . <u>Chair</u> envia as propostas para os <u>revisores</u> e fornece o <u>prazo da aceitação</u> .
Restrição:	Cada <u>revisor</u> não deve receber mais de 3 <u>artigos</u> . Cada <u>artigo</u> deve ser revisto por 3 <u>revisores</u> .
Exceções:	Se pelo menos um <u>artigo</u> não tiver 3 <u>revisores</u> : (SDsituation: "Formação do Comitê")
Metas flexíveis:	qualidade [revisão], reconhecimento [comitê]

Figura 3.22 – Exemplo da especificação da SDsituation: ACEITAÇÃO DE PROPOSTAS

6) Analisar a Racionalização das Metas dos Atores

Objetivo: Verificação dos modelos i^* para a melhoria da qualidade.

A sexta etapa é composta por três atividades: a) Identificar estruturas canônicas, b) Aplicar framework de perguntas e c) Verificar as perguntas respondidas, veja Figura 3.23.

Nosso processo focaliza a prevenção de defeitos ao invés da detecção, considerando tanto a eficiência quanto a eficácia do desenvolvimento de Sistemas Multi-Agentes.

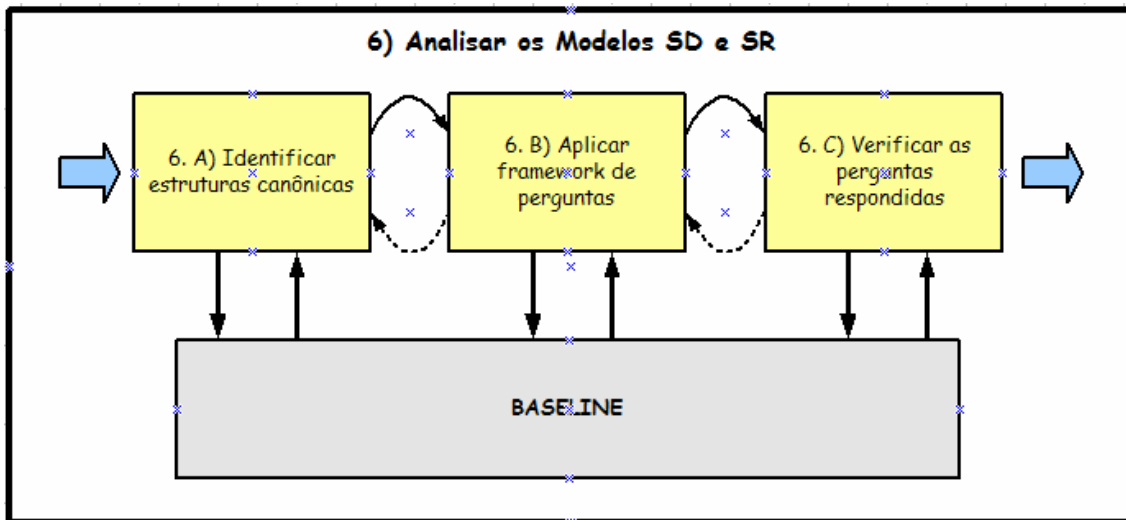


Figura 3.23 – Detalhamento da etapa 6 do Método ERi*c

Para cada modelo SD ou SR foi preparado um framework especial de perguntas. Após as duas atividades de diagnósticos, dentro da atividade Preparar Quadro de Metas x Problemas, uma subatividade de integrar as respostas consiste em casar os problemas potenciais para que estes possam ser analisados. Durante essa subatividade, o engenheiro de requisitos deve reclassificar as respostas que tratam de um mesmo problema. Neste ponto devem ser eliminadas também duplicações de diagnósticos que são naturais confirmações de um mesmo problema. O objetivo geral do framework de diagnóstico é revirar compostos de uma SDsituation com os SRconstructs correlatos procurando por problemas, omissões, deficiências e potenciais melhorias. A estratégia é integrar as duas modalidades de diagnósticos. “Diagnoses are important to deeply understand the problem before looking for the solution” [Oliveira 08b].

Quadro 3.1 – Resumo e correspondência entre as etapas do processo de diagnósticos (Analisar a Racionalização das Metas dos Atores) e o ciclo PDCA

P – Plan: Identificar estruturas canônicas

Obter modelos: SD e SR (normalmente mais de um)

Encontrar as estruturas (SDsituations e SRconstructs)

Ordenar estruturas para o diagnóstico
Preparar as orientações para os diagnósticos
Escolher verificadores

D – Do: Aplicar framework de perguntas

Preparar documentos de apoio com as perguntas do framework
Distribuir material (orientações, perguntas e estruturas) aos verificadores
Aplicar framework de verificação
Receber respostas e comentários dos diagnósticos individuais

C – Check: Verificar as perguntas respondidas

Casar perguntas e respostas de estruturas acopladas
Preparar Quadro de Metas x Problemas
Programar reuniões de verificação e validação
Escolher participantes da reunião (verificadores, moderador, escriba)
Realizar reuniões

A – Action: Corrigir problemas (atividades do método anteriores ao diagnóstico)

Acertar diagramas IP
Alterar modelos SD e SR
Atualizar diagrama de SDsituations
Atualizar especificação de SDsituations

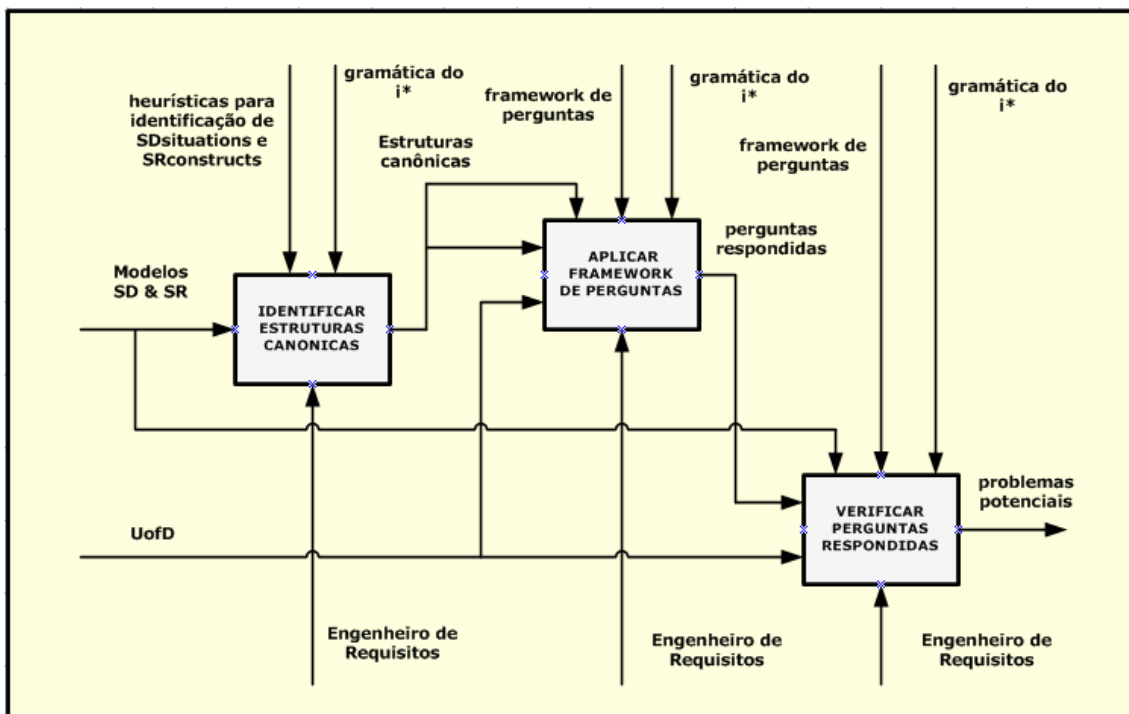


Figura 3.24 – Esquema SADT: Analisar a Racionalização das Metas dos Atores

6.A) Identificar estruturas canônicas

Para diagnosticá-las, a primeira subatividade é Identificar os construtos.

Considerando as SDSituations como unidades básicas, preparamos os diagnósticos dos modelos SD. O objetivo do diagnóstico é identificar quais são as faltas e deficiências do modelo e dessa maneira encontrar recomendações de mudanças nas dependências atuais, nas ligações entre SDSituations e também no “rationale” dos atores de modo que as metas do sistema desejado (“system-to-be”) sejam alcançadas. Diagnosticando com base nas SDSituations do sistema atual (“system-it-is”), podemos encontrar SDSituations do sistema desejado. Diagnósticos são importantes para compreender em profundidade os problemas antes de buscar uma solução.

Dada as estruturas básicas das SDSituations e dos SRconstructs, questões são formuladas. As palavras em negrito são “lacunas” ou “place-holders” a serem substituídas pelos nomes reais dos elementos presentes nos modelos.

Apesar de uma SDSituation ser formada por elementos de dependência, a finalidade de diagnosticar uma SDSituation é descobrir quais dependências são “problemáticas” e assim analisar se existe algum outro ator que poderia colaborar melhor naquela SDSituation. O Quadro 3.2 ilustra as questões que podem ser automaticamente preparadas.

Um SRconstruct possui um fim (**end**) a ser alcançado por uma ou mais tarefas que são meios (**means**). Por sua vez, as tarefas têm elementos agregados necessários para o alcance da meta pela tarefa. A finalidade de diagnosticar SRconstructs (Quadro 3.3) é identificar quais componentes são “problemáticos”, assim como descobrir se existe alguma maneira ou alternativa que poderia ser usada para o alcance de uma determinada meta.

6.B) Aplicar framework de perguntas

Esta atividade consiste em partir os diagramas i* em construtos, ou seja, partir o problema em pequenas peças que permitam seguir com uma análise detalhada de cada parte do diagrama. A subatividade Aplicar questionamento consiste em aplicar o framework de perguntas para cada SRconstruct de cada SDSituation.

Quadro 3.2 - Diagnóstico de SDSituations:

SDSITUATION: “nome da SDSituation”

- I. QUESTÕES EXTERNAS
1. Quem mais poderia colaborar com o “depende” para atingir “nome da meta da SDSituation”? Quanto ele pode colaborar? (completamente ou parcialmente)
 2. Porque o “dependee” colabora com o “depende” para atingir “nome da meta da SDSituation”?
 3. Quais SDSituations acontecem antes de “nome da SDSituation”?
 4. Que problemas com as SDSituations anteriores podem ser identificados para atingir “nome da meta da SDSituation”?
 5. E se o “dependee” não puder colaborar na “nome da SDSituation”?
- II. QUESTÕES INTERNAS
6. Quais são os problemas dentro da “nome da SDSituation”? Que tipos de problemas (precisão, deficiências, ambigüidades ou omissões) são identificados na “nome da meta da SDSituation”?
 7. De que detalhes o “depende” necessita?
 - a) Caso: dependência de recurso – Quais são os problemas de disponibilidade (tempo e precisão) de “nome do recurso”? Quando? Como? Quanto?
 - b) Caso: dependência de meta concreta – Quais são os problemas que “nome da meta concreta” encontra para ser alcançada pelo “dependee”? (tempo, habilidade) Quando? Como? Quanto?
 - c) Caso: dependência de meta flexível – Quais são os problemas que “nome da meta flexível” encontra para ser razoavelmente satisfeita pelo “dependee”? (capacidade) Existe “nome da meta flexível” na conclusão de “nome da SDSituation”? Por quê? Quem está demandando pela meta flexível?
 - d) Caso: dependência de tarefa – O “dependee” recebeu as orientações de como fazer “nome da tarefa”? Pode o “dependee” realizar a tarefa? (tempo, habilidade)
 8. Qual dependência possui maior responsabilidade para atingir “nome da meta da SDSituation”? Por quê?

Quadro 3.3 - Diagnóstico de SRconstructs:

SRCONSTRUCT: “nome da meta final do SRconstruct”

- I. QUESTÕES EXTERNAS
1. Quem mais tem como meta “nome da meta final”?
 2. Quais são as alternativas possíveis para que “nome da meta final” seja atingida? Por quê?
 3. Quais são os elementos de dependência dos dependees?
 4. Que tipos de problemas (precisão, deficiências, ambigüidades ou omissões) podem ser vislumbrados? E se os recursos ficarem indisponíveis? Como evitar tais problemas?
 5. E se a “nome da meta final” for partilhada com outro ator?
 6. Que outro construto depende dessa meta? Por quê? Quanto?
- II. QUESTÕES INTERNAS (PARA CADA TAREFA MEIO)
7. Quais são os problemas com a tarefa “nome da tarefa meio”? Por quê?
 8. Para a tarefa “nome da tarefa meio” que componentes são necessários para atingir “nome da meta final”?
 - a) Caso: recurso – Quais são os problemas de disponibilidade do recurso “nome do recurso”? (tempo, precisão) Quando? Como?
 - b) Caso: meta flexível – Quais são os problemas para que “nome da meta flexível” seja razoavelmente satisfeita pelo “dependee/ator”? (capacidade) Existe “nome da meta flexível” no final do “SRconstruct goal's name”? Por quê? Quais são as contribuições recebidas ou fornecidas por “nome da meta flexível”?
 - c) Caso: subTarefa – Pode o ator “dependee/ator” fazer “nome da tarefa”? (tempo, habilidade)
 9. Existe algum detalhe omitido na operacionalização? De que tipo? Por quê? Como? Quanto?
 10. Há falta de algum recurso? De que tipo? E se o recurso não estiver disponível?

OBS.: Para não haver repetição, não consideramos as metas concretas subcomponentes da decomposição de tarefas porque estas são o elemento principal de análise de outro SRconstruct.

6.C) Verificar as perguntas respondidas

Um resultado que pode ser obtido a partir dos diagnósticos é o quadro de Metas x Problemas, veja Quadro 3.4. A estratégia de diagnósticos indica, com o resultado, problemas potenciais ou deficiências as quais são colocadas na coluna número um do quadro. As metas afetadas pelos problemas que são identificadas nos SRconstructs e SDSituations são assinaladas nas demais colunas do quadro. O quadro Metas x Problemas tem a propriedade de concisão e mostra a influência dos problemas sobre as metas. Na ilustração, o quadro mostra que o problema **Pm** tem influência muito forte sobre grande parte das metas e por isso uma estratégia a ser considerada é reforçar os investimentos para que as chances de o problema **Pm** ocorrer sejam reduzidas.

Quadro 3.4 - Quadro de Metas x Problemas

Metas → Problemas	M 1	M 2	M 3	M 4	M 5	M i	M j	M l	M m	M n	M z
P1		X	X				X	X	X		
P2			X			X			X		
P3					X	X					
Pi				X		X					
Pm	X	X	X		X	X	X		X	X	X
Pn						X					X
Pz			X			X					

3.3. Conclusão

A finalidade desse capítulo foi apresentar o método Engenharia de Requisitos Intencional - ERi*c, que tem a intencionalidade dos atores de uma organização como norte. Para que esse objetivo fique completo precisamos identificar os proprietários das metas (concretas e flexíveis), isto é, os atores da organização. Apesar de existirem dificuldades para se descobrir papéis e posições ocupados, concordamos que a simples identificação de atores é na maior parte das vezes fácil de ser concluída, uma vez que os atores são mencionados diretamente em documentos e entrevistas. O núcleo do problema é: como elicitar e expressar a intencionalidade. Por intencionalidade deve-se entender os interesses e as motivações dos atores [Yu 95] e esse é o ponto central da

proposta deste capítulo. Nele apresentamos uma técnica para elicitar a intencionalidade dos atores sociais, de maneira sistemática, antes de modelá-la.

Nossa proposição pode ser considerada um processo indireto apoiado por questões [Potts 94] sobre o Léxico Ampliado da Linguagem – LAL [Leite 93]. Queremos destacar que dentro de cada símbolo do LAL (veja os exemplos nas Figuras 3.4, 3.5, 3.6 e 3.7) o elemento impacto tem a finalidade de representar as ações: as que acontecem, as que são reflexos de outra ação, as que são aplicadas e também aquelas que podem ocorrer.

Nossa idéia é simples: ações mudam estados e estes são metas. “A goal is a condition or state of affairs in the world that an actor would like to achieve” [Yu 95]. De modo a descobrir os estados desejados que uma ação quer mudar, precisamos perguntar *Por que a ação ocorre?*.

As metas (concretas e flexíveis) dos atores sociais são a principal origem dos requisitos. Uma meta possui duas conotações em nossa abordagem: forma e conteúdo. A forma significa aparência (conformação) “como a meta é comunicada” e conteúdo (substância) está ligado à significância, “que consequência a meta possui”. De maneira a fortalecer a idéia de forma e conteúdo escolhemos quadros com lacunas (“templates”) para guiar o Engenheiro de Requisitos. Foram criados três templates estimulados pelos quatro tipos de símbolos do LAL. Os “templates” foram preparados para receber respostas curtas elicitando os estados afetados pelos impactos (“behavioral responses”) dos símbolos do LAL. Cada “template” recebe “a resposta para a questão **por quê?**”.

Em consequência, na nossa abordagem, as metas possuem uma sintaxe estruturada. A meta concreta deve seguir a forma: **objeto + seja(m)|esteja(m) + verbo na passiva**. E a meta flexível: **tipo + [tópico]**, seguindo a nomenclatura do “NFR Framework de Chung [Chung 00], em que tipo é um atributo de qualidade e tópico é um símbolo do léxico da linguagem. Nos templates adotados, fixamos algumas partes para facilitar o trabalho do Engenheiro de Requisitos. As metas devem ser descritas utilizando os símbolos do LAL e os verbos (que denotam estados na voz passiva) do domínio da aplicação, os quais são normalmente repetitivos. A intenção dos “templates” é também facilitar o trabalho do engenheiro de requisitos ao expressar as metas na sintaxe adequada.

A Figura 3.1 fornece uma visão geral do método. No capítulo foram apresentados os passos e as atividades além de conceitos, heurísticas e sugestões para a aplicação do método. Três exemplos com os passos mais relevantes estão detalhados no capítulo 4.