

## 2

### O *framework* i\*

Este capítulo apresenta o *framework* i\*, no qual está baseada a abordagem orientada a metas usada nesta dissertação. Ao longo do capítulo será apresentada uma visão geral do *framework* i\*, os modelos SD e SR e por fim, duas extensões do *framework* i\* original: o modelo SA (*Strategic Actor*) e *SDsituations*.

#### 2.1 Visão geral do *framework* i\*

O *framework* i\* (i-estrela) [1] modela contextos organizacionais baseado nos relacionamentos de dependência entre os atores. O i\* é um *framework* usado para obter um melhor entendimento dos relacionamentos organizacionais entre os vários agentes organizacionais. Os relacionamentos entre os atores ajudam os engenheiros de requisitos a elicitar metas nas fases iniciais (Por que os atores têm estas dependências?). O uso do *framework* i\* possibilita a compreensão das razões internas dos atores, uma vez que as mesmas são expressas explicitamente, auxiliando na escolha de alternativas durante a etapa de modelagem do software.

No i\*, os participantes do contexto organizacional são chamados de atores. De acordo com Yu [1], atores são entidades ativas que efetuam ações para alcançar metas através do exercício de suas habilidades e conhecimentos, podendo ter dependências intencionais entre si. Uma dependência intencional ocorre quando o elemento da dependência está, de alguma forma, relacionado a alguma meta de um dos atores.

O *framework* i\*, como originalmente proposto por Yu em [1], usa dois modelos: o modelo SD (*Strategic Dependency*) e o modelo SR (*Strategic Rationale*), descritos a seguir.

## 2.2 Modelo SD

O modelo SD é usado para mapear a rede de dependências entre os atores organizacionais. O modelo SD é composto por um conjunto de nós que representam os atores (agentes, posições ou papéis) e elos que representam as dependências entre os atores. Uma dependência representa um acordo entre dois atores, onde um ator (“*dependor*” ou “*dependente*”) depende de um outro ator (“*dependee*” ou “*de quem se depende*”) para que uma meta (*goal*) seja alcançada, uma tarefa seja executada, um recurso seja disponibilizado ou uma meta-flexível (*softgoal*) seja razoavelmente satisfeita. Metas, metas flexíveis, tarefas e recursos que fazem parte das relações de dependência entre os atores são chamados de *elementos de dependência (dependum)* e caracterizam os quatro tipos de dependência possíveis, descritos a seguir:

- **Dependência por meta:** ocorre quando o *dependor* depende do *dependee* para que certo estado do mundo seja alcançado. Ao *dependee* é dada a liberdade de escolher como fazê-lo. Com uma dependência por meta, o *dependor* ganha a habilidade de assumir que a condição ou estado do mundo será alcançado, mas torna o *dependor* vulnerável, pois o *dependee* pode falhar em realizar tal condição;
- **Dependência por tarefa:** ocorre quando um ator (o *dependor*) depende de outro (o *dependee*) para que este outro desempenhe uma tarefa. Uma dependência por tarefa específica “como” a tarefa deve ser desempenhada, mas não específica “porquê”. O *dependor* é vulnerável, pois o *dependee* pode falhar em desempenhar a tarefa. A especificação de uma tarefa deve ser vista mais como uma restrição do que como o conhecimento prévio adequado (*knowhow*) para realização da tarefa.
- **Dependência por recurso:** ocorre quando um ator (o *dependor*) depende de outro (o *dependee*) para que uma entidade (física ou computacional) seja disponibilizada. Com o estabelecimento deste tipo de dependência, o *dependor* ganha a habilidade de usar a entidade como um recurso, ficando ao mesmo tempo vulnerável, pois a entidade pode tornar-se indisponível.

- **Dependência por meta flexível:** ocorre quando um ator (o *dependor*) depende de outro (o *dependee*) para que este desempenhe alguma tarefa para que uma meta flexível seja “satisfeita a contento” ou “razoavelmente satisfeita”, isto é, seja satisfeita a um nível considerado aceitável. A expressão "satisfeita a contento" aqui é utilizada como uma tradução do termo em inglês *satisficcy*, introduzido por Herbert Simon [2] nos anos 1950. O conceito de meta flexível é usado para designar metas cujos critérios para sua satisfação não estão claramente definidos *a priori*, enquanto o conceito de meta rígida, chamadas apenas metas, está baseado numa noção clara em relação a sua satisfação, do tipo sim ou não - metas (rígidas) são usadas para representar determinados estados em que o mundo está ou não. (Esta subjetividade quanto a sua satisfação é inerente às metas flexíveis, e aos requisitos não-funcionais - por se tratarem de requisitos de qualidade, os critérios para sua satisfação não estão claramente definidos *a priori*.) Numa dependência por meta flexível não há um acordo prévio entre os atores (*dependor* e *dependee*) sobre o que constitui a satisfação da meta (flexível). O significado da meta flexível é especificado em termos dos métodos que são escolhidos no curso da busca para alcançar a meta. Assim com em uma dependência por meta, o *dependor* ganha a habilidade de assumir que a condição ou estado do mundo será alcançado, mas torna-se vulnerável, pois o *dependee* pode falhar em realizar tal condição. Diferentemente do que ocorre com as metas, as condições para que as metas flexíveis sejam alcançadas são elaboradas ao passo que as tarefas são desempenhadas.

Estes quatro tipos de dependência também caracterizam como as decisões de processo (escolha de alternativas; seqüência de passos para execução de uma tarefa) recaem em cada lado da dependência. Em uma dependência por meta, o *dependee* tem liberdade para tomar as decisões necessárias para que a meta seja satisfeita. Em uma dependência por tarefa o *dependor* toma as decisões. Em uma dependência por recurso a questão da decisão não vem à tona, por ser o recurso considerado um produto final de algum processo de ação-deliberação, não havendo assim nenhuma decisão em aberto a ser considerada. Em uma decisão por meta flexível, o *dependor* toma a decisão final, mas faz isso beneficiado pelo

conhecimento prévio adequado (*knowhow*) do *dependee*. A Figura 2.1 ilustra como são representados os quatro tipos de relação em um modelo SD.

### 2.3 Modelo SR

O modelo SR tem como objetivo representar as estratégias internas de cada ator. O modelo SR provê uma descrição intencional do processo em termos dos elementos do processo e das decisões e escolhas por trás dele. O modelo SR permite representar explicitamente o entendimento detalhado do raciocínio (*rationale*) dos atores do processo. Estes modelos são elaborados de modo a expressarem o processo pelo qual: as metas são alcançadas, as tarefas são elaboradas, os recursos são disponibilizados e as metas flexíveis são refinadas (decompostas) e operacionalizadas. Isto é feito pela representação dos relacionamentos intencionais que são internos aos atores através de relacionamentos “meios-fim” que relacionam elementos de processo, provendo uma representação explícita do ‘porque’, do ‘como’ e de alternativas.

O modelo SR é um grafo, com alguns tipos de nós e elos que provêm uma estrutura de representação para expressar explicitamente as razões por trás do processo. Há quatro tipos de nós, idênticos aos tipos de *dependum* em um modelo SD: meta, tarefa, recurso e meta flexível. Há duas classes principais de elos: elos de decomposição de tarefa e elos “meios-fim”.

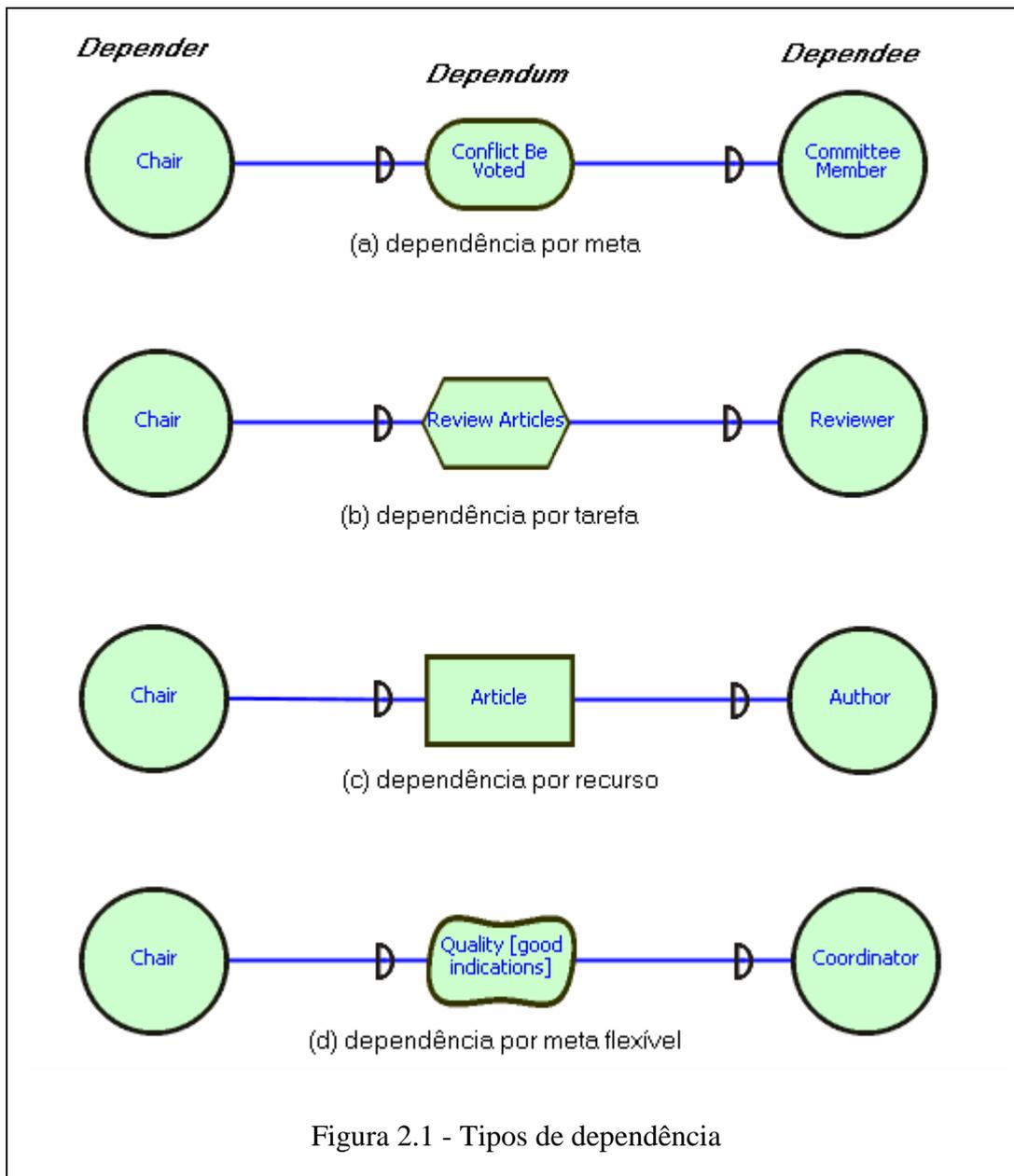
**Elos de decomposição de tarefa** – Uma tarefa (cuja noção está associada a ‘como fazer alguma coisa’) é modelada em termos de sua decomposição em suas sub-componentes, que podem ser: metas, tarefas, recursos e/ou metas flexíveis (os quatro tipos de nós).

Uma **meta** é uma condição ou estados de desejos no mundo que o ator deseja alcançar, expressa como uma assertiva na linguagem de representação. Como a meta deve ser alcançada não é especificado, possibilitando a consideração de várias alternativas.

Uma **tarefa** especifica um modo particular de fazer alguma coisa. Quando uma tarefa é especificada como sub-tarefa (parte) de outra tarefa (todo) ela

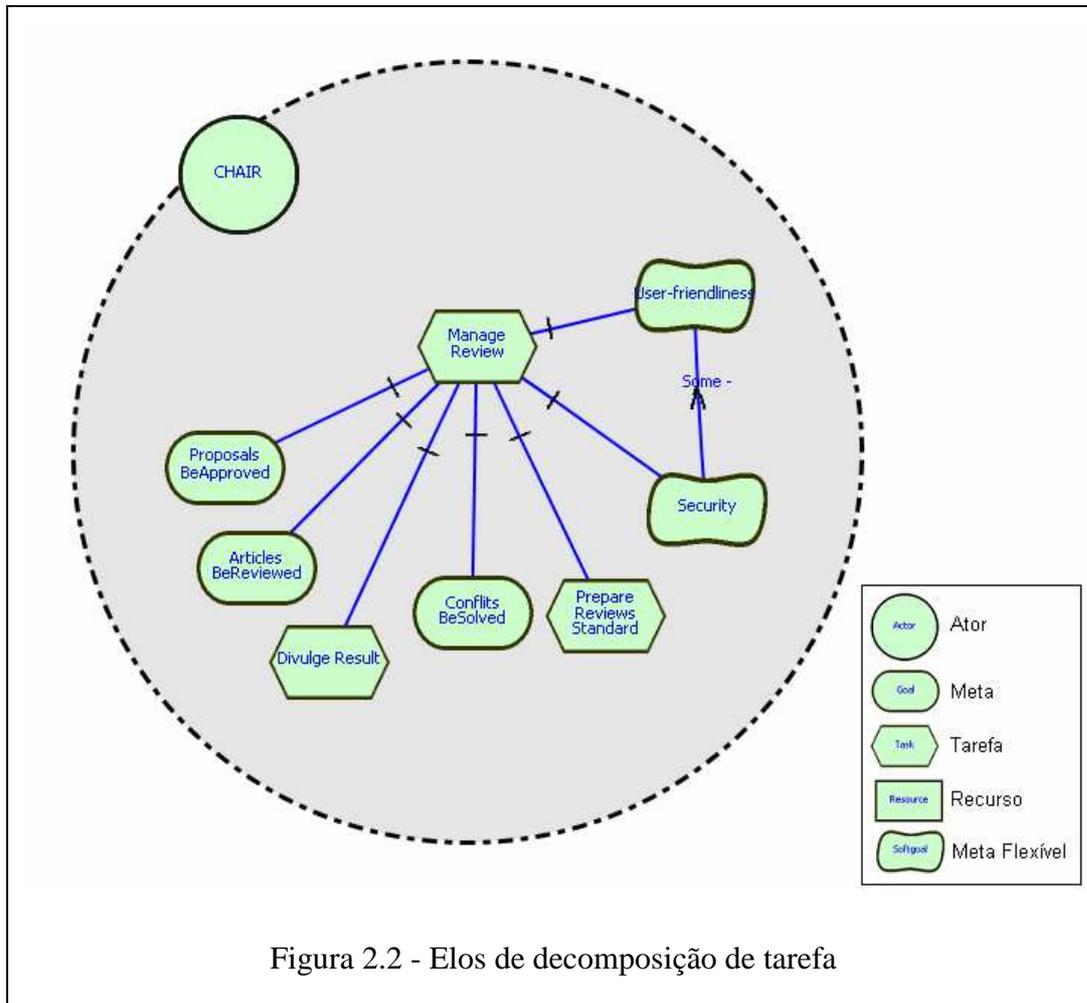
restringe esta outra tarefa (todo) a um curso de ação em particular, especificado pela tarefa sub-tarefa.

Um **recurso** é uma entidade (física ou informacional) que não é considerada problemática pelo ator. A principal característica é se está disponível (e por quem foi disponibilizado no caso de uma dependência externa).



Uma **meta flexível** é uma condição ou estado no mundo que o ator deseja alcançar, mas diferentemente de uma meta (rígida), o critério para a condição ser alcançada não é precisamente definido *a priori*, estando sujeito à interpretação. Quando uma meta flexível é um componente em uma decomposição de tarefa, ela

serve como uma meta de qualidade para aquela tarefa, guiando (ou restringindo) a seleção entre as alternativas para a decomposição da tarefa.



No exemplo com elos de decomposição de tarefa apresentado na Figura 2.2, a tarefa “Manage Review” é decomposta em três (sub) metas: “Proposals Be Approved”, “Articles Be Reviewed” e “Conflicts Be Solved”; duas (sub) tarefas: “Divulge Result” e “Prepare Reviews Standard”; e duas metas flexíveis: “User-friendliness” e “Security”. A meta flexível de segurança contribui negativamente para a meta flexível de usabilidade, como indica o elo de contribuição “Some-” de “Security” para “User-friendliness”.

**Elos Meios-Fim** – Elos do tipo meios-fim indicam relacionamentos entre um “fim”, que pode ser: uma meta a ser alcançada, uma tarefa a ser desempenhada,

um recurso a ser produzido, ou uma meta flexível a ser razoavelmente satisfeita (*satisfied*); e “meios” alternativos para alcançá-lo. Os meios são usualmente expressos na forma de tarefas, uma vez que a noção de tarefa está associada a ‘como fazer alguma coisa’. Na notação gráfica do i\*, uma cabeça de seta aponta dos meios para o fim. Os tipos de elos meios-fim<sup>2</sup> estão descritos a seguir:

Elo Tarefa-Meta – em um elo tarefa-meta o “fim” é especificado como uma meta e o “meio” é especificado como uma tarefa. Esta tarefa especifica “como” através de sua decomposição em seus componentes.

Elo Tarefa-Recurso - neste tipo de elo o “fim” é especificado como um recurso e o “meio” é especificado como uma tarefa.

Elo Tarefa-Meta Flexível - neste tipo de elo o “fim” é especificado como uma meta flexível e o “meio” é especificado como uma tarefa.

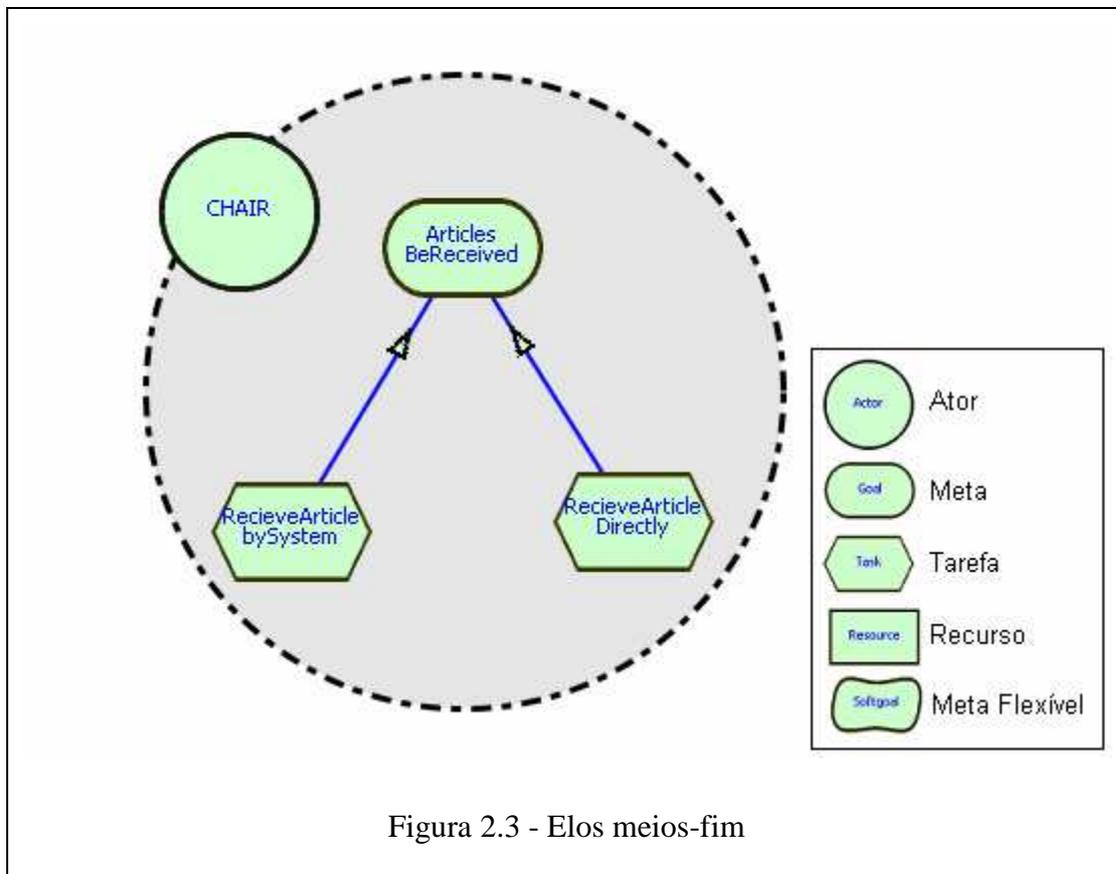
Elo Meta Flexível–Meta Flexível – nos elos do tipo Meta Flexível–Meta Flexível tanto o “fim” quanto o “meio” são especificados como metas flexíveis. Este tipo de elo possibilita o desenvolvimento de uma hierarquia meios-fim de metas flexíveis, até que eventualmente algumas metas flexíveis são operacionalizadas por tarefas (via elos Tarefa-Meta Flexível). Esta hierarquia é particularmente interessante para o refinamento de metas flexíveis, onde uma meta flexível com nível de abstração alto é refinada sucessivamente até que se consigam metas flexíveis mais fáceis de operacionalizar.

Os elos meios-fim que envolvem metas flexíveis possuem um atributo extra para indicar o tipo de contribuição. A escala de contribuição usada por Yu em [1] é simplificada e apresenta três valores para contribuição: positiva (“+”), negativa (“-”) e neutra (“?”), ao passo que Chung et al [7] usa uma escala de contribuição com cinco valores: *brake* (“--”), *hurt* (“-”), *unknown* (“?”), *help* (“+”) e *make* (“++”).

---

<sup>2</sup> Embora o tipo de elo se chame meios-fim (*means-end*), Yu [1] os descreve de forma invertida: fim-meios. Visando facilitar a compreensão, optamos por descrevê-los de forma direta, diferentemente de Yu.

Yu menciona ainda em [1, pág 35] a possibilidade de uso de outros tipos de elos meios-fim como meta-meta, embora este tipo de elo não apareça em nenhum dos exemplos apresentados.



No exemplo de uso de elos meios-fim apresentado na Figura 2.3, há dois elos do tipo Meta-Tarefa, em ambos a meta (“fim”) é “ArticlesBeReceived” sendo as tarefas “ReceiveArticlebySystem” e “ReceiveArticleDirectly” os meios.

## 2.4 Extensões do *framework i\**

Duas extensões ao *framework* original proposto por Yu serão usadas neste trabalho: o modelo SA (Strategic Actor) e *SDSituations*. Estas extensões são detalhadas a seguir.

### 2.4.1 O Modelo SA (*Strategic Actor*)

O modelo SA [13] é usado especificamente para modelar os atores em i\*. A adoção do modelo SA auxilia no entendimento dos atores, e seus relacionamentos, do ponto de vista estrutural. Embora Yu não tenha proposto um modelo específico com esta finalidade, o modelo SA usa os mesmos conceitos presentes em [1]: agente, posição e papel como refinamento dos atores; e os relacionamentos entre eles. Estes conceitos estão descritos a seguir, conforme definido por Yu [1]:

**Ator:** “Um ator é uma entidade ativa que desempenha ações para alcançar suas metas através do uso de seu conhecimento” [1, pág 12]. “O termo ator é usado para se referir genericamente a qualquer unidade a qual se possa atribuir dependências intencionais.” [1, pág. 17].

**Posição:** “Uma posição está em um nível intermediário de abstração entre um papel e um agente. É um conjunto de papéis tipicamente desempenhado”. [1, pág. 17].

**Papel:** “Um papel é uma caracterização abstrata do comportamento de um ator social em algum contexto ou domínio especializado. Suas características podem facilmente ser transferidas para outros atores sociais. Dependências são associadas com papéis quando estas dependências se aplicam independentemente de quem esteja desempenhando o papel”. [1, pág. 17].

**Agente:** “Um agente é um ator com manifestações físicas, concretas, tal qual um ser humano. O termo agente é usado no lugar de pessoa a fim de generalizar, podendo se referir tanto a agentes humanos quanto a agentes artificiais (hardware/software). Um agente tem dependências que se aplicam independentemente de que papel ele está desempenhando. Estas características não podem ser tipicamente transferidas para outros indivíduos, como por exemplo, suas habilidades, suas experiências, suas limitações físicas”. [1, pág. 17].

Em virtude do relacionamento de instanciação entre agentes, foi proposto em [13] um novo tipo de ator: o agente real. O agente real é mais específico que o

agente (mais genérico), podendo ser identificado unicamente, como por exemplo, uma pessoa específica ou um software específico.

Os relacionamentos entre estes conceitos conforme descritos por Leite et al. em [13], baseado nas definições e exemplos de [1], são apresentados resumidamente a seguir:

**Ocupa** - relacionamento de agente para posição – um agente pode ocupar mais de uma posição e uma posição pode ser ocupada por mais de um agente.

**Desempenha** – relacionamento de agente para papel – um agente pode desempenhar mais de um papel e um papel pode ser desempenhado por mais de um agente.

**Cobre** – relacionamento de posição para papel – uma posição pode cobrir mais de um papel e um papel pode ser coberto por mais de uma posição.

**Parte de** – relacionamento de posição para posição; de papel para papel; de agente para agente – posição, papel e agente podem ter mais de uma sub-parte, assim como podem ser sub-parte de mais de uma posição, papel e agente respectivamente. Há uma restrição para este relacionamento entre posições. Uma posição é parte de outra se todos os papéis desta posição também são cobertos pela outra posição.

**É um** – relacionamento de ator para ator; de posição para posição; de papel para papel; de agente para agente – ator, posição, papel e agente podem ser especializados por mais de um ator, posição, papel e agente respectivamente.

**Instancia** – relacionamento de agente real para agente – um agente real instancia um agente; um agente pode ser instanciado por mais de um agente real.

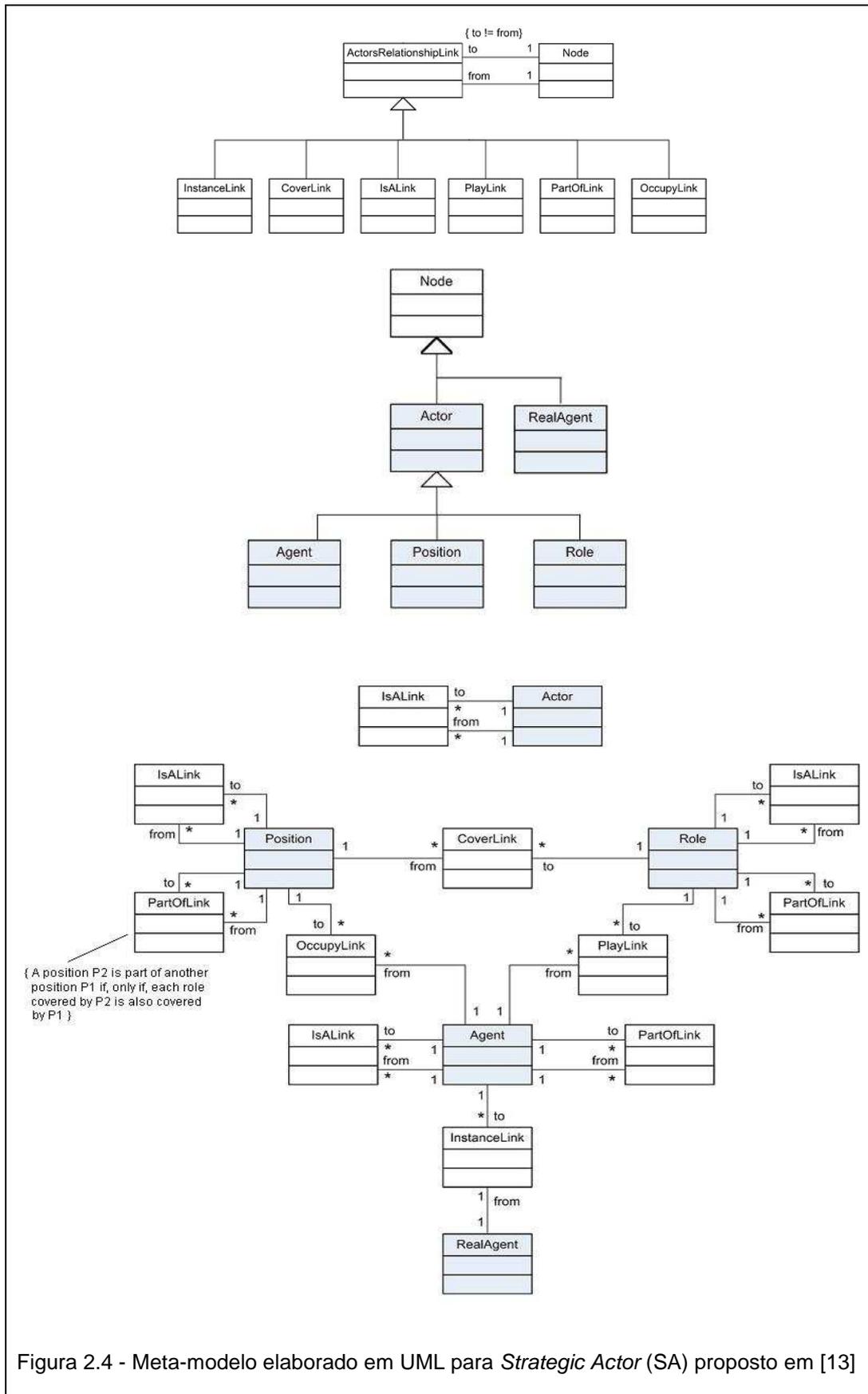


Figura 2.4 - Meta-modelo elaborado em UML para *Strategic Actor (SA)* proposto em [13]

Um modelo SA é um grafo cujos nós são derivados do refinamento de atores: posição, papel, agente e agente real; e cujos elos são os relacionamentos ocupa, desempenha, cobre, parte de, é um e instancia. A Figura 2.4 apresenta um meta-modelo elaborado em UML proposto em [13].

#### 2.4.2 *SDsituations* (Situação de Dependência Estratégica)

*SDsituation* é uma forma de representação estruturada de uma situação de dependência estratégica, proposta por Oliveira e Cysneiros em [4]. As situações de dependência ocorrem naturalmente em ambientes organizacionais. A idéia central é que cada elo de dependência envolvendo atores não é isolado, e sim parte de uma situação bem definida de colaboração chamada de “*strategic dependency situation*” ou *SDsituation*.

Uma *SDsituation* é composta por um ou mais elementos de dependência. Cada *SDsituation* pode ser identificada separadamente das outras *SDsituations* formando uma cadeia de interdependências. Os elementos de dependência de uma *SDsituation* são os mesmos elementos de dependência (*dependum*) presentes em um modelo SD: meta, tarefa, recurso e meta flexível. As interdependências entre *SDsituations* podem ser de três tipos: física, lógica e temporal. Mais de um tipo de interdependência entre as *SDsituations* podem ocorrer ao mesmo tempo. Os três tipos de interdependências entre *SDsituations* estão descritos a seguir:

- Física - ocorre quando um recurso é preparado por uma *SDsituation* e é necessitado por outra *SDsituation*;
- Lógica - ocorre quando uma ou mais *SDsituations* necessitam da conclusão de outra *SDsituation* para sua iniciação ou quando uma ou mais *SDsituations* necessitam da conclusão de outra *SDsituation* para sua conclusão.
- Temporal – ocorre quando uma ou mais *SDsituations* necessitam esperar algum tempo após o início de outra *SDsituation* ou quando uma ou mais *SDsituations* necessitam esperar algum tempo após a conclusão de outra *SDsituation*.

Uma *SDsituation* é caracterizada principalmente pelo conjunto de elementos de dependência que fazem parte da situação de dependência. Além deste conjunto de elementos de dependência, fazem parte da definição de uma *SDsituation* o título, a meta (a meta principal da situação de dependência), e a descrição da situação de dependência. A definição de uma *SDsituation* é feita seguindo três passos básicos:

- (i) identificação dos elementos da *SDsituation* - tanto os atores, *dependeer* e *dependee*, quanto os elementos de dependência entre eles, o *dependum* (metas, tarefas, recursos e metas flexíveis) podem ser identificados a partir do modelo SD. Caso tenha sido construído um Léxico Ampliado da Linguagem (LEL), este pode ser o ponto de partida para identificação dos elementos da *SDsituation* conforme sugerem Oliveira e Cysneiros em [4];
- (ii) composição de *SDSituations* – uma *SDsituation* é composta por uma ou mais dependências. Examinando as dependências no modelo SD, o engenheiro de requisitos deve definir que dependências fazem parte de uma mesma situação;
- (iii) identificação de interrelacionamento entre *SDsituations* – identificação de dependências (lógica, física ou temporal) entre as *SDsituations*. Apesar de uma *SDsituation* ser composta de uma ou mais dependências, uma dependência pode ser uma outra *SDsituation*. É necessário examinar as *SDsituations* identificando para cada uma as dependências de outras *SDsituations*, se existirem.

A seguir, é apresentada como exemplo a descrição da *SDsituation* ‘Revisar artigo’ (*Article Review*). As *SDsituations* serão apresentadas com mais detalhes no estudo de caso no capítulo 4.

<b>SDsituation Definition</b>				
SDsituation:	<b>ARTICLES REVIEW</b>			
Goal:	ArticlesBeReviewed			
Definition:	It represents a situation when the Reviewer reviews an article (the proposal of review has been previously accepted)			
ACTOR (depender)	ELEMENTS OF DEPENDENCY	TYPE	DEGREE	ACTOR (dependee)
	PROPOSAL ACCEPTANCE	DEPENDENCY	critical	
Chair	Review Article	task	critical	Reviewer
Chair	Quality [Good Review]	softgoal		Reviewer
Reviewer	Article to Review	resource	critical	Chair
Chair	Reviewed Article	resource	critical	Reviewer