

UNIVERSIDADE FEDERAL DO MARANHÃO
CENTRO DE CIÊNCIAS EXATAS E TECNOLOGIA
DEPARTAMENTO DE ENGENHARIA DE ELETRICIDADE
PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA DE ELETRICIDADE

FÁBIO LOPES VIEIRA

**UM SISTEMA MULTIAGENTE PARA APOIO AS DECISÕES NO
PROCESSO DE LICITAÇÃO PÚBLICA**

São Luís
2013

FÁBIO LOPES VIEIRA

**UM SISTEMA MULTIAGENTE PARA APOIO AS DECISÕES NO
PROCESSO DE LICITAÇÃO PÚBLICA**

Dissertação apresentada ao Programa de Pós-Graduação em Engenharia da Eletricidade da Universidade Federal do Maranhão, para a obtenção do título de mestre em Engenharia da Eletricidade - Área de Concentração: Ciência da Computação.

Orientadora: Dr^a Rosario Girardi

São Luís
2013

Vieira, Fábio Lopes.

Um sistema multiagente para apoio as decisões no processo de licitação pública / Fábio Lopes Vieira – São Luís, 2013.

120 f.

**Impresso por computador (fotocópia).
Orientador: Rosário Girardi.**

Dissertação (Mestrado) – Universidade Federal do Maranhão, Programa de Pós-Graduação em Engenharia de Eletricidade, 2013.

**1. Sistemas multiagente – Licitações públicas. 2. Sistemas de apoio a decisão.
I. Título.**

CDU 621.391

**UM SISTEMA MULTIAGENTE PARA APOIO AS
DECISÕES NO PROCESSO DE LICITAÇÃO PÚBLICA**

Fabio Lopes Viera

Dissertação aprovada em 07 de Janeiro de 2013.



Profa. María del Rosario Girardi Gutiérrez, Ph.D.
(Orientadora)



Prof. Evandro de Barros Costa, Dr.
(Membro da Banca Examinadora)



Prof. Francisco José da Silva e Silva, Dr.
(Membro da Banca Examinadora)

À minha amada esposa Luciana Vieira.

AGRADECIMENTOS

A Jeová Deus, pela vida, saúde e principalmente, poder além do normal nos momentos de dificuldade.

À minha esposa Luciana Vieira pelo apoio e compreensão, nos diversos momentos que estive ausente tanto em pessoa como em pensamento.

À professora Dra. Rosario Girardi, cujo apoio intelectual, dedicação e estímulo foram fundamentais para realização desta dissertação.

Aos colegas do GESEC. Em especial, Luis Eduardo, Adriana Leite, Ivo Serra, pela ajuda mútua durante a realização deste trabalho. Ao prof. Me. Oswaldo pela ajuda precisa no desenvolvimento do protótipo de interface. Ao prof. Me. Luis Claudio Oliveira pela ajuda precisa no desenvolvimento dos agentes. E mais uma vez, gostaria de lembrar do meu amigo Luis Eduardo, que foi fundamental para a conclusão desse trabalho.

Aos integrantes e ex-integrantes do grupo GESEC que contribuíram direta ou indiretamente para realização desse trabalho.

Ao programa de Pós-Graduação em Engenharia de Eletricidade da Universidade Federal do Maranhão, pela oportunidade de realização do curso.

“Só sei que nada sei.”
Sócrates

RESUMO

A licitação pública é um processo administrativo cujo objetivo é a compra de bens ou serviços para os órgãos pertencentes à Administração Pública e segue as normas da lei nº. 8.666/93. Nesse processo há necessidade de que os agentes públicos envolvidos tomem decisões como a escolha do tipo e da modalidade da licitação. Devido a complexidade da Lei que regula o processo de licitação e às inúmeras possibilidades de condução do processo licitatório em diversos cenários e também a seu dinamismo, diante das constantes alterações na legislação; foi desenvolvido um sistema multiagente para o processo de licitação pública visando otimizar a tomada de decisões dos responsáveis pela aquisição dos bens e serviços na Administração Pública. Um Sistema Multiagente é um sistema composto por vários agentes que se comunicam e são coletivamente capazes de atingir objetivos que não seriam capazes de satisfazer separadamente. A complexidade destes sistemas é abordada através das interações entre os agentes, ou seja, cada agente pode executar, dentro de suas limitações, uma tarefa simples, mas a boa coordenação da execução dessas tarefas simples por cada agente torna o sistema capaz de executar tarefas de grande complexidade. Foi utilizada a arquitetura de troca de mensagem entre agentes, onde os agentes se comunicam diretamente uns com os outros, através de mensagens assíncronas, utilizando um protocolo de conversação, o qual dita as regras e impõe o formalismo necessário para que as mensagens sejam encaminhadas e compreendidas pelos agentes. Para fazer a especificação do sistema, adotou-se o MADAE-Pro, um processo que guia o desenvolvimento de um sistema multiagente nas fases de especificação, projeto e implementação.

Palavras chaves: Agentes. Licitações Públicas. Sistemas de Apoio à Decisão. Sistemas Multiagente.

ABSTRACT

Public licitation is an administrative process which goal is to purchase goods or services to the sectors belonging to the public administration and follow the rules of law no. 8.666/93. In this process public officials need to take decisions such as choosing the type and modality of the licitation. Due to the complexity of the law governing the licitation process and the great possibilities of conducting the licitation process in different scenarios and also its dynamism in the face of constant changes in legislation, we developed a multi-agent system to optimize the decisions of those responsible for acquiring goods and services for the public administration. A Multiagent System is a system composed of several agents that communicate and are collectively capable of achieving goals that they would not be able to meet separately. The complexity of these systems is approached through interactions between agents. We used the exchange messages architecture, where agents communicate directly with each other through asynchronous messages using a chat protocol, which sets the rules and enforces the formalism necessary for messages to be sent and understood by the agents. To specify the system we adopted MADAE-Pro, a process which guides the development of multi-agent systems through the phases of specification, design and implementation.

Keywords: Agents. Public Licitation. Multiagent Systems. Decision Support Systems.

LISTA DE FIGURAS

Figura 1 – Agente Genérico interagindo com seu ambiente [84].....	34
Figura 2 - Elementos envolvidos no MADAE-Pro	45
Figura 3 - Metamodelo de Conceitos [61].....	50
Figura 4 – Metamodelo de Objetivos [61]	51
Figura 5 – Metamodelo de Papéis [61].....	52
Figura 6 – Metamodelo de Interações entre Papéis [61]	53
Figura 7 – Metamodelo do Conhecimento da Sociedade Multiagente [61].....	54
Figura 8 – Metamodelo da Sociedade Multiagente [61]	56
Figura 9 – Metamodelo de Interações entre Agentes [61]	57
Figura 10 – Metamodelo de Coordenação e Cooperação [61]	58
Figura 11 – Metamodelo do Conhecimento do Agente [61].....	59
Figura 12 – Metamodelo de Ações do Agente [61]	60
Figura 13 – Metamodelo de Comportamentos [61]	61
Figura 14 – Metamodelo de Atos de Comunicação [61].....	62
Figura 15 – Rede semântica com os principais conceitos de modelagem especificados pelas metodologias MADEM e MAAEM [61]	63
Figura 16 - Rede semântica das tarefas e subtarefas referentes à fase de Análise das metodologias MADEM e MAAEM [61]	64
Figura 17 – Modelo de Conceitos do Sistema SMPLP	67
Figura 18 – Modelo de Objetivos do Sistema SMPLP.....	69
Figura 19 - Modelo de Papéis do SMPLP referente ao objetivo específico “Realizar pedido de compra”	70
Figura 20 – Modelo de Papéis do SMPLP referente ao objetivo específico “Definir objeto de compra”	70
Figura 21 – Modelo de Papéis do SMPLP referente ao objetivo específico “Avaliar proposta de compra”.....	71
Figura 22 – Modelo de Papéis do SMPLP referente ao objetivo específico “Gerar Edital”	72
Figura 23 – Modelo de Interação entre Papéis do SMPLP referente ao objetivo específico “Realizar pedido de compra”.....	73
Figura 24 – Modelo de Interação entre Papéis do SMPLP referente ao objetivo específico “Definir objeto de compra”	74
Figura 25 – Modelo de Interação entre Papéis do SMPLP referente ao objetivo específico “Avaliar proposta de compra”	75
Figura 26 – Modelo de Interação entre Papéis do SMPLP referente ao objetivo específico “Gerar Edital”	76
Figura 27 – Protótipo de Interface com o Usuário “Login”	78
Figura 28 – Protótipo de Interface com o Usuário para o perfil “Setor da Administração”	78
Figura 29 – Protótipo de Interface com o Usuário “Elaboração do Pedido”	79
Figura 30 – Protótipo de Interface com o Usuário “Setor de Compras”	79
Figura 31 – Protótipo de Interface com o Usuário “Elaboração e Submissão do Miniprojeto”	80
Figura 32 – Protótipo de Interface com o Usuário – “Setor de Compras”	80
Figura 33 – Protótipo de Interface com o Usuário “Análise de Relevância para o Interesse Público”	81
Figura 34 – Protótipo de Interface com o Usuário “Gerar Edital”	82
Figura 35 – Protótipo de Interface com o Usuário – “Estimativa do Custo do Bem/Serviço”	83
Figura 36 – Modelo de Conhecimento da Sociedade Multiagente do Sistema SMPLP	85

Figura 37 – Modelo de Sociedade Multiagente do SMPLP apresentando os agentes “Interface” e “Analista”	86
Figura 38 – Modelo de Sociedade Multiagente do SMPLP apresentando o agente “Licitador”	87
Figura 39 - Modelo das Interações entre Agentes do Sistema SMPLP.....	88
Figura 40 – Modelo dos Mecanismos de Cooperação e Coordenação do Sistema SMPLP	89
Figura 41 – Modelo do Conhecimento do agente “Interface”	90
Figura 42 – Modelo do Conhecimento do agente “Analista”	91
Figura 43 – Modelo do Conhecimento do agente “Licitador”	92
Figura 44 – Modelo das Ações do agente “Interface”	94
Figura 45 – Modelo das Ações do agente “Analista”	95
Figura 46 – Modelo das Ações do agente “Licitador”	96
Figura 47 – Modelo de Comportamentos	98
Figura 48 – Modelo de Atos de Comunicação	99
Figura 49 – Parte do código do agente “Interface”	99
Figura 50 – Comunicação Entre Agentes	100
Figura 51 – Parte do código do agente “Interface”	100
Figura 52 – Agente “Interface” transmitindo uma mensagem ao agente “Analista”	101
Figura 53 – Troca de mensagens entre o agente “Interface”, “Analista” e “Licitador”	101
Figura 54 – Parte do código do agente “Licitador”	102
Figura 55 – Base de conhecimento no Prolog	102

LISTA DE TABELAS

Tabela 1 – Análise Comparativa de alguns Sistemas de Apoio à Licitação Pública.....	42
Tabela 2 - Fases, Insumos, Tarefas e Produtos da metodologia MAAEM [61].....	49

SUMÁRIO

1	INTRODUÇÃO.....	15
1.1	Relevância e motivação	15
1.2	Objetivos	16
1.2.1	Objetivo geral	16
1.2.2	Objetivos específicos	16
1.3	Organização da dissertação	16
2	A LICITAÇÃO PÚBLICA	18
2.1	Princípios	19
2.2	Exclusão da Obrigação de Licitar	20
2.3	Comissão de Licitação	21
2.4	Modalidades de licitação	21
2.5	Tipos de licitação.....	23
2.6	Fases da licitação.....	23
2.7	Audiência pública	25
2.8	Anulação e revogação	25
3	SISTEMAS MULTIAGENTE DE APOIO AS DECISÕES.....	27
3.1	Sistema de Apoio às Decisões.....	28
3.1.1	Decisão em Situação de Incerteza ou Risco	30
3.1.2	Decisão Com Risco, Com Múltiplos Objetivos e Múltiplos Cenários	30
3.1.3	Decisões com otimização	32
3.2	Sistemas Multiagente.....	34
3.2.1	Cooperação entre agentes	36
3.2.2	Arquitetura quadro-negro	36
3.2.3	Arquitetura de troca de mensagens.....	37
3.2.4	Arquitetura federativa.....	37
3.2.5	Coordenação entre agentes	37
3.2.6	Comunicação entre agentes	38
3.2.7	JADE.....	39
3.2.8	JADEx.....	39
3.2.9	JACK	39
3.2.10	Chimera.....	40
3.2.11	PROLOG	40
3.2.12	Trabalhos Correlatos.....	41
3.3	Considerações Finais	43
4	MADAE-PRO – UM PROCESSO BASEADO NO CONHECIMENTO PARA A ENGENHARIA DE DOMÍNIO E APLICAÇÕES MULTIAGENTE.....	44
4.1	A Fase de Engenharia dos Requisitos da Aplicação	46
4.2	A Fase de Projeto da Aplicação	47
4.3	A Fase de Implementação da Aplicação	48
4.4	A Metodologia MAAEM	48
4.4.2	O Modelo de objetivos.....	50
4.4.3	O Modelo de papéis	51
4.4.4	O Modelo de interações entre papéis.....	53
4.4.5	O Protótipo de interface com o usuário	54
4.4.6	O Modelo de conhecimento da sociedade multiagente	54
4.4.7	O Modelo da sociedade multiagente.....	55
4.4.8	O Modelo de interações entre agentes	56

4.4.9	O Modelo de cooperação e coordenação	57
4.4.10	O Modelo de conhecimento do agente	58
4.4.11	O Modelo de ações do agente	59
4.4.12	O Modelo de comportamentos.....	60
4.4.13	O Modelo de atos de comunicação	61
4.5	A ONTORMAS	62
4.6	Considerações Finais	64
5	MODELAGEM DO SISTEMA MULTIAGENTE NO PROCESSO DE LICITAÇÃO PÚBLICA (SMPLP).....	66
5.1	Fase de Análise da Aplicação.....	66
5.1.1	Modelagem de conceitos	66
5.1.2	Modelagem de objetivos.....	68
5.1.3	Modelagem de papéis	69
5.1.4	Modelagem de interações entre papéis	73
5.1.5	Protótipo da Interface com o Usuário	76
5.2	Fase de Projeto da Aplicação.....	83
5.2.1	Modelagem do conhecimento da sociedade multiagente	83
5.2.2	Modelagem da sociedade multiagente.....	85
5.2.3	Modelagem das interações entre agentes.....	87
5.2.4	Modelagem dos mecanismos de cooperação e coordenação	88
5.2.5	Modelagem do conhecimento do agente	89
5.2.6	Modelagem das ações do agente.....	92
5.3	Fase de Implementação da Aplicação	96
5.3.1	Modelagem de comportamentos.....	97
5.3.2	Modelagem de atos de comunicação	98
5.3.3	Implementação do SMPLP no JADE	99
5.4	Considerações Finais	103
6	CONCLUSÕES.....	104
6.1	Resultados e Contribuições da Pesquisa.....	104
6.2	Trabalhos Futuros	105
	REFERÊNCIAS	107
	APÊNDICES	114

1 INTRODUÇÃO

1.1 Relevância e motivação

A sobrevivência em um mercado de alta competitividade, em que as empresas cada vez mais buscam maximizar os lucros e minimizar os custos, é alicerçada pelo processo de tomada de decisões. No entanto, esse processo é complexo, uma vez que dado o excesso das informações existentes, é impossível analisar e tomar decisões com plena certeza.

Os Sistemas de Apoio às Decisões (SAD) sustentam o processo de tomada de decisões em todos os níveis organizacionais, oferecendo informações para solução de problemas de forma eficaz [90]. No entanto, para que um SAD seja capaz de apoiar as decisões de forma mais efetiva, faz-se necessário que o mesmo incorpore técnicas de Inteligência Artificial (IA), passando assim a ser conhecido como Sistemas de Apoio às Decisões Inteligente (SADI). Para implementar a inteligência de um SAD, é necessário um paradigma de desenvolvimento apropriado. É neste contexto que o paradigma baseado em agentes vem propor uma adequada abordagem (autonomia, abstração, decomposição e interações flexíveis) para lidar com a complexidade do processo de tomada de decisões.

A licitação pública destina-se a garantir a observância na Administração Pública do princípio constitucional da isonomia e a selecionar a proposta mais vantajosa. Seu processamento deverá ser realizado em estrita conformidade com os princípios básicos da legalidade, da impessoalidade, da moralidade, da igualdade, da publicidade, da probidade administrativa, da vinculação ao instrumento convocatório, do julgamento objetivo e dos que lhes são correlatos [58].

Tendo em vista a complexidade da Lei que regula o processo de licitação, devido às inúmeras possibilidades de condução do processo licitatório em diversos cenários e também a seu dinamismo, diante das constantes alterações na legislação, é de interesse o desenvolvimento de um Sistema Multiagente para o Processo de Licitação Pública (SMPLP) visando otimizar a tomada de decisões dos responsáveis pela aquisição dos bens e serviços da Administração Pública.

Um Sistema Multiagente (SMA) é um sistema composto por vários agentes que se comunicam e são coletivamente capazes de atingir objetivos que não seriam capazes de satisfazer separadamente. A complexidade destes sistemas é abordada através das interações entre os agentes, ou seja, cada agente pode executar, dentro de suas limitações, uma tarefa

simples, mas a boa coordenação da execução dessas tarefas simples por cada agente torna o sistema capaz de executar tarefas de grande complexidade [36].

Para fazer a especificação do SMPLP, adotou-se o MADAE-Pro [60], um processo que guia o desenvolvimento de um sistema multiagente através da realização da modelagem de objetivos, papéis, interações de entidades de uma organização e prototipação da interface com usuário.

1.2 Objetivos

1.2.1 Objetivo geral

Apoiar a tomada de decisões na Administração Pública, em particular, no processo licitatório, através da aplicação do paradigma computacional orientado a agentes.

1.2.2 Objetivos específicos

- Analisar o processo licitatório, suas atividades, fluxo de informações e atores envolvidos.
- Desenvolver um sistema multiagente de apoio ao processo de tomada de decisões nas licitações públicas com a utilização do processo de desenvolvimento de sistemas multiagente MADAE-Pro, elaborado no grupo GESEC.
- Avaliar as contribuições da utilização de representações baseadas no conhecimento para a melhora da eficácia do processo de tomada de decisões.

1.3 Organização da dissertação

Esta dissertação está estruturada em seis capítulos.

No segundo capítulo são apresentados os conceitos fundamentais do processo de licitação pública. Descreve os aspectos que suspendem o processo de licitação, como as modalidades e os tipos de licitações que servem como base para o julgamento do processo licitatório.

O terceiro capítulo descreve os conceitos básicos dos sistemas de apoio às decisões, como os ambientes nas quais essas decisões são tomadas. Descreve também o conceito fundamental dos sistemas multiagente.

No quarto capítulo é apresentada a especificação do processo MADAE-Pro. É apresentada também a ferramenta ONTORMAS que suporta as atividades de modelagem do MADAE-Pro.

O quinto capítulo descreve a modelagem do SMPLP utilizando a ferramenta ONTORMAS, a qual tem como objetivo guiar o desenvolvedor no desenvolvimento do sistema multiagente baseado nos requisitos levantados.

No sexto capítulo são apresentadas as considerações finais do trabalho, destacando os resultados obtidos e os trabalhos futuros que poderão ser desenvolvidos a partir desta pesquisa.

2 A LICITAÇÃO PÚBLICA

Para o Estado prestar serviços à sociedade, ele necessita adquirir bens e serviços. Mas, diferentemente das empresas privadas, os agentes públicos não podem adquirir bens e serviços em nome do Estado livremente. Para que isso ocorra, o art. 37, inciso XXI, da Constituição Federal, ressalvados os casos especificados na legislação, menciona que as obras, serviços, compras e alienações serão contratados mediante processo de licitação pública que assegure igualdade de condições a todos os concorrentes [31][58].

A Lei nº 8666, de 21 de Junho de 1993, regula o processo de licitação, estabelecendo normas gerais sobre licitações e contratos administrativos pertinentes a obras, serviços, inclusive de publicidade, compras, alienações e locações no âmbito dos Poderes da União, dos Estados, do Distrito Federal e dos Municípios. As obras, serviços, compras, alienações, concessões, permissões e locações da Administração Pública, quando contratadas com terceiros, serão necessariamente precedidas de licitação, ressalvadas as hipóteses previstas na lei.

A licitação é um certame que as entidades governamentais devem promover e no qual abrem disputa entre os interessados em com elas travar determinadas relações de conteúdos patrimoniais, para escolher a proposta mais vantajosa às conveniências públicas. De fato, visa selecionar a proposta mais vantajosa para a Administração Pública, garantindo que todos os possíveis interessados em ofertar seus bens e serviços ao poder público possam fazê-lo em condições de igualdade [70].

A licitação é “um processo administrativo porque, além da sucessão de atos e fases, há sujeitos diversos – os licitantes interessados no processo, que dele participam perante a Administração, todos, inclusive esta, tendo direitos, deveres, ônus, sujeições [67].”

Entender o procedimento da licitação como uma sequência de atos é compreender que cada ato subsequente vai depender de um ato anterior e assim será desde a adjudicação do objeto licitado até a autorização para a compra, obra ou serviço. Como procedimento, cada ato é independente e autônomo, podendo ser impugnado a partir do momento em que se aperfeiçoa. Ao se desmembrar o procedimento, é possível a identificação dos sujeitos que atuam na prática dos atos que o compõem, de modo a identificar a responsabilidade que cada agente traz consigo ao praticar determinado ato dentro do procedimento da licitação.

2.1 Princípios

A atividade administrativa vem norteada por uma série de princípios, que, mais do que diretrizes, são limites concretos à atuação dos agentes públicos, condicionando a validade do ato administrativo. Esses princípios constitucionais, expressos e implícitos, norteadores do direito administrativo, são: a supremacia do interesse público, legalidade, finalidade, razoabilidade, proporcionalidade, impessoalidade, publicidade, moralidade administrativa, eficiência .

O princípio da supremacia do interesse público proclama a superioridade do interesse da coletividade, firmando a prevalência dele sobre o do particular, como condição, até mesmo, da sobrevivência e asseguramento deste último. Isso não quer dizer que a Administração vá agir em interesse próprio, mas sim no interesse da coletividade.

O Art. 5º, inciso II da Constituição Federal, menciona que ninguém será obrigado a fazer ou deixar de fazer alguma coisa senão em virtude de lei. Para a Administração Pública, significa dizer que ela só pode fazer aquilo autorizado por lei, em contra partida, o particular, pode fazer tudo o que não está proibido na Lei [13].

A impessoalidade como princípio visa garantir igualdade para todos aqueles que se encontram em uma mesma situação. Este princípio nada mais é que o clássico princípio da finalidade, o qual impõe ao administrador público que só pratique o ato para o seu fim legal, o qual terá sempre um objetivo certo e inafastável de qualquer ato administrativo: o interesse público [69].

O princípio da publicidade vem como complemento e garantia de que estão sendo observados os princípios da legalidade e da impessoalidade. Salvo raras exceções, todos os atos da administração são públicos.

A moralidade como princípio não se confunde com a legalidade. Aqui se trata de uma conduta ética, pautada por princípio de lealdade e boa-fé. Este princípio está implícito no art. 5º, inciso LXXIII, da Constituição Federal, que prevê ação popular tendente a anular ato lesivo à moralidade administrativa.

O princípio da eficiência foi inserido pela EC nº19/98, no âmbito da reforma administrativa, onde se buscava um enfoque menos burocrático e mais gerencial. Embora se busque o resultado, se engana quem pensa que em nome do princípio da eficiência pode-se passar por cima dos outros princípios, como, por exemplo, não observar o princípio da legalidade. Deseja-se uma Administração eficiente, mas que respeite os outros princípios. O

agente público deve ser eficiente, sem perder de vista a moralidade administrativa, sem ultrapassar os limites legais, dando publicidade de suas ações [24].

O Princípio da razoabilidade é princípio geral do direito que obriga o administrador a adotar critérios objetivos, seja na elaboração do ato normativo, seja na interpretação do fato, seja na aplicação da norma, isto é, sua conduta deve ser afastada de preconceitos, ideologia e não deve conter favoritismos.

No art. 3º do Estatuto das Licitações, Lei 8.666/93, o legislador ordinário, deixou expresso que os agentes públicos que participam de um processo de licitação devem observar os princípios da isonomia, da legalidade, da impessoalidade, da moralidade, da igualdade, da publicidade, da probidade administrativa, da vinculação ao instrumento convocatório, do julgamento objetivo e dos que lhes são correlatos [58].

Para garantir uma justa competição entre todos aqueles que vão participar do certame licitatório, segue-se o princípio da isonomia, o qual tem por objetivo tratar todos que estão em uma mesma situação de uma mesma forma, sem distinção.

O princípio da probidade administrativa está intimamente ligado ao princípio da moralidade, o qual traz para os agentes públicos o dever de conduta ética e moral, observando as regras da boa administração.

A vinculação ao instrumento convocatório, como princípio, torna obrigatório, não só para os licitantes, mas, também, para a Administração, o dever de observar as regras fixadas no edital ou no convite. Com isso, se evita surpresa aos licitantes, que sabem, desde o lançamento do instrumento convocatório, das regras da disputa, possibilitando-lhes, assim, um tratamento isonômico.

O princípio do julgamento objetivo obriga a administração a escolher a proposta de acordo com os critérios traçados no edital e na legislação correlata, retirando da estrita discricionariedade do agente público a escolha da proposta mais vantajosa para a Administração.

2.2 Exclusão da Obrigação de Licitar

As hipóteses previstas na Lei 8.666/93, em seu art. 24 e 25, que suspendem o processo de licitação são: inexigibilidade de licitação e dispensa. A licitação é inexigível quando houver inviabilidade de competição, em especial: para aquisição de materiais, equipamentos, ou gêneros que só possam ser fornecidos por produtor, empresa ou representante comercial exclusivo, vedada a preferência de marca; para a contratação de

serviços técnicos, de natureza singular, com profissionais ou empresas de notória especialização, vedada a inexigibilidade para serviços de publicidade e divulgação; para contratação de profissional de qualquer setor artístico, diretamente ou através de empresário exclusivo, desde que consagrado pela crítica especializada ou pela opinião pública.

Alguns dos casos de dispensa são: obras e serviços de engenharia, se o valor da contratação for de até R\$ 15.000,00, sendo que para outros serviços e compras até o limite de R\$ 8.000,00; nos casos de guerra ou grave perturbação da ordem; emergência ou de calamidade pública. Assim, caso alguma dessas opções sejam satisfeitas na hora da Administração adquirir um bem ou serviço, o processo de licitação não será executado, do contrário, o processo iniciará através de uma cotação de preço do mercado, a qual servirá para definir a modalidade de licitação [58].

2.3 Comissão de Licitação

O art. art. 6º, inciso XVI, traz na definição de comissão a sua função. Nos termos do referido normativo, considera-se “Comissão” – comissão permanente ou especial, criada pela Administração com a função de receber, examinar e julgar todos os documentos e procedimentos relativos às licitações e ao cadastramento de licitantes [58].

O art. 51, menciona:

A habilitação preliminar, a inscrição em registro cadastral, a sua alteração ou cancelamento, e as propostas serão processadas e julgadas por comissão permanente ou especial de, no mínimo 3 (três) membros, sendo pelo menos 2 (dois) deles servidores qualificados pertencentes aos quadros permanentes dos órgãos da Administração responsáveis pela licitação [58].

Logo, a comissão permanente tem a responsabilidade de dirigir a licitação autorizada, a qual receberá os documentos referentes à habilitação das empresas participantes como suas propostas e realizará seu julgamento.

Devido o alto grau de responsabilidade da comissão permanente, seus membros respondem solidariamente por todos os atos praticados, exceto se ressaltarem posição individual, que deverá vir registrada em ata.

2.4 Modalidades de licitação

A Lei 8.666/93, em seu art. 22, estipula como modalidades de licitação: a concorrência, a tomada de preços, o convite, o concurso e o leilão. As modalidades de licitação: concorrência, tomada de preços e convite serão determinadas de acordo com o valor

estimado daquilo que se quer comprar. Devido a essa necessidade, após a comissão permanente aprovar o projeto de compra, a mesma fará uma cotação de preço no mercado para obter a média do(s) produto(s) a serem licitado(s).

Uma vez com a média dos preços em sua posse, a Lei menciona que para as compras de maior vulto, obras e serviços de engenharia acima de R\$1.500.000,00 e compras e outros serviços acima de R\$ 650.000,00, concorrência; para as intermediárias, obras e serviços de engenharia de até R\$1.500.000,00 e compras e outros serviços de até R\$ 650.000,00, tomada de preços; para as de menor vulto, obras e serviços de engenharia de até R\$ 150.000,00 e compras e outros serviços de até R\$ 80.000,00, convite. No caso em que couber convite, a Administração poderá utilizar tomada de preços e, em qualquer caso, a concorrência [58].

Para modalidade concurso, a Administração somente a utiliza quando existe a necessidade de se contratar algum trabalho intelectual (técnico, artístico ou científico), onde o valor a ser pago ao vencedor é fixado de antemão pelo órgão.

Quando a Administração Pública deseja vender produtos que não lhe servem mais, utiliza-se a modalidade leilão, o qual será vencido por aquele que der o maior lance no bem ofertado.

Além das modalidades de licitação regidas pela lei 8.666/93, o governo brasileiro criou a modalidade de licitação pregão, através da lei federal 10.520/2002, instituída inicialmente pela Medida Provisória 2.026, de 4 de maio de 2000, com o objetivo de acelerar o processo de aquisição da Administração Pública, realizando contratos administrativos de bens e serviços comuns, independentemente do valor estimado. Essa modalidade foi criada com o objetivo de aumentar a quantidade de participantes e baratear o processo licitatório [56][68].

A modalidade pregão pode ser realizada tanto através do pregão eletrônico como do pregão presencial. O “pregão eletrônico” permite a participação de várias empresas de diversos estados, na medida em que dispensa a presença dos contendentes, sendo uma modalidade ágil, transparente e que possibilita uma negociação eficaz entre os licitantes. Já o “pregão presencial”, obriga os participantes a comparecerem à negociação, liderados por um pregoeiro devidamente designado pelo órgão da Administração Pública licitante.

2.5 Tipos de licitação

Para o julgamento formais das propostas, deve-se observar o tipo de licitação previsto no edital, regido pela lei 8.666/93, que são:

- Menor preço;
- Melhor técnica;
- Técnica e preço;
- Maior lance ou oferta.

O tipo de licitação menor preço, é aquela em que o fato decisivo é o menor preço, em termos absolutos, não se cogitando qualidade, rendimento, produtividade, prazo de entrega, condições de pagamento ou outro fator.

Já o tipo de licitação melhor técnica é aquela em que o fato de julgamento é uma das melhores tecnologias adotadas pelo proponente da execução do objeto. É própria para as licitações em que se quer a tecnologia mais moderna, dentro dos recursos financeiros destinados para tanto.

O tipo de licitação técnica e preço é aquela em que, após a classificação das propostas técnicas, abrem-se de seus proponentes os envelopes contendo as propostas comerciais e delas faz-se a avaliação e valoração de acordo com os critérios consignados no edital (pesos), sendo que com os classificados far-se-á a ordenação final de acordo com a média aritmética ponderada das valorações técnicas e de preço [25][31].

Quando a Administração Pública deseja conceder as permissões de uso de bens ou serviços públicos, ou locação dos mesmos, utiliza-se a licitação do tipo maior lance ou oferta, onde será vencedora a proposta que fizer a maior oferta.

2.6 Fases da licitação

O procedimento da licitação inicia-se na repartição interessada com a abertura de processo em que a autoridade competente determina sua realização, define seu objeto e indica os recursos hábeis para a despesa [69]. Enquanto os atos são praticados exclusivamente no âmbito da Administração, entende-se que os mesmos fazem parte da fase interna, mas, no momento que a Administração convoca os terceiros, fornecedores de bens e serviços, a participarem do certame, abre-se a fase externa.

A fase interna é composta das seguintes etapas:

- Solicitação expressa do setor requisitante interessado, com indicação de sua necessidade;
- Elaboração do projeto básico e, quando for o caso, o executivo;
- Aprovação da autoridade competente para início do processo licitatório, devidamente motivada e analisada sob a ótica da oportunidade, conveniência e relevância para o interesse público;
- Autuação do processo correspondente, que deverá ser protocolizado e numerado;
- Elaboração da especificação do objeto, de forma precisa, clara e sucinta, com base no projeto básico apresentado;
- Estimativa do valor da contratação, mediante comprovada pesquisa de mercado;
- Indicação dos recursos orçamentários para fazer face à despesa;
- Verificação da adequação orçamentária e financeira, em conformidade com a lei de Responsabilidade fiscal, quando for o caso;
- Elaboração de projeto básico, obrigatório em caso de obras e serviços;
- Definição da modalidade e do tipo de licitação a serem adotados.
- Aprovação do edital pelo jurídico.

Já a fase externa é composta das seguintes etapas:

- Publicidade do instrumento Convocatório;
- Recebimento das propostas: abertura dos envelopes contendo a documentação relativa à habilitação dos concorrentes e sua apreciação;
- Concessão de prazo para recurso aos inabilitados;
- Devolução dos envelopes fechados aos concorrentes inabilitados, contendo as respectivas propostas;
- Abertura dos envelopes contendo as propostas dos concorrentes habilitados;
- Julgamento e classificação das propostas de acordo com os critérios de avaliação constantes do edital;
- Deliberação da autoridade competente quanto à homologação e adjudicação do objeto da licitação.

2.7 Audiência pública

Para despesas de grande vulto, cujo valor ultrapasse 100 (cem) vezes o valor para obras e serviços de engenharia na modalidade concorrência, exige-se mais do que um exame balizado de oportunidade e conveniência do agente público. É necessário que o lançamento do edital seja precedido de consulta popular, a denominada audiência pública.

Art. 39 da Lei 8.666/93 reza:

Sempre que o valor estimado para uma licitação ou para um conjunto de licitações simultâneas ou sucessivas for superior a 100 (cem) vezes o limite previsto no art. 23, inciso I, alínea “c” desta Lei, o processo licitatório será iniciado, obrigatoriamente, com uma audiência pública concedida pela autoridade responsável com antecedência mínima de 15 (quinze) dias úteis da data prevista para a publicação do edital, e divulgada, com a antecedência mínima de 10 (dez) dias úteis de sua realização, pelos mesmos meios previstos para a publicidade da licitação, à qual terão acesso e direito a todas as informações pertinentes e a se manifestar todos os interessados.

Para evitar a audiência pública, o legislador pode tentar “dividir” as licitações, lançando-as simultâneas e sucessivamente. No entanto, mesmo que o administrador divida o objeto e realize mais de um certame, se a similitude do objeto e o lapso temporal caracterizarem licitação simultânea ou sucessiva, deverá realizar a audiência pública, se a soma dos objetos licitados atingir o limite estabelecido no normativo supra.

2.8 Anulação e revogação

O ato de anulação e o de revogação da licitação, somente poderá ser feito por razões de interesse público, estando devidamente comprovado. No caso de anulação, deve-se apontar qual ou quais ilegalidades insanáveis que levaram a realização da mesma, sendo que para revogação, deve-se evidenciar qual o motivo que desqualificou a proposta vencedora como a mais vantajosa para a Administração Pública, ou que impossibilita a contratação naquele momento.

Art. 49 da Lei 8.666/93 reza:

A autoridade competente para aprovação do procedimento somente poderá revogar a licitação por razões de interesse público decorrente de fato superveniente devidamente comprovado, permanente e suficiente para justificar tal conduta, devendo anulá-lo por ilegalidade, de ofício ou por provocação de terceiros, mediante parecer escrito e devidamente fundamentado [58].

A anulação do procedimento licitatório por motivo de ilegalidade não gera obrigação de indenizar. No entanto, quando existe motivo para anular a licitação, o contratado deverá ser indenizado pelo que já foi prestado, desde que ele não seja o responsável pela

nulidade do processo, sendo que a Administração também deverá ressarcir-lo pelas perdas e danos [58].

Quando a licitação é autorizada, a autoridade competente faz um juízo de valor, elencando os motivos para a realização dessa. Ao final, faz-se uma nova verificação a fim de ver se a contratação ainda atende ao interesse público. Caso constate que a mesma está ultrapassada, como, por exemplo, o surgimento de uma nova tecnologia, poderá revogar a licitação.

Sendo comprovada verdadeiramente a inconveniência da celebração do contrato, a Administração Pública revogará a licitação, tendo dever de indenizar a quem for de direito. Caso o erro da avaliação tenha sido do administrador, a responsabilidade há de recair sobre a autoridade que tem competência para revogar a licitação, e poderá ela vir a suportar regressivamente os danos sofridos pela administração, sem prejuízos das demais responsabilidades já apresentadas, especialmente multa por ato de gestão ilegítimo ou antieconômico de que resulte prejuízo ao erário [57].

3 SISTEMAS MULTIAGENTE DE APOIO AS DECISÕES

Todos os dias, as pessoas precisam tomar decisões em suas vidas pessoais (acordar, trabalhar, estudar, namorar, ir ao cinema, farmácia, etc), e cada decisão tomada gera um impacto e um aprendizado. Nas organizações, tomar decisões é muito mais complexo, pois além de envolver todos os níveis funcionais da organização, envolve diversos colaboradores, com suas percepções variadas acerca do mesmo problema.

Para solucionar os diversos problemas existentes, é preciso que a organização faça uma análise do problema, verificando as decisões anteriormente tomadas, ou mesmo um novo estudo e qual será a melhor decisão a ser tomada para sua resolução. Para isso, é preciso saber se esse problema é estruturado, semiestruturado, ou não-estruturado, e a que nível da organização o mesmo pertence.

Uma das tecnologias que tem se tornado imprescindível no processo de tomada de decisões nas organizações devido à alta complexidade dos problemas que a mesma possui, como, riscos, incerteza, múltiplos objetivos, são os Sistemas de Apoio às Decisões (SAD), que visam processar os dados e gerar informações para aperfeiçoar o processo de tomada de decisões.

Para melhorar a efetividade de um SAD, usamos o que hoje é conhecido como sistema de conhecimento ou Sistema Baseado em Conhecimento (SBC), que são programas de computador que usam conhecimento representado explicitamente para resolver problemas que requerem uma quantidade razoável de conhecimento humano.

E devido à necessidade de solucionar decisões de extrema complexidade em campos específicos, surge a figura do Sistema Especialista (SE), que é um tipo de sistema baseado no conhecimento especialmente projetado para emular a especialização humana de algum domínio específico. Ele possui uma Base de Conhecimento (BC) formada de fatos, regras e heurísticas sobre o domínio, tal como um especialista humano faria, e deve ser capaz de oferecer sugestões e conselhos aos usuários e, também, adquirir novos conhecimentos e heurísticas com essa interação.

Além do SE, que é limitado por trabalhar exclusivamente dentro do universo para o qual foi desenvolvido, temos também a figura do agente computacional, que é uma entidade autônoma que percebe seu ambiente através de sensores e age sobre o mesmo utilizando-se dos executores. Um agente possui uma memória interna que irá atualizar-se com a chegada de novas percepções. Essa memória é utilizada nos procedimentos de tomada de decisão, os quais irão gerar ações para serem executadas. De forma a manter um histórico do

comportamento do agente, a memória do agente é também atualizada a partir da ação selecionada.

Esse capítulo tem por objetivo geral analisar os sistemas baseados em conhecimento para apoio à tomada de decisões. Como objetivos específicos, fazer um estudo do estado da arte dos sistemas baseados em conhecimento, do processo de tomada de decisões, como um estudo dos agentes racionais no processo de tomada de decisões.

3.1 Sistema de Apoio às Decisões

Um Sistema de Apoio às Decisões (SAD) é um conjunto de procedimentos (programas de softwares) baseados em modelos para processar dados e analisar problemas, tendo como finalidade dar assistência aos administradores no processo de decisão. Os problemas resolvidos por um SAD podem ser semi-estruturados, não estruturados ou estruturados, conforme o ponto de vista da(s) pessoa(s) quem toma(m) as decisões [90].

Os SAD's surgiram devido a crescente competição que as empresas enfrentavam no país e no exterior e as dificuldades em acompanhar as numerosas operações empresariais. Para sanar essas dificuldades um SAD é desenvolvido com o objetivo de ser um sistema flexível e expansível, capaz de dar apoio a análise *ad hoc* dos dados e dos modelos de decisão, procurando obter resultados eficazes no planejamento de longo, médio e curto prazo.

Um SAD é formado por três componentes interativos:

- Sistema de linguagem: que permite a comunicação entre os usuários e os subsistemas do SAD. No SMPLP, a comunicação é realizada usando KQML e a interface do usuário, apresentado na Seção 3.2.6 e 5.1.5;
- Sistema de conhecimento: responsável pelo processamento da base de conhecimento através dos mecanismos de inferências. No SMPLP, é utilizado o PROLOG como mecanismo de inferência para processar a base de conhecimento, composta de fatos e regras, apresentado na seção 3.2.11;
- Sistema para processar problemas: estrutura e executa os modelos para resolver os problemas apresentados. No SMPLP, são utilizados os agentes, os quais interagem entre si, possuem um comportamento autônomo, o que proporciona a resolução de problemas com maior efetividade, apresentado na seção 5.3.3.

Um SAD caracteriza-se por [93]:

- Direcionar-se, principalmente, para problemas não estruturados, que ocorrem com mais frequência nos altos níveis gerenciais;

- Combinar a utilização de técnicas de modelagens e analíticas, com funções tradicionais de acesso e recuperação de informações;
- Ter sua interface construída com os usuários visando, principalmente, a facilidade de uso;
- Enfatizar a flexibilidade e adaptabilidade para acompanhamento das mudanças, tanto do ambiente, quanto nas diferentes necessidades de utilização por parte dos usuários.

Diferente das outras abordagens de Sistemas de Informação, onde é preciso entender apenas como operar o sistema, num SAD o conhecimento de métodos de apoio às decisões é imprescindível para sua correta utilização [1]. Vários autores sugerem que os tomadores de decisões atuem de forma integrada com um analista de decisões ou de pesquisa operacional [7][93].

Para que o SAD auxilie no processo de tomada de decisão é preciso ser composto de uma base de dados (que auxilia o sistema), uma base de modelos (que provê a capacidade de análise) e o diálogo (que provê a interação entre o usuário e o sistema).

Na base de dados, estão presentes os dados internos e os externos, que podem ser acessados diretamente pelo usuário ou entram como input para a base de modelos. Os dados internos referem-se ao conhecimento a priori de gerentes e aos dados provenientes dos sistemas transacionais da organização. Já os dados externos são necessários especialmente quando as decisões são dos níveis mais altos da organização, sendo que a base de dados deverá fornecer as informações de forma rápida e permitir sua manipulação de forma eficiente [43].

A base de modelos provê a capacidade de análise para o SAD, onde os algoritmos servem para gerar informação para auxiliar a tomada de decisão usando uma representação matemática do problema.

O diálogo pode ser entendido como uma combinação de software, hardware e pessoas, que permite uma interação entre o usuário e o SAD. Trata-se do componente mais importante para o usuário, pois para ele o diálogo é a forma como o sistema opera e funciona.

No desenvolvimento de um SAD, a abordagem evolutiva tende a ser extremamente relevante, pelo fato de o sistema partir de uma versão menor do problema e evoluir até uma versão mais abrangente [93]. A ideia é reduzir o tamanho do problema para que o SAD possa atender de início a uma parte do problema e assim ter o seu desenvolvimento evolutivo à medida que a equipe aprende mais sobre a natureza do problema não estruturado.

3.1.1 Decisão em Situação de Incerteza ou Risco

Em muitos problemas de decisão, as alternativas que compõem um cenário ocorrem com certo nível de possibilidade ou probabilidade denominado risco, o qual é medido pelo resultado alcançado ao se tomar uma decisão, isto é, diz-se que a decisão foi um sucesso quando se alcançou o objetivo esperado, do contrário, um fracasso.

A decisão tomada sob-risco é aquela na qual pode haver dois ou mais resultados possíveis para cada alternativa de ação devido a impossibilidade de controle dos estados da natureza por parte do tomador de decisão. Se for assumido que ele conhece ou pode estimar a probabilidade de ocorrência dos estados da natureza, então é dito que a decisão é tomada sob risco [96].

O risco trata de acontecimentos futuros e está presente na vida de todos em diversas ocasiões: sem assumir riscos é impossível um progresso econômico e até mesmo a manutenção do *status quo* [71].

Ao tomar uma decisão de maior risco, isso não quer dizer que o tomador de decisão simplesmente ignorou o risco dos resultados possíveis, mas ele levou em conta outras considerações que aparentemente podem ter sido mais atrativas diante do risco existente.

Para tomar uma decisão em situação de risco é necessário informação, tempo, bem como recursos. A informação é necessária para conhecer melhor o impacto no resultado das várias opções de escolha; para obter as informações, é necessário tempo para aguardar até a ocorrência do evento incerto e escolher a melhor alternativa; e obviamente que sem recursos, será impossível obter as melhores informações como aguardar o tempo necessário [65].

3.1.2 Decisão Com Risco, Com Múltiplos Objetivos e Múltiplos Cenários

A percepção do risco é um processo complexo e subjetivo, isto porque dois componentes influenciam diretamente nossas percepções: o fator medo e o fator controle. Por exemplo, a percepção do risco é maior em um acidente nuclear devido ao fator medo, ou é maior para um passageiro de um carro do que para o motorista, pois a diferença está na percepção do controle [90].

O componente maior da percepção de risco está em como percebemos perda e ganho. Quando estão em posição de ganho, as pessoas têm cada vez mais aversão ao risco porque desejam manter seus ganhos, enquanto as pessoas que estão em posição de perda tornam-se mais procuradoras de risco, porque não possuem muita coisa a perder.

De fato, o que percebemos como perda ou ganho depende do ponto de referência pessoal, que por sua vez, se altera de acordo com a situação e no decorrer do tempo. Por exemplo, um motorista de um carro, viaja a uma velocidade de 160km/h enquanto é solteiro e não tem filhos. Mas, ao se casar e constituir família, sua percepção de risco muda, fazendo com que o mesmo viaje no máximo a uma velocidade de 110km/h.

Devido a essa abordagem, vemos que o processo de tomada de decisão é complexo e depende tanto das características pessoais do tomador de decisões quanto da situação em que está envolvido e da maneira como percebe essa situação. De fato, tomar uma decisão é escolher entre alternativas exequíveis, aquelas que atendam o objetivo e superem as restrições dos problemas.

Essa questão da tomada de decisão complexa, é demonstrada numa pesquisa realizada por Gary Klein, psicólogo cognitivo - um cartógrafo da mente humana, que mapeia como as pessoas percebem e observam, pensam e raciocinam, agem e reagem. Ele é diretor da Klein Associates e autor do livro *Sources of Power - How People Make Decisions* (Fontes de Poder - Como as Pessoas Tomam Decisões, ainda sem tradução no Brasil). Klein escreveu o livro baseado nos depoimentos de pessoas que, sob altíssima pressão, tem que tomar decisões de vida ou morte, como: enfermeiras de Unidades de Terapia Intensiva (UTI's), pilotos de combate (caças e helicópteros), bombeiros e até pilotos de tanques de guerra M-1.

O que sempre impressionou Klein foi que essas pessoas estudavam muito, às vezes por anos, e na hora de tomar decisões, jogavam os manuais no lixo e seguiam sua intuição. Também notou que as mentes de pessoas treinadas decidiam rapidamente durante os momentos de pressão que, depois, raramente conseguiam explicar racionalmente porque aquela decisão havia sido tomada, e não uma outra qualquer.

Um caso prático típico foi dado a Klein por um piloto da Força Aérea. Quando começou a pilotar, esse piloto estava sempre aterrorizado. Se cometesse um erro, certamente morreria. Tinha que seguir regras e listas de procedimentos para voar corretamente. Isso era tremendamente estressante. Mas, em algum ponto do seu desenvolvimento, uma mudança profunda ocorreu. De repente, não era mais ele voando dentro de um avião - era ele que voava. Todos os procedimentos haviam sido internalizados de tal forma que agora faziam parte da sua personalidade. Regras já não eram mais necessárias.

Klein menciona que com o passar do tempo acumulamos experiência e, subconscientemente, criamos categorias de eventos de acordo com a forma como devemos reagir. Quando você encontra uma determinada situação, seu cérebro revê seu catálogo de lembranças e procura algo que se assemelhe o máximo possível com o que está ocorrendo.

Com base nisso, o cérebro decide imediatamente por uma determinada ação ou estratégia em milionésimos de segundo.

Visto desta forma, a intuição é na verdade um aprendizado - temos que aprender a ver e reconhecer rapidamente o que está acontecendo.

Diz a teoria clássica da Tomada de Decisões que você deve:

- a) Identificar opções.
- b) Avalia-as.
- c) Dar notas (prós e contras).
- d) Escolher a maior nota.

Mas é justamente o contrário que ocorre - em situações extremas, as pessoas entrevistadas dizem nem pensar em opções - elas simplesmente sabem qual é a resposta certa. O que elas fazem é, uma vez decidido o curso de ação, controlam se tudo está correndo como previsto, ou se novas consequências, inesperadas, surgem. Neste caso, é preciso tomar uma nova decisão - e o ciclo ocorre novamente.

Também se achava que os experts, na hora de decidir, deliberassem cuidadosamente os méritos de cada ação, e que os novatos decidissem impulsivamente pela primeira opção que aparecesse, mas, é justamente o contrário que ocorre. São os novatos que tem que comparar diferentes ângulos da questão para resolver um problema. Um expert cria rapidamente uma solução e, rapidamente também, decide se esse plano vai funcionar ou não. Um expert é rápido porque faz menos.

Às vezes, pensamos que um expert fica mais lento pelo excesso de informação, fatos, lembranças - que uma decisão seria muito lenta porque tanta informação tem que ser analisada. Mas é justamente o contrário. O acúmulo de experiências não atrapalha - ele torna as pessoas mais leves. E mais rápidas. E, principalmente, na hora de decidir, faz com que confiem completamente na sua intuição - a melhor ferramenta que até hoje temos disponível.

3.1.3 Decisões com otimização

O desenvolvimento tecnológico e o aumento da competitividade faz com que a tomada de decisões, na maioria das vezes, se fundamente na análise das diversas situações, tendo como objetivos, por exemplo, obter o lucro máximo ou então minimizar as despesas e desperdícios.

A complexidade destes processos de otimização, dar-se devido ao elevado número de variáveis e as limitações ou restrições das mesmas. Em termos matemáticos, a resolução

destes problemas conduz obrigatoriamente a sistemas de equações e inequações com elevado número de variáveis, sendo necessário o recurso a sistemas informáticos e processos computacionais.

O desenvolvimento tecnológico faz com que estes processos sejam mais rápidos e possuam maiores capacidades, proporcionando um aprofundamento e ampliação do conhecimento matemático nesta área.

Devido a isso, a Programação Linear (PL) é uma das técnicas da Pesquisa Operacional¹ das mais utilizadas com problemas de otimização, pois buscam a distribuição eficiente de recursos limitados para atender um determinado objetivo, como é necessário também que se defina quais as atividades que consomem recursos e em que proporções os mesmos são consumidos, pois o que se busca, num problema PL é a função objetivo, isto é, a maximização do lucro ou a minimização dos custos. A essa solução dá-se o nome de solução ótima.

A solução ótima (quando ela existe) sempre será um ponto de “quina” ou um ponto extremo do conjunto convexo definido pelas restrições. De modo que a busca da solução ótima é um problema combinatório para encontrar a combinação de restrições que define o ponto ótimo. O método do simplex (e suas variantes) é o utilizado pela maioria dos programas de computador para a solução de problemas de PL.

O método Simplex é o algoritmo que sistematiza o processo de obtenção da solução ótima para qualquer número de variáveis e equações de restrição. É um método consagrado tanto na comunidade científica como na empresarial [90].

Quanto ao Solver do Excel, se nas opções indicarmos que o problema é linear (o que é recomendável), ele utiliza o simplex caso contrário, um algoritmo de busca por gradiente, que não é o simplex. O Solver é menos amigável para usar e para interpretar seus resultados, mas está disponível para todos os usuários de Excel e ele pode resolver problemas não lineares, e por ser uma ferramenta do Excel, traz facilidades oriundas do ambiente da planilha.

Quando ocorrem equações não lineares, o problema de otimização deve ser resolvido por Programação Não Linear ou Programação Dinâmica, cuja resolução em geral é bastante trabalhosa [90].

¹ A Pesquisa Operacional (PO) trata da modelagem matemática de fenômenos estáticos (determinísticos) ou dinâmicos (estocásticos).

A dificuldade para resolver problemas de programação não linear ocorre mesmo quando o número de variáveis é pequeno. Se a função objetivo não for linear, podemos ter o caso de otimização com vários pontos de máximo ou de mínimo. Se uma das equações de restrição (ou todas) for não linear, podemos ter uma região de busca da solução descontínua ou não convexa. Os casos de otimização não linear exigem algoritmos que utilizam aproximação linear das equações, ou devem ser resolvidos por algoritmos heurísticos.

3.2 Sistemas Multiagente

Um agente é tudo o que pode ser considerado capaz de perceber seu ambiente por meio de sensores e de agir sobre esse ambiente por intermédio de atuadores. Um agente humano tem olhos, ouvidos e outros órgãos como sensores, e tem mãos, pernas, boca e outras partes do corpo que servem como atuadores. Um agente robótico poderia ter câmeras e detectores da faixa de infravermelho funcionando como sensores e vários motores como atuadores. Um agente de software recebe sequências de teclas digitadas, conteúdo de arquivos e pacotes de rede como entradas sensoriais e atua sobre o ambiente exibindo algo na tela, gravando arquivos e enviando pacotes de rede [74].

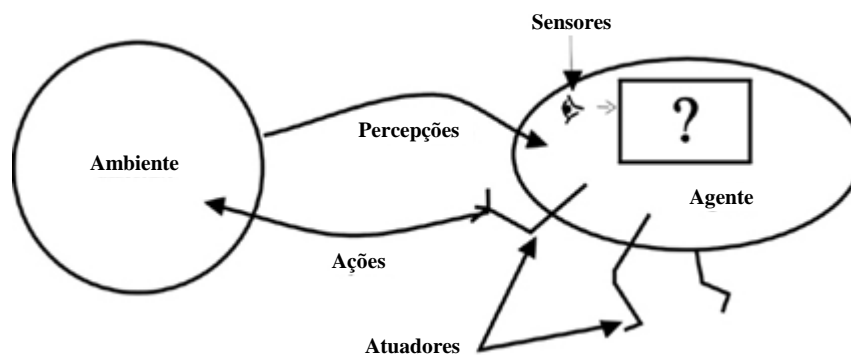


Figura 1 – Agente Genérico interagindo com seu ambiente [84]

A interrogação existente na Figura 1 define o mapeamento das percepções para as ações, definindo o tipo do agente:

- Reativo - é o agente do tipo condição ação ou do tipo se então, mapeamento direto das percepções para as ações;
- Deliberativo - capaz de realizar um planejamento de suas ações e se utiliza de um mecanismo de inferência.

A principal característica do agente é a autonomia, devido à capacidade dele atuar sem a intervenção humana ou de outros sistemas através do controle que ele possui do seu

estado interno e do seu comportamento [36]. Possui também o constante monitoramento do ambiente e sociabilidade, que são características de fundamental importância no processo de tomada de decisões num ambiente complexo e semi-estruturado que é o da licitação.

Quando o sistema se torna muito complexo, um agente não é capaz de executar todas as tarefas com a máxima efetividade possível. Devido isso, surge o Sistema Multiagente (SMA), que é um sistema composto por vários agentes que se comunicam e são coletivamente capazes de atingir objetivos que não seriam capazes de satisfazer separadamente. Em tais sistemas, a complexidade deve emergir das interações entre os agentes, ou seja, cada agente pode executar, dentro de suas limitações, uma tarefa simples, mas a boa coordenação da execução dessas tarefas simples por cada agente torna o sistema capaz de executar tarefas de grande complexidade [9].

O paradigma de agentes simplifica o desenvolvimento de sistemas complexos porque os agentes podem decidir sobre o tipo e escopo das suas interações em tempo de execução. Os agentes são capazes de iniciar uma interação e responder a outros agentes de maneira autônoma e flexível. Portanto, a abordagem dos SMA é adequada para a modelagem de sistemas, porque os agentes são capazes de reagir a eventos não previstos, explorando as oportunidades que surgem e realizando dinamicamente acordos com outros agentes cuja presença pôde não ter sido prevista no projeto inicial [36].

A arquitetura de um SMA mostra a maneira como o sistema está implementado em termos de propriedades e estrutura, e como os agentes que o compõem podem interagir a fim de garantir a funcionalidade do sistema.

De acordo com a sua complexidade, a arquitetura de um sistema multiagente pode ser classificada em três grupos [54]:

- Arquitetura simples: quando é composta por um único e simples agente.
- Arquitetura moderada: quando é composta por agentes que realizam as mesmas tarefas, mas possuem diferentes usuários e podem residir em máquinas diferentes;
- Arquitetura complexa: quando é composta por diferentes tipos de agentes, cada um com certa autonomia, podendo cooperar e estar em diferentes plataformas.

Além disso, as arquiteturas dos sistemas multiagente também podem ser classificadas segundo os mecanismos de cooperação e coordenação utilizados na sociedade.

3.2.1 Cooperação entre agentes

Nos sistemas multiagente, uma política de cooperação se faz necessária, uma vez que é através deste mecanismo que os agentes expressam suas necessidades a outros agentes a fim de realizar uma determinada tarefa. O mecanismo de cooperação entre agentes visa determinar a maneira como os agentes expõem suas necessidades a outros agentes para atingirem determinados objetivos. Esse processo pode acontecer de duas maneiras: partilha de tarefas e partilha de resultados. A primeira caracteriza-se pela necessidade de agentes auxiliares durante a execução de uma tarefa por um determinado agente, enquanto que na segunda, os agentes disponibilizam informações para a sociedade, prevendo que algum outro agente possa necessitar dela em determinado momento [36].

Durante a realização de tarefas, na grande maioria das vezes, os agentes necessitam da ajuda dos outros agentes da sociedade, ou de forma contrária, auxiliam os outros para que o objetivo geral do sistema possa ser alcançado. Devido a isso, é extremamente necessário que se localize determinado agente dentro da sociedade para que a cooperação possa ser efetuada.

As principais abordagens de arquiteturas de SMA, onde a cooperação está em destaque são: a arquitetura quadro-negro, a arquitetura de troca de mensagens e a arquitetura federativa, apresentadas a seguir.

3.2.2 Arquitetura quadro-negro

O quadro-negro é uma estrutura de dados persistente onde existe uma divisão em regiões ou níveis, visando facilitar a busca de informações. Ele é um meio de interação entre os agentes, onde estes escrevem e leem mensagens que serão usadas para atingir o objetivo do sistema. Pode-se assim dizer que um quadro-negro é uma memória de compartilhamento global onde existe uma quantidade de informações e conhecimento usados para leitura e escrita pelos agentes [36].

Em uma sociedade baseada na arquitetura quadro-negro os agentes não se comunicam entre si de maneira direta, mas sempre através do quadro, devido ao fato dele ser usado como um repositório de perguntas e respostas. Assim, os agentes que necessitam de alguma informação escrevem seu pedido no quadro e ficam à espera que outros agentes respondam à medida que acessem o mesmo.

Tendo em vista a lentidão do processo de gravação e recuperação das mensagens pelos agentes com o uso da arquitetura quadro-negro, o mesmo se torna inviável para sistemas que precisem da resposta em tempo real.

3.2.3 Arquitetura de troca de mensagens

Nesta arquitetura, os agentes se comunicam diretamente, uns com os outros, através de mensagens assíncronas, onde cada agente sabe os nomes e endereços de todos os agentes que formam o sistema.

Para que as trocas de mensagens ocorram de maneira adequada entre os agentes é necessário estabelecer um protocolo de conversação, o qual dita às regras e impõe o formalismo necessário para que as mensagens sejam encaminhadas e compreendidas pelos agentes [36].

3.2.4 Arquitetura federativa

Quando se tem um número muito grande de agentes, a arquitetura de troca de mensagens pode não ser viável, devido ao tempo em que as mensagens levarão para serem transmitidas, inviabilizando todo o processo de comunicação do sistema.

Para resolver esse problema, surgiu a arquitetura federativa onde os agentes da sociedade são divididos em grupos ou federações segundo um critério de agrupamento escolhido. Junto a cada grupo de agentes encontram-se os agentes chamados de facilitadores, responsáveis por receber a mensagem que chega a cada grupo e encaminhá-la para o agente destinatário presente naquele grupo [36].

O grande objetivo da arquitetura federativa é fazer com que as mensagens trocadas entre os agentes sejam enviadas apenas aos seus respectivos destinatários, reduzindo assim o fluxo de mensagens desnecessárias existentes na arquitetura de troca de mensagens, o qual envia as mensagens para todos os agentes.

3.2.5 Coordenação entre agentes

A coordenação entre agentes refere-se à maneira em que os agentes estão organizados a fim de cooperar para alcançar um objetivo comum do sistema. Existem dois

mecanismos básicos para coordenar os agentes: o mecanismo mestre-escravo e o mecanismo de mercado.

Nas arquiteturas do tipo “mestre-escravo” existe duas classes de agentes: os gerentes (mestres) e os trabalhadores (escravos). Os agentes trabalhadores são coordenados por um gerente que distribui as tarefas entre estes e espera o resultado. O agente facilitador pode existir se os agentes estiverem divididos em grupos.

Nas arquiteturas baseadas no mecanismo de mercado, todos os agentes estão em um mesmo nível e sabem as tarefas que cada agente é capaz de desempenhar. Esta arquitetura visa diminuir a quantidade de mensagens trocadas, tendo em vista que cada agente já conhece todos os outros. Baseado nesse mecanismo surgiu a Programação Orientada ao Mercado (MOP), onde os agentes podem ser classificados como produtores e consumidores e são utilizados para maximizar lucros através de um processo de negociação [36].

3.2.6 Comunicação entre agentes

A FIPA (Foundation for Intelligent Physical Agents) direciona esforços para a produção de especificações para a tecnologia de agentes. Um conjunto de especificações FIPA muito importante é o da ACL (Agent Communication Language). Esse conjunto inclui as especificações de uma série de protocolos de interação entre os agentes (como o Contract-Net, Query, Request, entre outros). As especificações FIPA ACL incluem a definição do formato das mensagens trocadas entre os agentes [28][9].

A ACL foi definida para a comunicação entre agentes e pode ser dividida em [26]:

- Vocabulário: dicionário de conceitos do domínio de aplicação do sistema multiagente e seus relacionamentos. Também conhecido como ontologia.
- Linguagem interna: KIF (Knowledge Interchange Format), linguagem de programação em lógica de primeira ordem com capacidade de codificar dados simples, restrições, regras e expressões.
- Linguagem externa: KQML (Knowledge Query and Manipulation Language) que é uma camada lingüística capaz de encapsular estruturas KIF, fornecendo uma comunicação mais eficiente.

Assim, uma mensagem ACL nada mais é do que uma expressão KQML onde os argumentos são termos ou sentenças KIF formadas por termos da ontologia [36].

3.2.7 JADE

O Java Agent Development framework (JADE) é um ambiente open source para desenvolvimento de aplicações baseado em agentes, conforme as especificações da FIPA, o qual tem por objetivo garantir um padrão de integração de SMA através de um conjunto de agentes e de serviços de sistema [48].

JADE oferece facilidades para o desenvolvimento de sistemas multiagente que previnem o desenvolvedor de se preocupar com a implementação de uma plataforma eficiente de agentes, ou com a comunicação, troca de mensagens e muitos outros atributos que um sistema multiagente necessita.

Além disso, JADE oferece uma variedade de ferramentas de monitoração, gerenciamento e depuração que ajudam tanto no desenvolvimento quanto na manutenção e suporte a sistemas multiagente. Vale ressaltar a grande preocupação que JADE tem em sempre manter os padrões especificados pela FIPA. Fato que aumenta o grau de interoperabilidade e escalabilidade do ambiente em relação a outros SMA [9].

3.2.8 JADEX

JADEX é um framework implementado em Java que suporta a arquitetura BDI (Beliefs, Desire and Intentions) [80].

Os agentes possuem Beliefs que representam o seu modelo do ambiente, também conhecidos como crenças. Eles possuem Goals que são os estados objetivos e Intentions que é um script de como alcançar os objetivos. O JADEX pode ser utilizado integrado ao JADE ou com outro middleware.

3.2.9 JACK

Jack é uma plataforma de execução de agentes comercial implementado em JAVA. Assim como o JADEX, ele segue o padrão BDI [46].

Um agente JACK pode ter tanto comportamentos reativos (dirigidos a eventos) quanto comportamentos pró-ativos (dirigidos a objetivos). Cada agente JACK possui:

- Um conjunto de crenças sobre o mundo;
- Um conjunto de eventos que ele estará hábil a responder;
- Um conjunto de objetivos que ele desejará alcançar;

- Um conjunto de planos que dizem como ele deve alcançar seus objetivos ou responder a eventos. Um plano descreve a sequencia de ações que um agente deve tomar quando um evento ocorre. Cada plano corresponde a um evento. Ele descreve exatamente o que o agente deve fazer quando um dado evento acontecer.

Jack também é um framework de desenvolvimento de agentes. Ele possui uma linguagem própria para desenvolver os agentes chamada JACK Agent Language.

3.2.10 Chimera

Chimera é um ambiente de execução de agentes. Ele faz parte do software LPA Winprolog. Os agentes são desenvolvidos utilizando a linguagem Prolog [12].

Um agente Chimera é composto de dois componentes:

- Socket: refere-se à parte comum (comunicações) do agente, controlada por um Socket (predicados Winsock do WIN-Proloog), que suporta comunicações TCP/IP;
- Predicado: refere-se à parte events (eventos) do agente. Trata-se de um programa WIN-Prolog que é chamado automaticamente sempre que um evento ocorre.

3.2.11 PROLOG

A linguagem Prolog foi criada em 1972 por Colmerauer e Roussel, com o objetivo de trabalhar com programação em lógica, a qual se baseia no cálculo de predicados. Essa linguagem não possui um código para manipular a memória ou realizar desvios condicionais, porém, ela é a linguagem mais adequada para solucionar uma determinada categoria de problemas. Esta categoria diz respeito aos problemas onde é necessário representar algum tipo de conhecimento, por exemplo, em aplicações que realizem computação simbólica, na compreensão de linguagem natural ou em sistemas especialistas.

Um programa Prolog constitui-se de uma coleção de fatos (base de dados) e regras (relações lógicas), esses itens descrevem o domínio de um determinado problema. Esta descrição do problema é avaliada por um interpretador, o qual utilizando um “motor de inferência” realiza deduções em busca de conclusões válidas para consultas realizadas pelos usuários.

Embora inicialmente destinado a processamento de linguagem natural, a linguagem Prolog estendeu-se, desde então, a outras áreas como prova de teoremas, sistemas especialistas, jogos, sistemas automatizados de atendimento, ontologias e sofisticados sistemas de controle.

3.2.12 Trabalhos Correlatos

Vários sistemas de apoio a tomada de decisões no domínio da licitação pública são apresentados na literatura, como o Cnet [79], INFOLICIT [23] e SMA Pregoeiro [75]. A Tabela 1 apresenta os resultados de uma análise comparativa entre eles e o sistema SMPLP de acordo com os seguintes critérios: paradigma de desenvolvimento, linguagem e paradigma de modelagem, modalidades de licitação suportadas.

Em relação ao paradigma de desenvolvimento, bons sistemas de apoio ao processo de licitação pública devem preferivelmente utilizar o paradigma baseado em agentes, devido a algumas características, como constante monitoramento do ambiente, autonomia e sociabilidade.

Os agentes estão num ciclo constante de monitoramento do ambiente, percebendo o que ocorre e atuando quando necessário. Por exemplo, o agente “Interface” do sistema SMPLP fica monitorando se um novo pedido de compra foi solicitado.

Diferentemente dos objetos, os agentes são entidades computacionais autônomas, que podem agir sem a necessidade de intervenção do usuário. Por exemplo, no sistema SMPLP o agente ao perceber que um novo pedido de compra foi solicitado, envia automaticamente uma mensagem para o setor de compras elaborar o miniprojeto desse pedido.

Devido ao dinamismo e a complexidade do processo licitatório, são necessários diversos agentes interagindo entre si e com tarefas bem definidas para conduzir adequadamente o processo. Por exemplo, como descrito no modelo atos de comunicação (Figura 13), o agente “Analista” informa que existe um projeto para o agente “Licitação” realizar a cotação de preços, com o intuito de definir a modalidade e o tipo de licitação.

Dentro os sistemas analisados todos utilizam o paradigma baseado em agentes, comprovando assim, que este paradigma é apropriado para apoio a tomada de decisões no processo de licitação pública.

No referente à linguagem de modelagem, o processo de modelagem é imprescindível para o desenvolvimento de qualquer sistema, pois permite ao desenvolvedor

uma ampla visão do seu funcionamento, proporcionando assim, uma forma mais efetiva de se desenvolvê-lo. Além disso, a linguagem adotada para a modelagem deve ser adequada ao paradigma de desenvolvimento utilizado.

Dentre os sistemas baseados em agentes estudados, apenas o Cnet foi modelado utilizando a UML, sendo esta linguagem inapropriada para a modelagem de sistemas multiagente, por possuir um vocabulário voltado para o desenvolvimento orientado a objetos. O sistema SMA Pregoeiro utiliza a linguagem AUML, a qual é uma adaptação da UML para o paradigma baseado em agentes, herdando assim, muitas características da orientação a objetos. Já os sistemas INFOLICIT e SMPLP utilizam a MADAE-ML, uma linguagem de modelagem para a engenharia de domínio e aplicação multiagente, que é a mais apropriada por possuir um vocabulário adequado e específico para o desenvolvimento de sistemas multiagente.

Em relação às modalidades de licitação suportadas, a complexidade do processo de licitação pública é proveniente das diversas fases, tipos, modalidades e atores envolvidos. Sistemas de apoio às decisões que deem suporte a esses elementos, como tipo e modalidade de licitação são necessários. O sistema SMA Pregoeiro dá suporte a modalidade pregão selecionando os concorrentes para o leilão. Já o sistema Cnet aborda a modalidade convite, através do envio do edital de licitação para fornecedores cadastrados. O sistema INFOLICIT não aborda nenhuma modalidade de licitação, reservando-se a filtrar e enviar os editais que são de interesse para os fornecedores cadastrados no sistema. Dentro os sistemas analisados, o SMPLP é o mais completo por orientar a comissão permanente a selecionar as modalidades e os tipos de licitação mais adequados para uma eventual licitação.

Cr�terios de compara�o	Cnet	INFOLICIT	SMA Pregoeiro	SMPLP
Linguagem e processo de Modelagem	UML	MADAE-ML	AUML	MADAE-ML
Paradigma de desenvolvimento	Agentes	Agentes	Agentes	Agentes
Modalidades de Licita�o	Convite	N�o aborda	Preg�o	Convite Tomada de pre�o Concorr�ncia

Tabela 1 – An lise Comparativa de alguns Sistemas de Apoio   Licita o P blica

3.3 Considerações Finais

Este capítulo apresentou conceitos importantes da área de engenharia de software e do conhecimento, iniciando com Sistemas Multiagente, até as ferramentas que serão utilizadas no desenvolvimento do SMPLP.

Conforme apresentado, um agente é tudo o que pode ser considerado capaz de perceber seu ambiente por meio de sensores e de agir sobre esse ambiente por intermédio de atuadores. Sua principal característica é a autonomia, devido à capacidade dele atuar sem a intervenção humana ou de outros sistemas através do controle que ele possui do seu estado interno e do seu comportamento. Para o SMPLP, os agentes são do tipo deliberativo por serem agentes capazes de realizarem um planejamento de suas ações e de utilizarem um mecanismo de inferência.

Foi apresentado que o Sistema Multiagente (SMA) é um sistema composto por vários agentes que se comunicam e são coletivamente capazes de atingir objetivos que não seriam capazes de satisfazer separadamente. Como o SMPLP possui três agentes, a abordagem dos SMA é adequada para a modelagem de sistemas, porque os agentes serão capazes de reagir a eventos não previstos, explorando as oportunidades que surgem e realizarão acordos dinamicamente com outros agentes cuja presença pôde não ter sido prevista no projeto inicial, proporcionando maior efetividade no processo de tomada de decisão.

Foi apresentado que o JADE é um ambiente open source para desenvolvimento de aplicações baseado em agentes, conforme as especificações da FIPA, o qual oferece facilidades para o desenvolvimento de sistemas multiagente que previnem o desenvolvedor de se preocupar com a implementação de uma plataforma eficiente de agentes, ou com a comunicação, troca de mensagens e muitos outros atributos que um sistema multiagente necessita. Logo, será a ferramenta utilizada para desenvolver os agentes do SMPLP.

Quanto ao motor de inferência usado pelos agentes no sistema SMPLP, foi apresentado o Prolog, um programa que se constitui de uma coleção de fatos (base de dados) e regras (relações lógicas), os quais descrevem o domínio de um determinado problema.

4 MADAE-PRO – UM PROCESSO BASEADO NO CONHECIMENTO PARA A ENGENHARIA DE DOMÍNIO E APLICAÇÕES MULTIAGENTE

O MADAE-Pro [60] é um processo para o desenvolvimento e reuso de famílias de sistemas multiagente. Ele integra dois subprocessos complementares. Um é baseado nos conceitos da Engenharia de Domínio, isto é, visa construir artefatos reutilizáveis que representem uma família de aplicações e o outro, baseado na Engenharia de Aplicações, guia o desenvolvimento de uma aplicação específica reutilizando os produtos do primeiro subprocesso.

Outra característica do MADAE-Pro é ser um processo baseado no conhecimento. Isto é, a definição das suas fases, tarefas e produtos são classes da ONTORMAS e os relacionamentos entre esses conceitos são especificados através dos “slots” dessa ontologia. Durante o desenvolvimento das famílias de sistemas multiagente as classes da ONTORMAS são instanciadas formando uma base de conhecimento, o que faz da ONTORMAS também um repositório para os modelos desenvolvidos com o MADAE-Pro.

O MADAE-Pro envolve o uso de metodologias que especificam “como” realizar as tarefas de desenvolvimento, de um ciclo de vida que define a “ordem” de realização dessas atividades de desenvolvimento, de papéis do processo que representa “quem” é responsável por realizar cada tarefa e de tecnologias (ferramentas e linguagens) que apóiam os papéis do processo na realização de suas atividades.

Na rede semântica ilustrada na Figura 2 estão os principais elementos envolvidos no MADAE-Pro. O processo adota as metodologias MADEM e MAAEM. A MADEM guia o desenvolvimento de artefatos de software reusáveis integrantes de uma família de aplicações multiagente enquanto a MAAEM, o desenvolvimento de uma aplicação de software multiagente particular que pode ocorrer sem reuso ou reutilizando artefatos de software desenvolvidos a partir da metodologia MADEM. Uma técnica está associada com cada fase de desenvolvimento para guiar as tarefas de desenvolvimento das metodologias. As técnicas GRAMO, DDEMAS e DIMAS integram a metodologia MADEM, e estão associadas, respectivamente, as suas fases de Análise de Domínio, Projeto de Domínio e Implementação de Domínio. As técnicas SRAMO, ADEMAS e AIMAS integram a metodologia MAAEM e estão associadas, respectivamente, as suas fases de Engenharia dos Requisitos da Aplicação, Projeto da Aplicação e Implementação da Aplicação [59][60][61].

O ciclo de vida adotado no MADAE-Pro é o iterativo e incremental, devido a complexidade inerente ao desenvolvimento de sistemas multiagente. O processo é suportado

pela ferramenta ONTORMAS, que é utilizada para guiar as tarefas de desenvolvimento, realizar a modelagem visual, documentar e armazenar os artefatos produzidos durante a execução do processo. A linguagem para modelagem de sistemas multiagente adotada é a MADAE-ML. Esta linguagem estabelece uma representação gráfica para os modelos e conceitos de modelagem das metodologias MADAEEM e MAAEM. Por fim, o processo define os papéis do processo (ex. Programador, Analista de Sistemas), encarregados da realização de uma ou mais tarefas durante a execução do processo.

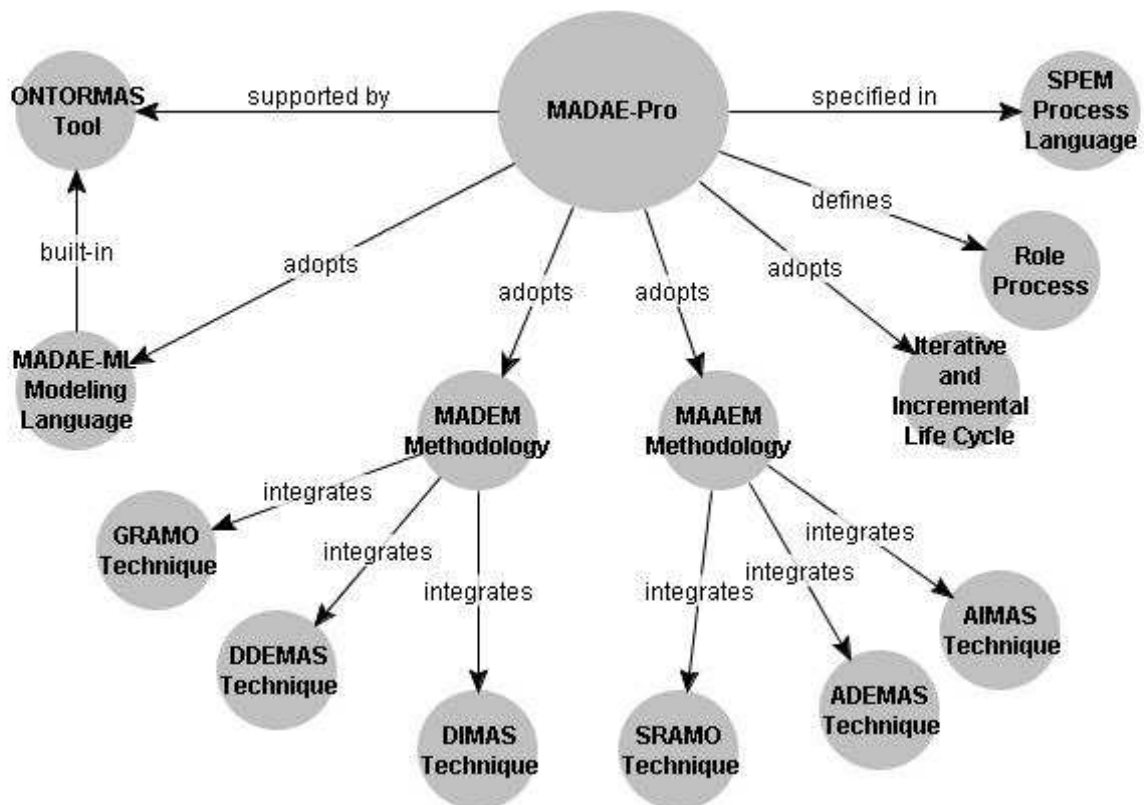


Figura 2 - Elementos envolvidos no MADAE-Pro

Quanto ao reuso o processo MADAE-Pro se enquadra na seguinte classificação [81]:

- Quanto ao conteúdo: componentes, que são, no nível de especificação, os modelos desenvolvidos com instâncias da ONTORMAS, descrita na seção 3.3.2. No nível de implementação, os componentes são agentes de software.
- Quanto ao escopo: vertical, pois o reuso é realizado em um mesmo domínio de aplicação.
- Quanto ao modo: sistemático, utilizando os processos da Engenharia de Domínio e de Aplicações.
- Quanto à técnica: composicional.

- Quanto à intenção: caixa-branca (“white-box”), o reuso tanto no nível de especificação quanto no nível de implementação permite a realização de adaptações.
- Quanto ao Produto: modelos de Domínio, Frameworks, Agentes.

O ciclo de vida do processo MADAE-Pro é composto por três fases de desenvolvimento, Análise da Aplicação, Projeto da Aplicação e Implementação da Aplicação, as quais se referem à Engenharia de Aplicação Multiagente. Esse ciclo é iterativo, incremental e dirigido por objetivos. Seu desenvolvimento sempre é realizado a partir de sucessivos incrementos, objetivando reduzir sua complexidade.

4.1 A Fase de Engenharia dos Requisitos da Aplicação

No subciclo de vida da fase de Engenharia dos Requisitos da Aplicação os requisitos de uma aplicação particular são identificados. Para isso, são reusados modelos desenvolvidos na Engenharia de Domínio, através da modelagem de conceitos, objetivos, papéis, interação entre papéis e prototipação da interface com o usuário.

O subciclo da fase de Engenharia dos Requisitos da Aplicação, inicia-se com a Modelagem de Conceitos, na qual é realizada uma “tormenta de ideias” sobre os conceitos do domínio e estes são representados em uma rede semântica. A seguir é feita a Modelagem de Objetivos. Nessa tarefa, primeiro buscam-se Modelos de Objetivos preexistentes na ONTORMAS, a fim de tornar essa tarefa mais produtiva. Se não houver modelos disponíveis para reuso, então se definem os objetivos geral e específicos da aplicação multiagente e as responsabilidades necessárias para alcançá-los através de um documento de requisitos da aplicação. Ao final desta tarefa o Modelo de Objetivos será produzido[60][61].

O próximo passo é reusar Modelos de Papéis preexistentes que estejam associados aos objetivos específicos presentes no Modelo de Objetivos, através das responsabilidades que compartilham. Caso não haja Modelos de Papéis disponíveis para reuso, então se realiza a Modelagem de Papéis, a partir de responsabilidades identificadas no Modelo de Objetivos que são associadas a papéis (cada responsabilidade é associada a um papel), é definido ainda o conhecimento requerido para realização dessa responsabilidade, as entidades externas com as quais é necessário interagir e as suas pré e pós condições.

Na tarefa seguinte, é realizada a modelagem de interações entre papéis. Nessa tarefa são reusados Modelos de Interações entre Papéis desenvolvidos no processo da Engenharia de Domínio que estejam associados as mesmas responsabilidades definidas no

Modelo de Objetivos ou, no caso de não haver modelos disponíveis para reuso, esse modelo é construído identificando-se as necessidades de comunicação dos papéis definidos no Modelo de Papéis para alcançar os objetivos específicos da aplicação.

Por último, é realizada a prototipação da interface com o usuário, onde é possível reusar protótipos de interface genéricos e especializar para a aplicação específica ou defini-la a partir do Modelo de Interações entre Papéis.

4.2 A Fase de Projeto da Aplicação

Na fase de Projeto da Aplicação, o desenvolvedor poderá reutilizar soluções de projeto relacionadas a uma família de aplicações e adaptá-la aos requisitos específicos da aplicação em desenvolvimento. Esta fase consiste em realizar o Projeto Arquitetural da Sociedade Multiagente, incluindo seus mecanismos de coordenação e cooperação e o Projeto do Agente, onde é feito o projeto interno de cada agente da sociedade, modelando sua estrutura de conhecimento e seus comportamentos.

O subciclo da fase de Projeto da Aplicação inicia-se com a realização da subtarefa de modelagem do conhecimento da sociedade multiagente, onde o conhecimento compartilhado pelos agentes da sociedade é representado em uma rede semântica, gerando o Modelo de Conhecimento da Sociedade Multiagente. A outra possibilidade é o reuso de Modelos do Conhecimento da Sociedade Multiagente desenvolvido no subprocesso da Engenharia de Domínio [60].

O próximo passo é a modelagem da sociedade multiagente. Aqui são definidos os agentes da sociedade a partir dos papéis definidos no Modelo de Papéis da fase de Análise da Aplicação ou se já houver Modelos da Sociedade Multiagente desenvolvidos no subprocesso da Engenharia de Domínio da mesma família esse é reusado. A seguir, a modelagem de interações entre agentes é realizada, onde as interações dos agentes da sociedade são definidas. Aqui também podem ser reusados modelos preexistentes.

A próxima tarefa é a modelagem dos mecanismos de cooperação e coordenação, onde é definido como os agentes irão cooperar e estar organizados na sociedade. Para isso, usam-se padrões já popularizados na comunidade, como o padrão de mercado, mestre-escravo, camadas, etc. Modelos pré-existentes também podem ser reusados nesse passo. Após a construção do Modelo dos Mecanismos de Cooperação e Coordenação finaliza-se a tarefa de projeto arquitetural e dá-se início a tarefa de projeto do agente.

No projeto do agente a modelagem de conhecimento é realizada, onde é definido o conhecimento particular de cada agente da sociedade. A seguir, a modelagem das ações do agente é realizada, onde são definidas as ações que o agente terá para realizar suas responsabilidades. Aqui também podem ser reusados Modelos de Conhecimento e Modelos de Ações desenvolvidos no subprocesso da Engenharia de Domínio.

4.3 A Fase de Implementação da Aplicação

No subciclo da fase de Implementação, são construídos o Modelo de Comportamentos e o Modelo de Atos de Comunicação. Com base nesses dois modelos é realizada a codificação da aplicação multiagente. No Modelo de Comportamentos são especificados os comportamentos que os agentes terão e no Modelo de Comunicação as interações que eles precisam realizar para cumprir suas responsabilidades. Esses modelos são construídos de forma semelhante aos da fase de Implementação de Domínio e o papel do processo é o mesmo, a diferença é que aqui é uma aplicação específica é implementada a partir do reuso de agentes de software genéricos pertencentes a uma família de aplicações multiagente [60].

4.4 A Metodologia MAAEM

A MAAEM é uma metodologia para a análise, o projeto e a implementação de aplicações multiagente através da reutilização de artefatos de software anteriormente produzidos na Engenharia de Domínio Multiagente. Sendo inspirada no desenvolvimento baseado em componentes, ela envolve a seleção, adaptação e composição desses artefatos para a construção de uma aplicação. Na fase de análise dos requisitos, pretende-se identificar e especificar os requisitos de uma determinada aplicação, partindo de modelos de domínio, elaborados na fase correspondente da Engenharia de Domínio Multiagente.

Desse modo, a tarefa central do desenvolvedor é reusar um conjunto de requisitos da família em um domínio. Tais requisitos específicos da aplicação, quando provenientes da família, são levantados a partir da seleção dentre os requisitos comuns ou variáveis no domínio. Um resumo das tarefas realizadas, dos insumos requeridos e dos produtos obtidos em cada fase de desenvolvimento de uma aplicação multiagente seguindo as diretrizes da metodologia MAAEM está descrito na Tabela 2. Nessa tabela, têm-se além dos insumos presentes na Tabela , os produtos desenvolvidos pela metodologia MADEM [61].

Fases	Insumos	Tarefas	Produtos		
Engenharia dos Requisitos da Aplicação	Modelo de Conceitos (MADEM)	Modelagem de Conceitos	Modelo de Conceitos		Especificação dos Requisitos da Aplicação
	Conhecimento de Especialistas, Literatura no Domínio, Aplicações de Software Existentes				
	Modelo de Objetivos (MADEM)	Modelagem de Objetivos	Modelo de Objetivos		
	Requisitos da família de aplicações				
	Modelo de Papéis (MADEM)	Modelagem de Papéis	Modelo de Papéis		
	Modelo de Objetivos				
	Modelo de Interações entre Papéis (MADEM)	Modelagem de Interações entre Papéis	Modelo de Interações entre Papéis		
	Modelo de Objetivos				
	Modelo de Papéis				
	Protótipo da Interface com o usuário (MADEM)	Prototipagem da Interface com o usuário	Protótipo da Interface com o usuário		
Modelo de Papéis					
Modelo de Interações entre Papéis					
Projeto da Aplicação	Modelo do Conhecimento da Sociedade Multiagente (MADEM)	Modelagem Arquitetural	Modelagem do Conhecimento da Sociedade Multiagente	Modelo do Conhecimento da Sociedade Multiagente	Arquitetura da Aplicação
	Modelo de Conceitos		Modelagem da Sociedade de Agentes	Modelo da Sociedade Multiagente	
	Modelo de Interações entre Papéis		Modelagem das Interações entre Agentes	Modelo de Interações entre Agentes	
	Modelo da Sociedade Multiagente (MADEM)		Modelagem dos Mecanismos de Cooperação e Coordenação	Modelo dos Mecanismos de Cooperação e Coordenação	
	Modelo de Papéis	Projeto do Agente	Modelagem do Conhecimento do Agente	Modelos do Conhecimento do Agente	
	Modelo de Interações entre Agentes (MADEM)		Modelagem das Ações dos Agentes	Modelos das Ações do Agente	
	Modelo da Sociedade Multiagente				
	Modelo dos Mecanismos de Cooperação e Coordenação (MADEM)				
	Modelo de Objetivos (Requisitos Não-Funcionais)				
	Modelo da Sociedade Multiagente				
	Modelo das Interações entre Agentes				
	Modelos do Conhecimento do Agente (MADEM)				
	Modelo de Interações entre Papéis				
	Modelo do Conhecimento da Sociedade Multiagente				
Modelos das Ações do Agente (MADEM)					
Modelo da Sociedade Multiagente					
Modelo de Interações entre Agentes					
Implementação da Aplicação	Modelo de Comportamentos (MADEM)	Modelagem de Comportamentos	Modelo de Comportamentos		Modelo de Implementação da Aplicação
	Modelos de Ações				
	Modelo de Atos de Comunicação (MADEM)	Modelagem de Atos de Comunicação	Modelo de Atos de Comunicação		
	Modelo de Interações entre Agentes				
	Agentes de Software Executáveis (MADEM)	Implementação dos Agentes	Agentes de Software Executáveis		
	Modelo de Comportamentos				
Modelo de Atos de Comunicação					

Tabela 2 - Fases, Insumos, Tarefas e Produtos da metodologia MAAEM [61]

4.4.1 O Modelo de conceitos

O Modelo de Conceitos tem por objetivo reunir o máximo de informações relevantes ao domínio. Nele são representados os conceitos do domínio e suas relações em uma rede semântica, onde os conceitos representam os nodos e suas relações os arcos. Este modelo é construído com base em possíveis Modelos de Conceitos preexistentes, no conhecimento de especialistas no domínio, em aplicações já desenvolvidas e na literatura disponível no domínio. Na Figura 3, é ilustrado graficamente o metamodelo de conceitos, onde estão definidos os conceitos e relacionamentos disponíveis [60].

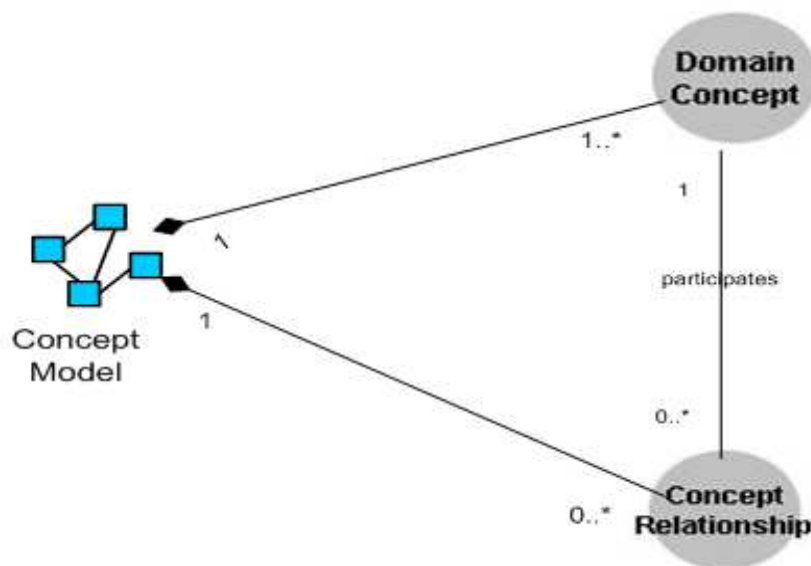


Figura 3 - Metamodelo de Conceitos [61]

4.4.2 O Modelo de objetivos

Na tarefa de modelagem de objetivos são identificados os objetivos comuns e variáveis da família de aplicações multiagente, as entidades externas, e as responsabilidades necessárias para alcançá-los. Ele é construído com base em possíveis modelos de objetivos preexistentes e a partir dos requisitos de uma família de aplicações multiagente. Nesta tarefa, a modelagem de variabilidade é realizada identificando-se pontos de variação em objetivos específicos relacionados com grupos de responsabilidades variantes.

A Figura 4 ilustra o metamodelo de objetivos, com os conceitos e os relacionamentos semânticos existentes entre esses conceitos. O conceito “General Goal”, representa o objetivo geral da família de aplicações multiagente; “Specific Goal”, representa um objetivo específico associado ao objetivo geral; “Responsibility”, representa uma

responsabilidade necessária para alcançar um ou mais objetivos específicos; “External Entity”, representa uma entidade externa com a qual é necessário interagir para alcançar objetivos específicos [60].

O conceito “General Goal” está relacionado a dois ou mais conceitos “Specific Goal” através do relacionamento semântico “leads to”, isso significa que dois ou mais objetivos específicos conduzem ao objetivo geral. O conceito “Responsibility” está associado a “Specific Goal” pelo relacionamento semântico “achieves”, isso denota que um objetivo específico será alcançado através da execução de uma ou mais responsabilidades. O conceito “External Entity” está associado a “Specific Goal”, indicando que para alcance de um objetivo específico pode haver necessidade de interação com entidades externas. O conceito “Specific Goal” está associado a ele mesmo através do relacionamento “leads to” para indicar os casos onde existe uma hierarquia de objetivos específicos.

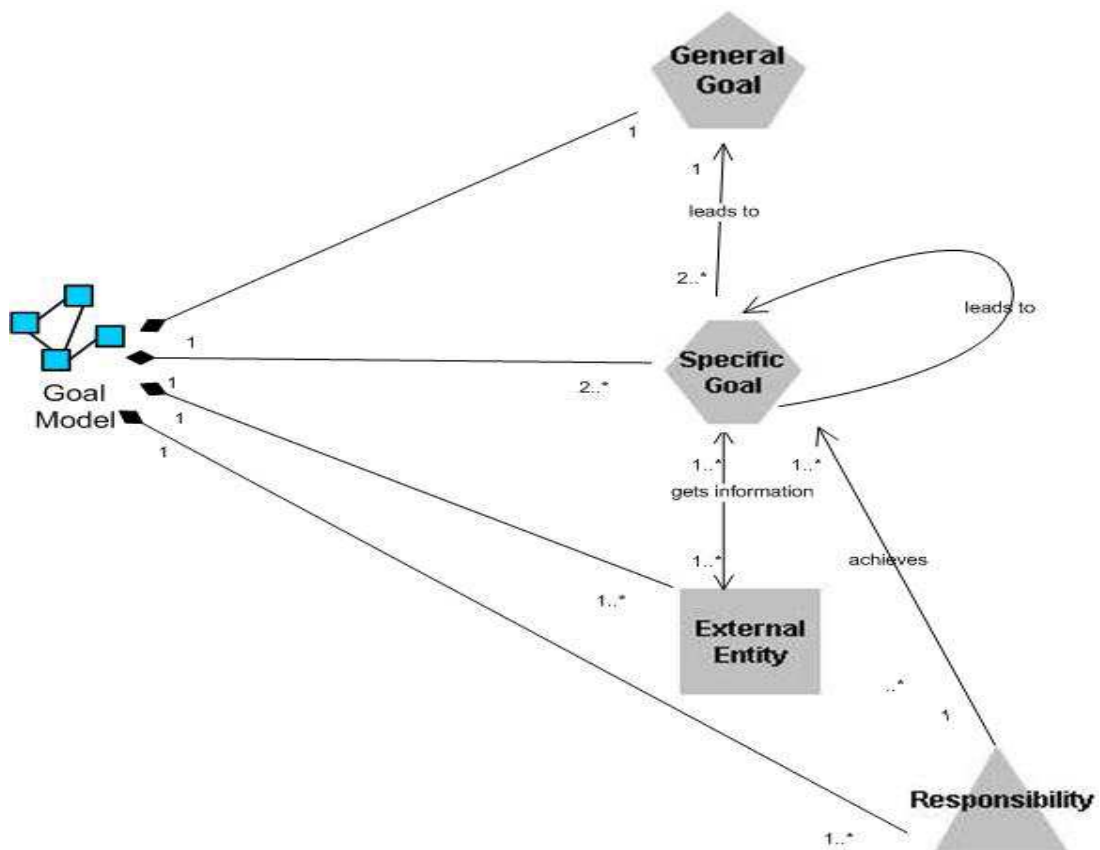


Figura 4 – Metamodelo de Objetivos [61]

4.4.3 O Modelo de papéis

A tarefa de modelagem de papéis associa as responsabilidades, comuns ou variantes, identificadas na tarefa de modelagem de papéis aos papéis que serão encarregados

delas. As pré e pós condições que devem ser satisfeitas antes e depois da execução de uma responsabilidade é identificada. O conhecimento requerido a partir de outras entidades (papéis ou entidades externas) para a realização das responsabilidades e o conhecimento produzido a partir da sua execução é identificado. Esta tarefa produz um conjunto de modelos de papéis, um para cada objetivo específico ou, tendo um ou mais pontos de variação, um modelo de papéis para cada variante, especificando papéis, responsabilidades, pré e pós condições, conhecimento e os relacionamentos entre estes conceitos.

Os conceitos definidos nesse metamodelo são: “Role”, “Responsibility”, “Knowledge”, “External Entity” e “Condition”. O conceito “Role” possui o relacionamento “in charge of” com o conceito “Responsibility”, significando que um papel é encarregado pela realização de uma responsabilidade. O conceito “Responsibility” possui os relacionamentos semânticos “used by” e “produces” com o conceito “Knowledge”, isso significa que para a/depois da realização de uma responsabilidade conhecimento pode ser requerido/produzido. O conceito “External Entity” tem o relacionamento semântico “satisfied by” e “has knowledge” com os conceitos “Condition” e “Responsibility”, respectivamente, isso significa que uma entidade externa é satisfeita através de uma condição e tem conhecimento que pode ser utilizado por uma responsabilidade. A Figura 5 ilustra o Modelo de Papéis do ONTOSERS [60].

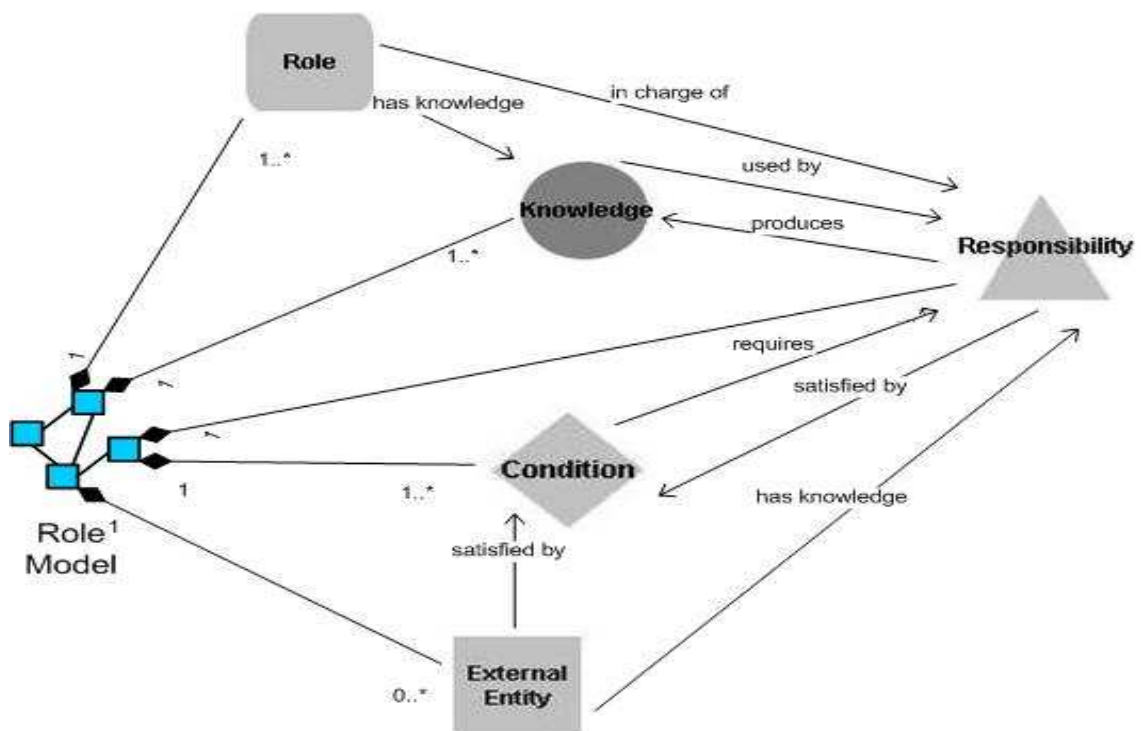


Figura 5 – Metamodelo de Papéis [61]

4.4.4 O Modelo de interações entre papéis

O Modelo de Interações entre Papéis representa a cooperação das entidades externas (usuários, outros sistemas, etc) e internas (papéis) para alcançar um objetivo específico. Esse modelo é construído reusando Modelos de Interações entre Papéis preexistentes e com base no Modelo de Objetivos e no Modelo de Papéis. O Modelo de Objetivos na associação entre um objetivo específico para cada Modelo de Interações entre Papéis e o Modelo de Papéis para associar papéis responsáveis do alcance desses objetivos específicos.

Na Figura 6 o metamodelo de papéis é ilustrado. Nesse metamodelo, estão disponíveis os conceitos de “Role”, “External Entity” e “Role Interaction”. O conceito de “Role Interaction” representa as interações entre dois conceitos “Role” ou entre um conceito “Role” e um conceito de “External Entity”, através do relacionamento semântico “participates” [60].

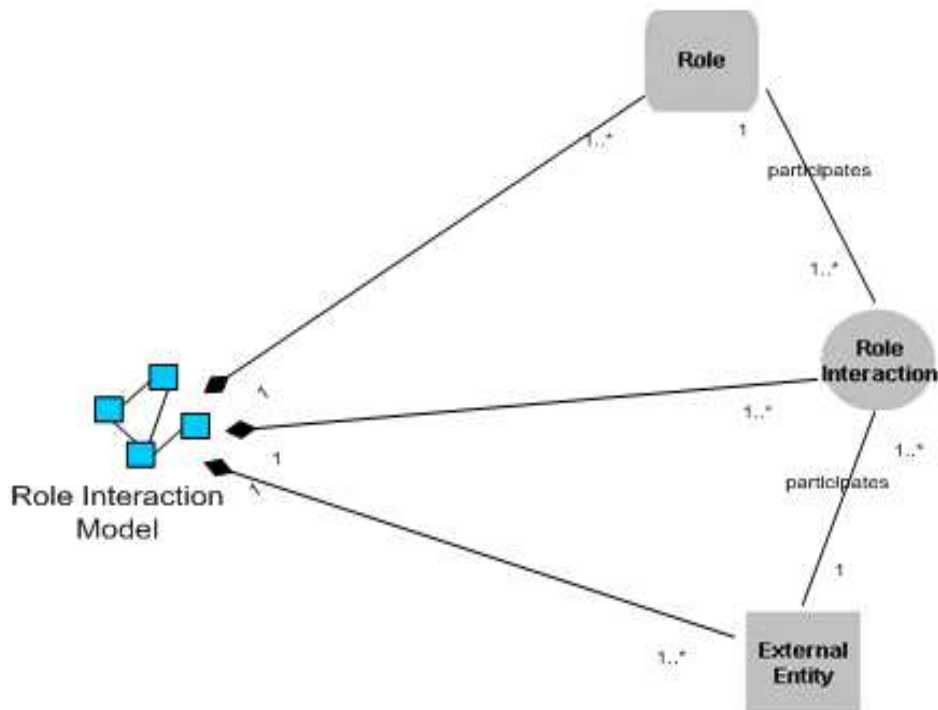


Figura 6 – Metamodelo de Interações entre Papéis [61]

4.4.5 O Protótipo de interface com o usuário

O Protótipo da Interface com o Usuário representa as interações dos usuários com o sistema. Esse protótipo é construído a partir do reuso de protótipos de interface com os usuários preexistentes e com base no Modelo de Papéis e no Modelo de Interações entre Papéis.

4.4.6 O Modelo de conhecimento da sociedade multiagente

O Modelo do Conhecimento da Sociedade Multiagente representa os conceitos que os agentes da sociedade precisam entender para comunicar entre si. Este modelo é construído a partir do reuso de Modelos do Conhecimento da Sociedade Multiagente desenvolvidos na Engenharia de Aplicações e com base no Modelo de Interações entre Papéis. O conhecimento trocados pelos papéis durante as suas interações são integrados e refinados neste modelo [60].

Na Figura 7 o metamodelo do conhecimento da sociedade multiagente é ilustrado. Nesse metamodelo, estão disponíveis os conceitos de “Knowledge”, que representa o conhecimento do domínio e de “Knowledge Relationship”, que representa os possíveis relacionamentos entre conceitos de conhecimento. “Part of”, “kind of” e “is a” são exemplos de “Knowledge Relationship”.

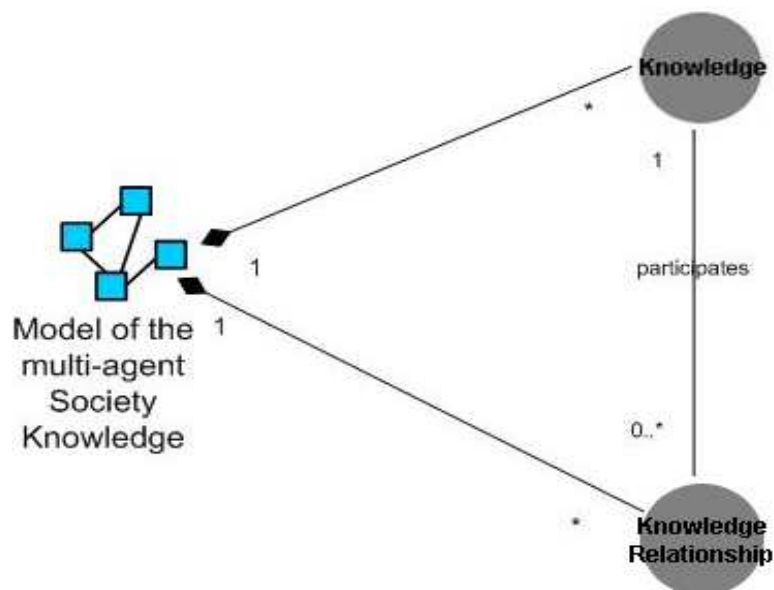


Figura 7 – Metamodelo do Conhecimento da Sociedade Multiagente [61]

4.4.7 O Modelo da sociedade multiagente

O Modelo da Sociedade Multiagente representa os agentes da sociedade, juntamente com o conhecimento, as destrezas e as pré e pós-condições necessárias para realização das suas responsabilidades. Esse modelo é construído a partir do reuso de Modelos da Sociedade Multiagente preexistentes e com base no Modelo de Papéis da fase de Análise de Domínio. Cada papel presente no Modelo de Papéis é mapeado para agentes. Pré e pós-condições, responsabilidades, conhecimento e entidades externas também são obtidas a partir do Modelo de Papéis. Um agente pode representar um ou mais papéis de acordo com a afinidade entre suas responsabilidades, número de interações entre eles ou critério de coesão funcional [60].

Na Figura 8 o metamodelo da sociedade multiagente é ilustrado. Nesse metamodelo, os conceitos representados são: “Agent”, representando os agentes da sociedade; “Responsibility”, representando as responsabilidades dos agentes; “Knowledge”, representando o conhecimento que o agente usa e produz durante a realização de suas responsabilidades; “Condition”, representando as pré e pós-condições que devem ser satisfeitas para/depois da realização de uma responsabilidade; “Skill”, que representa uma habilidade que o agente deve possuir para realizar uma determinada responsabilidade; “Activity”, representando as atividades que compõem uma responsabilidade; “External Entity”, representando as entidades externas com as quais os agentes interagem para realizar suas responsabilidades e os seus relacionamentos: “in charge of”, que indica que um agente será encarregado uma responsabilidade; “requires”; utilizado quando para realização de uma responsabilidade é necessário atender uma pré ou pós-condição; “required by”, representando que uma destreza é necessária para realização de uma determinada responsabilidade; “has knowledge”, que indica o conhecimento fornecido por entidades externas; “exercised through”, representando a realização de uma responsabilidade através de um conjunto de atividades; “used by” e “produces”, representando o conhecimento requerido e produzido através da realização de uma atividade.

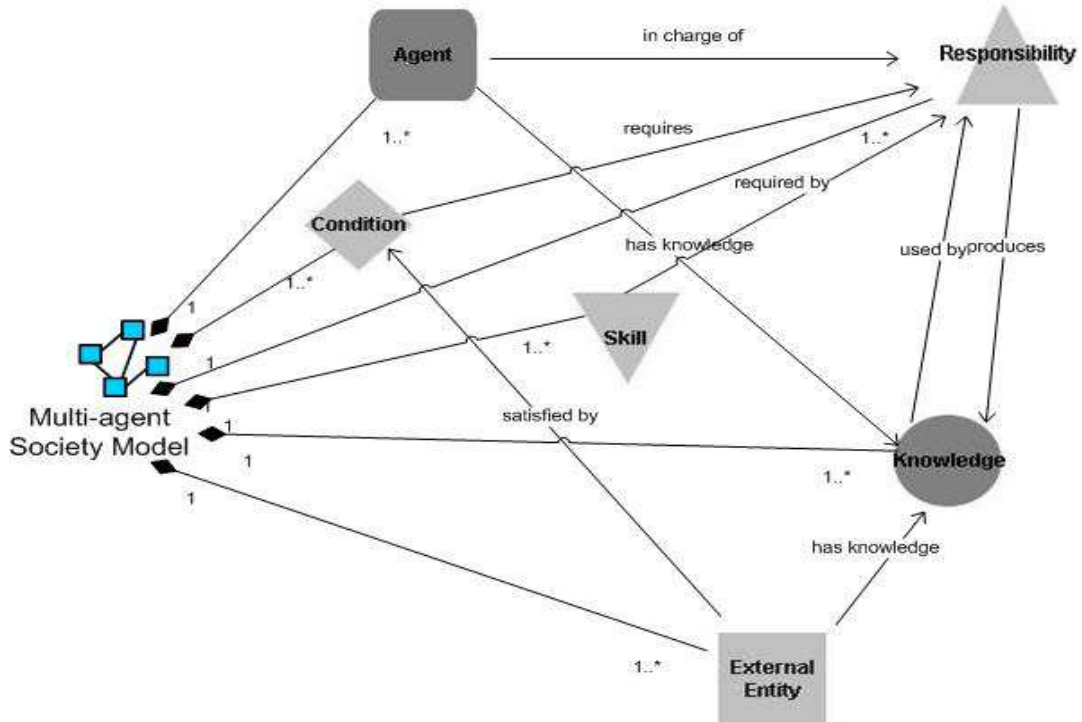


Figura 8 – Metamodelo da Sociedade Multiagente [61]

4.4.8 O Modelo de interações entre agentes

O Modelo de Interações entre Agentes representa as interações entre os agentes da sociedade. Este modelo é semelhante a um diagrama de sequência da UML. Fazendo uma analogia entre estes dois modelos, um agente ou entidade externa no Modelo de Interações entre Agentes seria um objeto no diagrama de sequência da UML, possuindo linha de vida e trocando mensagens entre si. As mensagens são especificadas seguindo as diretrizes da FIPA-ACL[28]. Este modelo é construído a partir do reuso de Modelo de Interações entre Agentes preexistentes e com base no Modelo da Sociedade Multiagente. A partir do Modelo da Sociedade Multiagente são obtidos os agentes, as entidades externas com que eles interagem e o conhecimento trocado em suas comunicações.

Na Figura 9, o metamodelo de interações entre agentes é ilustrado. Nesse metamodelo são representados os conceitos de “Agent”, representando cada agente da sociedade, “Knowledge”, representando o conhecimento trocado nas comunicações entre agentes e “External Entity”, representando as entidades externas com as quais os agentes têm necessidade de se comunicar. O conceito “Agent Interaction” representa as interações entre agentes e “Agent Notification from External Entity” representa as interações entre um agente e uma entidade externa.

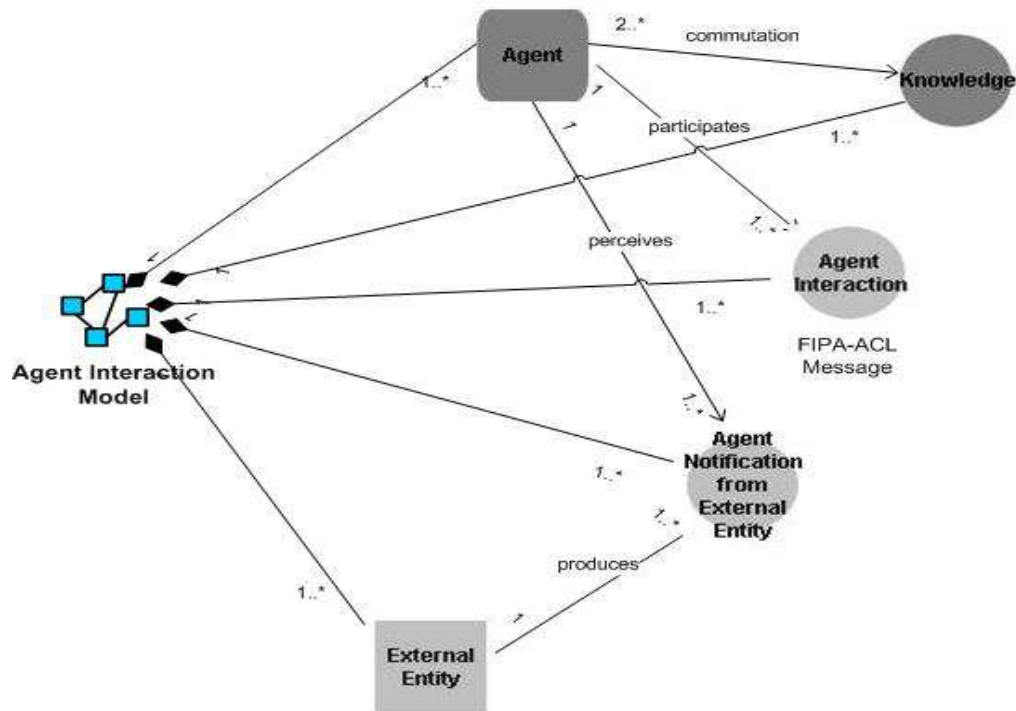


Figura 9 – Metamodelo de Interações entre Agentes [61]

4.4.9 O Modelo de cooperação e coordenação

Para o Modelo de Cooperação e Coordenação ser construído é reusado ou criado um mecanismo de cooperação e coordenação entre agentes que satisfaçam os requisitos da família de aplicações presentes no Modelo de Objetivos.

Esse modelo especifica como os agentes da sociedade vão cooperar, organizar e interagir para realizar uma determinada tarefa. O padrão de cooperação determina como os agentes vão interagir para realizar suas responsabilidades. Exemplos de padrões de cooperação são o quadro-negro e o federativo [36]. O padrão de coordenação especifica como os agentes estarão organizados. Eles podem estar em um mesmo nível hierárquico como é o caso do padrão de mercado, onde os agentes são classificados em consumidores e vendedores ou em níveis de hierarquia diferentes como, por exemplo, o mecanismo de mestre-escravo, onde os agentes são classificados em gerentes (mestres) e em trabalhadores (escravos). Os agentes trabalhadores são coordenados por um gerente que distribui as tarefas entre estes e espera o resultado. Esses padrões estão incluídos na ONTORMAS, como instâncias da classe “Architectural Pattern”. O Modelo de Cooperação e Coordenação é construído a partir do reuso de Modelos de Cooperação e Coordenação desenvolvidos preexistentes e com base no Modelo de Sociedade Multiagente, Modelo de Interações entre Agentes e no Modelo de

Objetivos. A partir do Modelo da Sociedade Multiagente são identificados os agentes da Sociedade [60].

No Modelo de Interações entre Agentes, o conjunto de interações entre agentes é identificado, auxiliando na escolha da melhor arquitetura de cooperação/coordenação. A partir do Modelo de Objetivos serão obtidos os requisitos não-funcionais que devem ser atendidos na escolha da arquitetura de cooperação/coordenação.

Na Figura 10, o metamodelo de cooperação e coordenação é ilustrado. Nesse metamodelo o conceito de “Architectural Pattern” representando o padrão arquitetural que os agentes estarão organizados e o conceito de “External Entity”, que representa as entidades externas com as quais os agentes interagem.

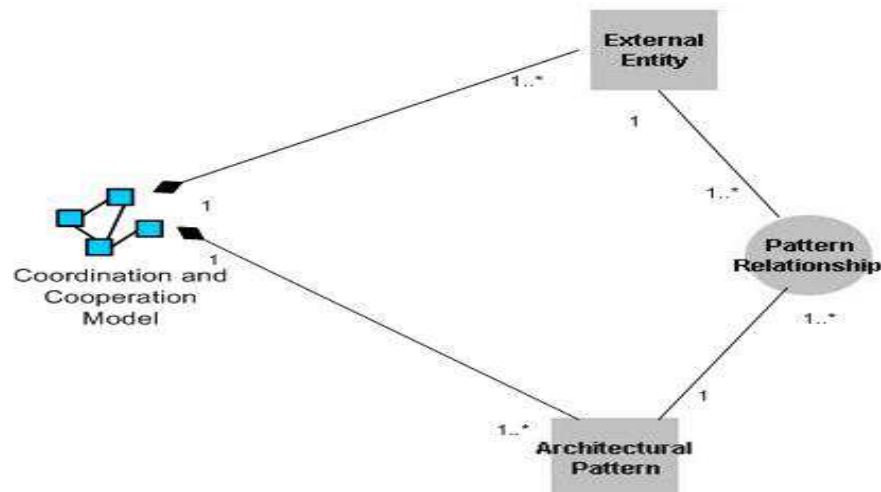


Figura 10 – Metamodelo de Coordenação e Cooperação [61]

4.4.10 O Modelo de conhecimento do agente

O Modelo de Conhecimento do Agente representa o conhecimento particular de cada agente da sociedade. Ele é construído a partir do reuso de Modelos de Conhecimento do Agente desenvolvidos na Engenharia de Aplicações e com base no Modelo de Interações entre Papéis, o qual provê o conhecimento compartilhado entre os papéis durante as suas interações, e no Modelo do Conhecimento da Sociedade Multiagente, do qual se obtém os conhecimentos específicos de cada agente. Na Figura 11, o metamodelo do conhecimento do agente é ilustrado. Esse metamodelo possui os conceitos de “Knowledge” e “Knowledge Relationship” que representam, respectivamente, o conhecimento particular de cada agente e o relacionamento entre esses conhecimentos.

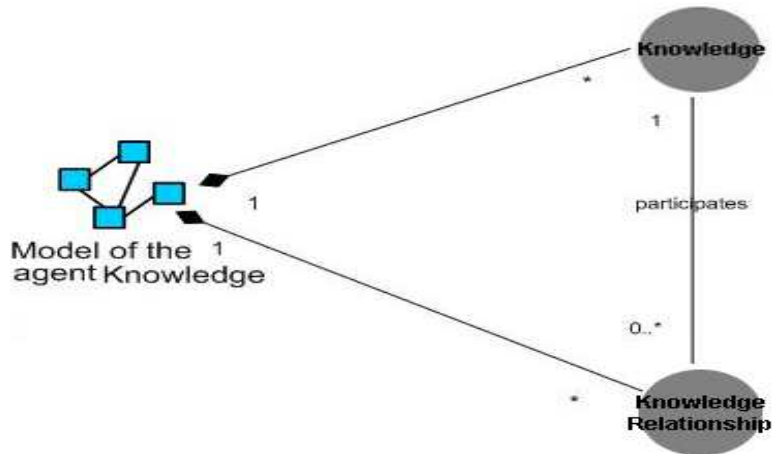


Figura 11 – Metamodelo do Conhecimento do Agente [61]

4.4.11 O Modelo de ações do agente

O Modelo de Ações do Agente representa as ações que o agente terá no seu ambiente de acordo com as responsabilidades de que tiver sido encarregado. Nesse modelo as percepções que o agente tem do ambiente: um evento ou as possíveis mensagens que ele poderá receber de outros agentes ou de entidades externas que irão disparar as suas ações são representadas.

Existem duas formas de mapeamento dessas percepções para ações. A primeira é quando uma percepção é mapeada diretamente para uma ação. Nesse caso, o agente é considerado tipo reativo. A outra forma é quando ocorre um planejamento e na busca da ação adequada de acordo com a percepção do agente para atingir um objetivo. Nesse caso, o agente é considerado do tipo deliberativo e usa um motor de inferência no seu processo de raciocínio. Para realizar uma ação o agente pode necessitar de uma destreza (uma destreza é uma habilidade que o agente deverá possuir para a execução de suas responsabilidades) e também atender a uma determinada pré-condição. Ao termino de uma ação, uma pós-condição também pode ser necessária [60].

É construído um Modelo de Ações do Agente para cada agente da sociedade. Na Figura 12 é ilustrado o metamodelo das ações do agente.

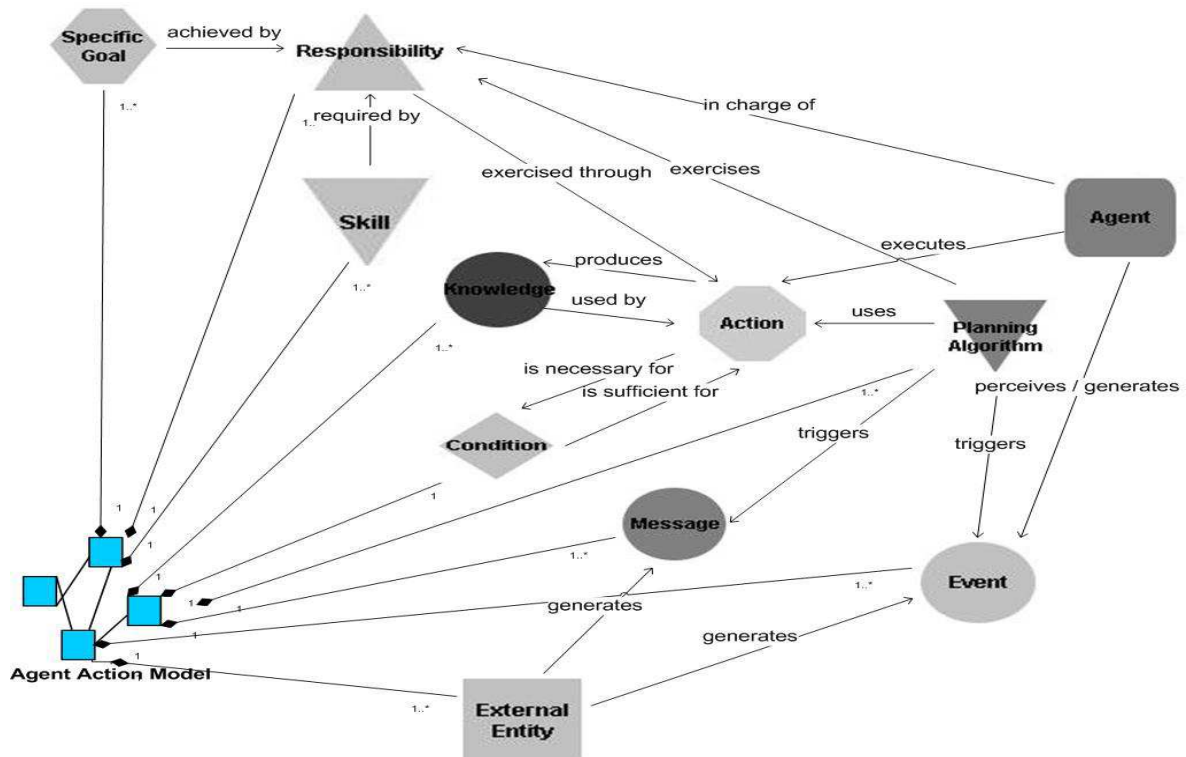


Figura 12 – Metamodelo de Ações do Agente [61]

Na fase de Implementação do Domínio, a técnica DIMAS recomenda o mapeamento de modelos de projeto para agentes, comportamentos e atos de comunicação envolvidos no framework JADE [48], o qual tem sido adotado como plataforma de implementação. Um Modelo de Implementação da Sociedade Multiagente é construído como produto desta fase da MADEM, composto de um Modelo de Comportamentos e de um Modelo de Atos de Comunicação.

4.4.12 O Modelo de comportamentos

O Modelo de Comportamentos representa os comportamentos dos agentes em uma determinada linguagem/plataforma de implementação. Para cada responsabilidade identificada no Modelo de Atividades é associado um comportamento. Este modelo é construído a partir de Modelos de Comportamentos preexistentes e a partir do mapeamento de agentes, responsabilidades e atividades presentes no Modelo de Atividades. Um agente é mapeado para uma classe de agente JADE, uma responsabilidade é mapeada para um comportamento JADE e uma atividade é mapeada para um método JADE. Estão incluídos na base de conhecimento da ONTORMAS os possíveis comportamentos e métodos JADE, como instâncias das classes “JADE Behaviour” e “Method JADE Behaviour”, respectivamente.

Na Figura 13, o metamodelo de comportamentos é ilustrado. Nesse metamodelo, o conceito “JADE Agent”, representa um agente JADE; “JADE Behaviour”, representa um comportamento JADE e “Method JADE Behaviour”, representa um método do comportamento JADE.

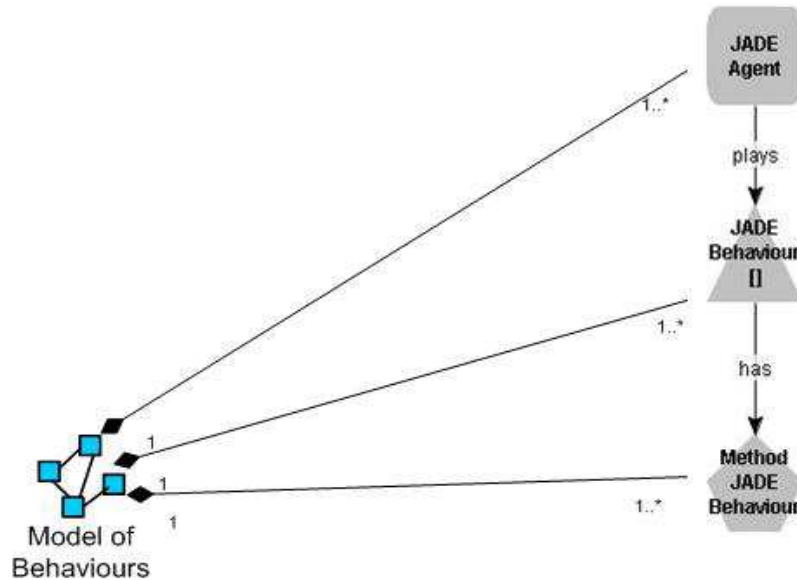


Figura 13 – Metamodelo de Comportamentos [61]

4.4.13 O Modelo de atos de comunicação

No Modelo de Atos de Comunicação as mensagens trocadas entre agentes presentes no Modelo de Interações entre Agentes são mapeadas para a linguagem FIPA-ACL [28]. As interações identificadas no Modelo de Interações entre Agentes são mapeadas para uma linguagem de comunicação entre agentes. A partir do Modelo de Comportamentos e do Modelo de Atos de Comunicação a codificação de agentes genéricos a uma família de aplicações é realizada. Agentes desenvolvidos na Engenharia de Aplicações também podem ser reusados nesta tarefa [60].

Na Figura 14, o metamodelo de atos de comunicação é ilustrado. Nesse metamodelo o conceito “JADE Agent”, representa um agente JADE; “JADE Behaviour”, representa um comportamento JADE; “Method JADE Behaviour”, representa um método do comportamento JADE e “Performative”, representa uma performativa presente na especificação da linguagem de comunicação entre agentes FIPA-ACL.

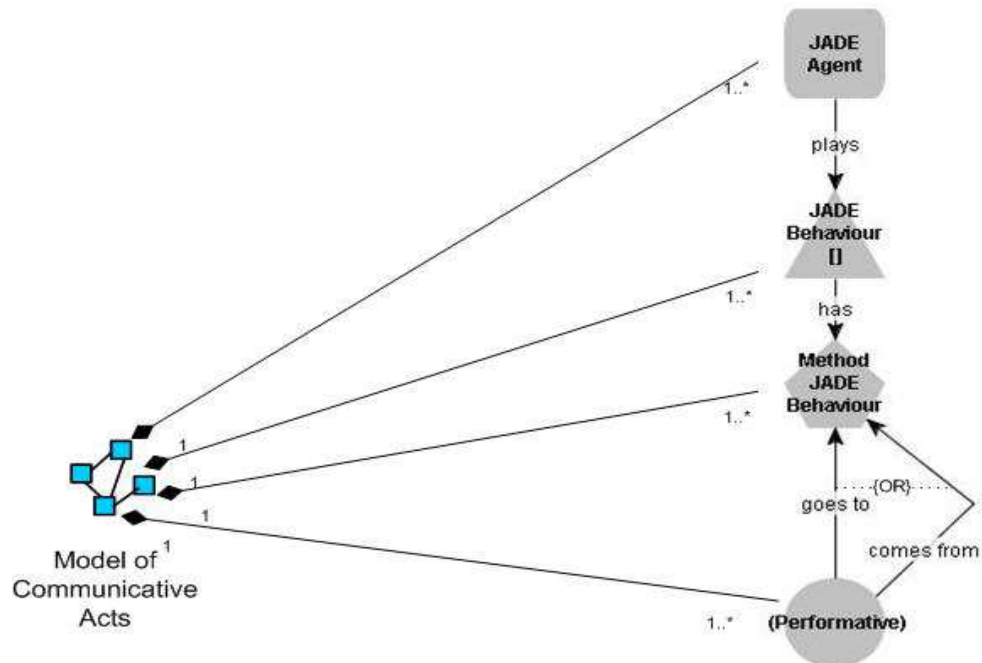


Figura 14 – Metamodelo de Atos de Comunicação [61]

4.5 A ONTORMAS

A ONTORMAS é uma ontologia que representa uma conceitualização das metodologias MADEM e MAAEM. Ela é uma extensão da ONTOMADEM (*Ontology for Multi-Agent Application Domain Engineering Methodology*) [40] que conceitualizava apenas a metodologia MADEM [60].

Na ONTORMAS foram expressas todas as diretrizes para o desenvolvimento de sistemas multiagente especificadas por essas metodologias. A ontologia foi desenvolvida no ambiente de desenvolvimento de sistemas baseados no conhecimento Protégé [82], como um sistema de frames, no qual cada frame representa um conceito e os slots representam as características e relacionamentos entre esses conceitos.

A ONTORMAS é formada por um conjunto de classes organizadas hierarquicamente, tendo como superclasses principais: “Variable Concepts”, “Modeling Concepts”, “Modeling Tasks” e “Modeling Products”. A superclasse “Variable Concepts” e suas correspondentes subclasses são utilizadas para especificar a variabilidade de uma família de aplicações multiagente. Isso é realizado através da definição de “pontos de variação” e de “variantes”. Um ponto de variação é a representação de um conceito sujeito a variação. Uma variante representa uma variação do conceito propriamente dito. Uma variante poderá representar um grupo de variações alternativas ou opcionais. A superclasse “Modeling Concepts” especifica os conceitos fundamentais de modelagem presentes nas metodologias

MADEM e MAAEM, apresentado na Figura 15. Esses conceitos são básicos para o desenvolvimento de sistemas multiagente e estão especificados em níveis de abstração indo do nível mais abstrato (Fase de Análise dos Requisitos da Aplicação) ao mais concreto (Fase de Implementação). Os conceitos de modelagem podem ter entre si a relação de sinonímia, generalização ou especialização. Na ONTORMAS, conceitos específicos de uma plataforma de implementação de agentes, como o JADE, também podem ser representados para possibilitar a geração automática de código [60].

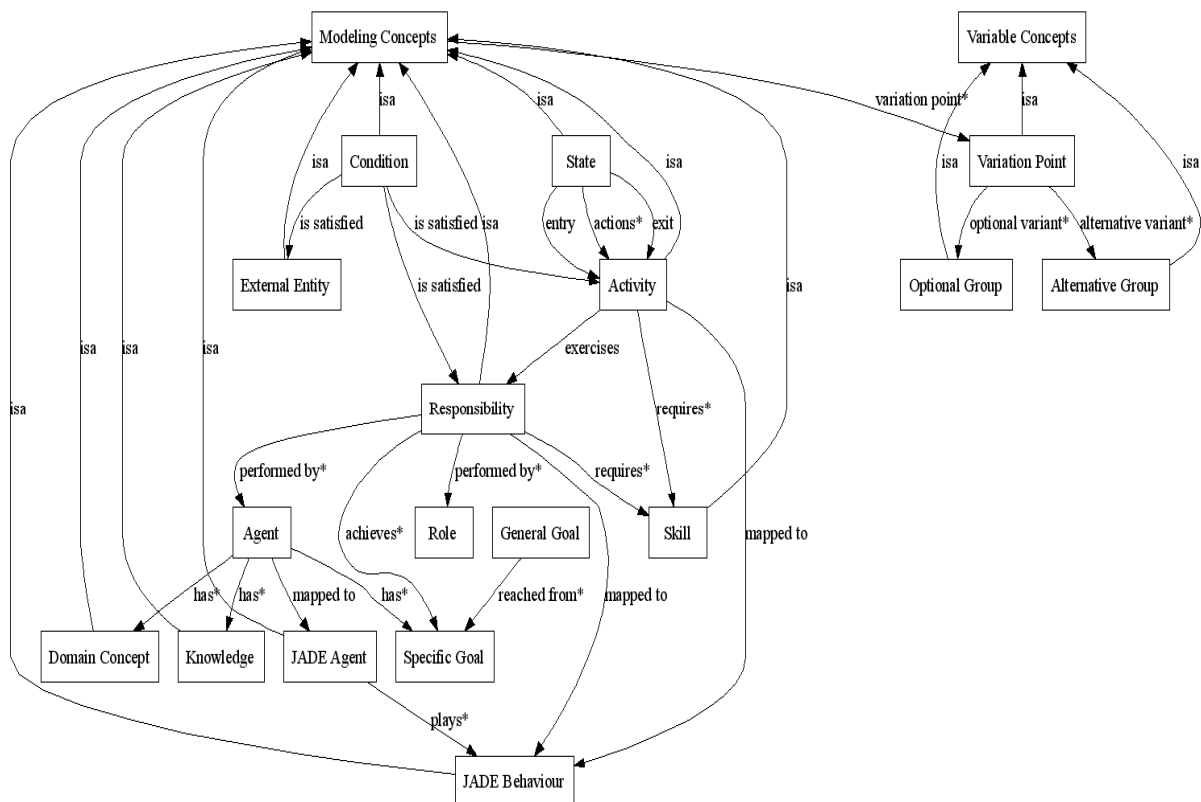


Figura 15 – Rede semântica com os principais conceitos de modelagem especificados pelas metodologias MADEM e MAAEM [61]

Na superclasse “Modeling Tasks” e em suas correspondentes subclasses são definidas as tarefas de modelagem especificadas pelas metodologias MADEM e MAAEM.

A Figura 16 ilustra a representação das tarefas realizadas na fase de Análise de Domínio e na de Análise dos Requisitos da Aplicação. Essas tarefas são divididas em “Domain Engineering Tasks”, cujas subtarefas são relacionadas à metodologia MADEM e “Application Engineering Tasks”, relacionadas à metodologia MAAEM.

Na superclasse “Modeling Products” estão definidas as classes que representam os possíveis produtos de modelagem produzidos através das metodologias MADEM e MAAEM. Alguns destes produtos como “Domain Model” são compostos por subprodutos, isso é especificado através do relacionamento “subproducts”.

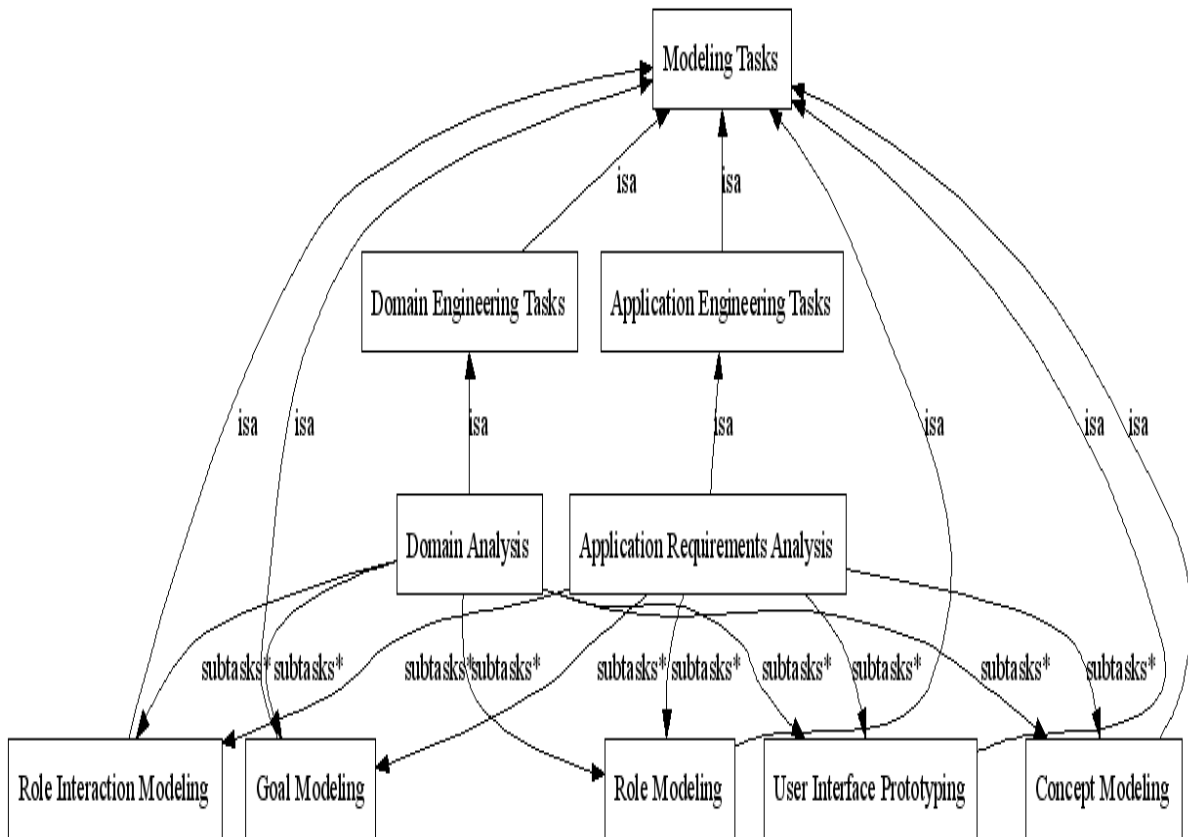


Figura 16 - Rede semântica das tarefas e subtarefas referentes à fase de Análise das metodologias MADEM e MAAEM [61]

A ONTORMAS é utilizada no MADAE-Pro para realizar três tarefas básicas. A primeira é para realizar a criação dos modelos definidos no processo. A segunda é como repositório desses artefatos e, por último, como ferramenta de seleção, adaptação e integração dos componentes. A aplicação das metodologias MADEM e MAAEM e os produtos das mesmas são obtidos através da instanciação dos correspondentes conceitos na hierarquia de classes da ONTORMAS.

4.6 Considerações Finais

Este capítulo apresentou um processo para o desenvolvimento e reuso de famílias de sistemas multiagente, MADAE-Pro, o qual integra dois subprocessos complementares. Um é baseado nos conceitos da Engenharia de Domínio, e o outro, baseado na Engenharia de Aplicações.

O MADAE-Pro tem como característica ser um processo baseado no conhecimento, isto é, a definição das suas fases, tarefas e produtos são classes da ONTORMAS e os relacionamentos entre esses conceitos são especificados através dos “slots” dessa ontologia. Durante o desenvolvimento das famílias de sistemas multiagente as classes

da ONTORMAS são instanciadas formando uma base de conhecimento, o que faz da ONTORMAS também um repositório para os modelos desenvolvidos com o MADAE-Pro.

Devido a capacidade de capturar e armazenar os produtos da Engenharia de Domínio e de Aplicações Multiagente, escolheu-se a ONTORMAS como ferramenta para desenvolvimento do SMPLP. Essa ferramenta é baseada em ontologias e irá suportar as tarefas de modelagem conforme a metodologia MAAEM, que é uma metodologia que guia o desenvolvimento de aplicações multiagente, com suporte ao reuso, desde a especificação dos requisitos até a sua implementação. Outra razão para a escolha da ONTORMAS é a sua integração ao ambiente de desenvolvimento de ontologias do Protegé, que fornece toda a infraestrutura necessária para que qualquer ontologia seja criada, recuperada e integrada.

Espera-se assim, com o uso da MAAEM e da ONTORMAS facilitar a compreensão dos produtos, realizando-se a modelagem dos artefatos definidos pela MAAEM.

5 MODELAGEM DO SISTEMA MULTIAGENTE NO PROCESSO DE LICITAÇÃO PÚBLICA (SMPLP)

Esta seção apresenta a especificação de requisitos e o projeto de um sistema multiagente para o processo de licitação pública com a aplicação do processo de desenvolvimento de sistemas multiagente MADAE-Pro.

5.1 Fase de Análise da Aplicação

Na fase de análise da Engenharia de Aplicações Multiagente, pretende-se elaborar a especificação dos requisitos de uma determinada aplicação, partindo de modelos de domínio, que são artefatos de software reutilizáveis elaborados na fase correspondente da Engenharia de Domínio Multiagente [64].

Nas subseções seguintes, é descrita a aplicação das fases da Análise da Aplicação (Modelagem de Conceitos, a Modelagem de Objetivos, a Modelagem de Papéis e a Modelagem de Interações entre Papéis) à modelagem do sistema SMPLP (Sistema Multiagente para o Processo de Licitação Pública) [64][73][75][92].

5.1.1 Modelagem de conceitos

A Figura 17 mostra o modelo de conceitos do SMPLP, uma rede semântica representando os principais conceitos do processo licitatório, o qual se inicia pela necessidade que um ou mais setores tem de adquirir produtos ou serviços, sendo formalizada através de um pedido de compra que é enviado ao setor de compras. Ao receber o pedido de compra, o setor de compras, que tem a responsabilidade de controlar o estoque da organização, faz a solicitação a comissão permanente de licitações, através de um projeto básico [23][25][31].

A comissão permanente ao receber o projeto básico, se tiver dúvidas quanto ao conteúdo técnico ou jurídico, poderá solicitar maiores informações ao setor técnico ou jurídico. Uma vez sanada todas as dúvidas, ela avalia se o projeto tem relevância para o interesse público quanto a ótica da oportunidade e conveniência. Caso a avaliação seja positiva, ela avalia a possibilidade de adquirir os produtos ou serviços através da contratação direta, dispensando o processo licitatório, previsto na Lei nº 8.666, de 1993, art. 24 e 25, através da dispensa ou da inexigibilidade de licitação [58].

Uma vez observada pela comissão permanente, a impossibilidade legal da contratação direta, inicia-se a fase interna do processo licitatório, onde na abertura do

processo é elaborada uma especificação do objeto requerido, com base no projeto apresentado. Logo após, uma pesquisa de mercado dos itens ou serviços a serem licitados será feita. Uma vez realizada, a comissão permanente verificará se tem orçamento financeiro para adquirir o bem ou serviço solicitado. Caso o órgão possua, definir-se-á o tipo (menor preço, melhor técnica, técnica e melhor preço) e a modalidade da licitação (concorrência, tomada de preços, convite) que será adotada, como a geração e publicação do edital, finalizando a fase interna.

Quando o edital é publicado, inicia-se a fase externa, onde surge a figura dos licitantes, que são os interessados em participar do processo licitatório, que fornecem produtos ou serviços e ficam monitorando os editais para poder participar do processo. Na etapa de documentos, o licitante faz o levantamento de toda a documentação necessária, prepara a proposta e envia para a comissão permanente avaliar se eles podem participar do processo licitatório. Na etapa de julgamento, uma vez qualificado, o licitante irá disputar com os demais licitantes, baseado na modalidade e no tipo da licitação, e o licitante que melhor se adequar ao que foi licitado será considerado o ganhador do processo licitatório. Por último, na etapa contratação, o licitante terá que apresentar os documentos finais para assinar o contrato de prestação de serviço ou venda de produto [25][31].

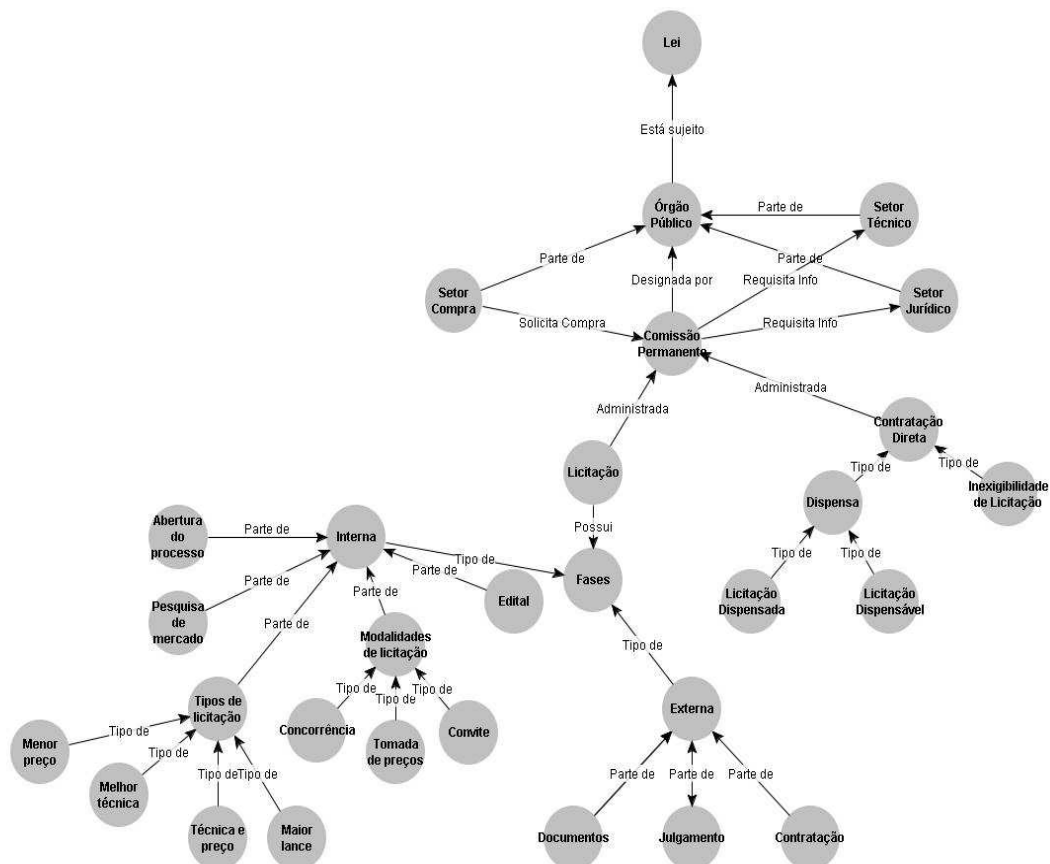


Figura 17 – Modelo de Conceitos do Sistema SMPLP

5.1.2 Modelagem de objetivos

A Figura 18 apresenta o modelo de objetivos do sistema multiagente para o processo de licitação pública, no qual é mostrado o objetivo geral do sistema, as quatro entidades externas envolvidas no processo, como seus objetivos específicos e as responsabilidades que deverão ser realizadas para o alcance desses [23].

O objetivo geral do sistema é “otimizar a tomada de decisões no processo de licitação pública”. Para alcançar esse objetivo, o setor da administração que desejar adquirir um produto ou serviço, tem como objetivo específico realizar o pedido de compra, através da elaboração de um pedido de compras que será enviado ao setor de compras. Uma vez recebido o pedido, o setor de compras tem como objetivo específico, definir o objeto de compra através da elaboração de um projeto básico e submissão a comissão permanente.

A comissão permanente ao receber o projeto básico tem como objetivo específico avaliar a proposta de compra, por analisar se a mesma tem relevância para o interesse público quanto a ótica da oportunidade e a conveniência. Após analisado e aprovado, a comissão permanente analisará a possibilidade de adquirir o bem ou serviço solicitado através da contratação direta, previsto na Lei nº 8.666, de 1993, art. 24 e 25 [25][31][58].

Uma vez aprovado o projeto básico e não sendo possível efetuar a contratação direta, a comissão permanente tem como objetivo específico gerar o edital, através da estimativa do custo do bem ou serviço, o qual será realizada por um dos agentes do sistema. Uma vez que o agente cote a média de preço do bem ou serviço a ser licitado, ele sugerirá a modalidade e o tipo da licitação a comissão permanente, bastando apenas que a mesma confirme o que foi sugerido, ou altere de acordo com sua necessidade. Para finalizar o objetivo específico, o agente ainda terá como missão elaborar e publicar o edital de licitação.

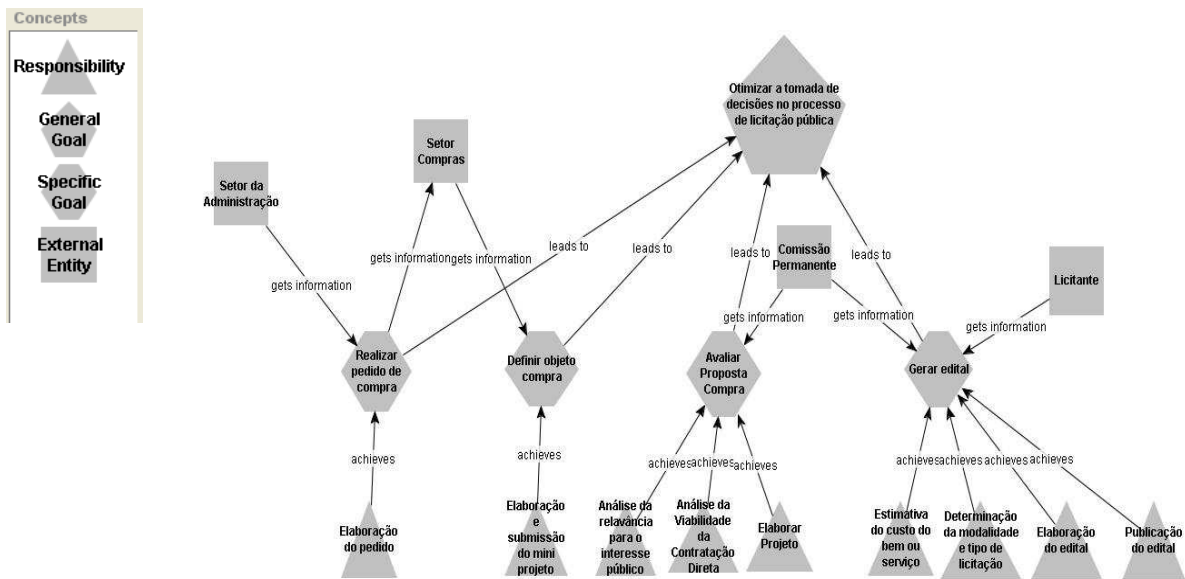


Figura 18 – Modelo de Objetivos do Sistema SMPLP

A entidade externa “Setor da Administração” representa todos os usuários do sistema, de suas respectivas repartições públicas. É esta entidade que irá fornecer as informações necessárias para a realização do pedido de compra, como por exemplo, características, quantidade do produto ou serviço a ser adquirido.

A entidade externa “Setor de Compras” é responsável por monitorar o estoque da repartição pública a que pertence, como receber o pedido de compra enviado de um dos setores da administração pública e elaborar um miniprojeto baseado no mesmo e encaminhá-lo a comissão permanente.

A entidade externa “Comissão Permanente” é responsável por receber o miniprojeto e a avaliar a relevância do mesmo para o interesse público quanto a ótica da oportunidade e conveniência. Uma vez aprovada, ela verificará se é possível adquirir o bem ou serviço através da contratação direta ou se será necessário abrir um processo de licitação.

A entidade externa “Licitante” é responsável por monitorar os editais que são publicados pela comissão permanente disponibilizando-os para os licitantes interessados em participar do processo de licitação pública ou a contratação direta.

5.1.3 Modelagem de papéis

Cada uma das responsabilidades identificadas na modelagem de objetivos é atribuída a um papel, o que é mostrado na Figura 19, 20, 21, e 22. Durante o desenvolvimento de suas responsabilidades, cada papel utiliza e produz certos conhecimentos. Além disso,

algumas pré-condições precisam ser realizadas para que as responsabilidades sejam efetivadas como pós-condições após o cumprimento da realização dessas responsabilidades [23].

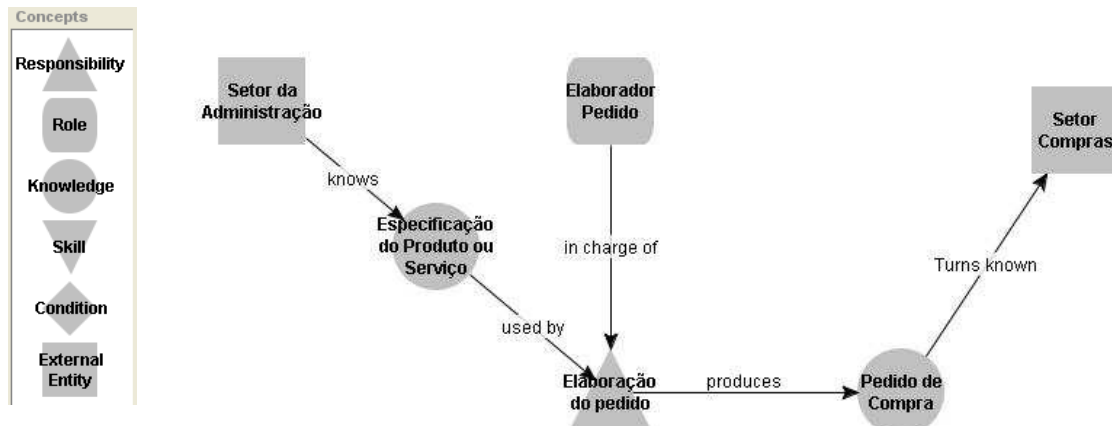


Figura 19 - Modelo de Papéis do SMPLP referente ao objetivo específico “Realizar pedido de compra”

O primeiro papel, “Elaborador do Pedido”, é quem desempenha a responsabilidade de elaborar o pedido de compras, utilizando o conhecimento dos produtos e serviços que são fornecidos pela entidade externa, “Setor da Administração”.

Após cumprida a responsabilidade de elaborar o pedido, o papel “Elaborador do Pedido”, produz o conhecimento, pedido de compra, que é fornecido a entidade externa, “Setor de Compras”.

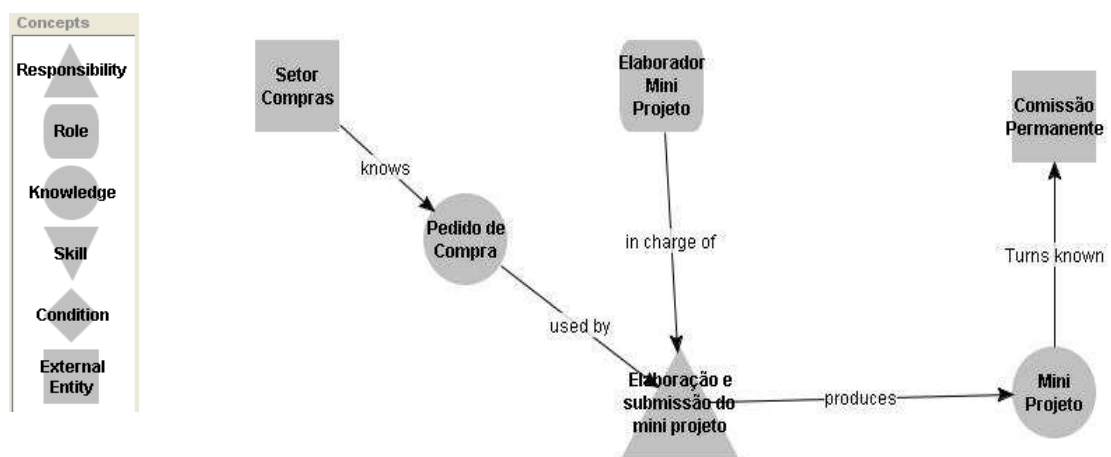


Figura 20 – Modelo de Papéis do SMPLP referente ao objetivo específico “Definir objeto de compra”

O segundo papel, “Elaborador do Miniprojeto”, é quem desempenha a responsabilidade de elaborar e submeter o miniprojeto, utilizando o conhecimento pedido de compra recebido da entidade externa, Setor de Compras.

Após cumprida a responsabilidade de elaborar e submeter o miniprojeto, o papel “Elaborador do Miniprojeto”, produz o conhecimento, miniprojeto, que é fornecido a entidade externa, “Comissão Permanente”.

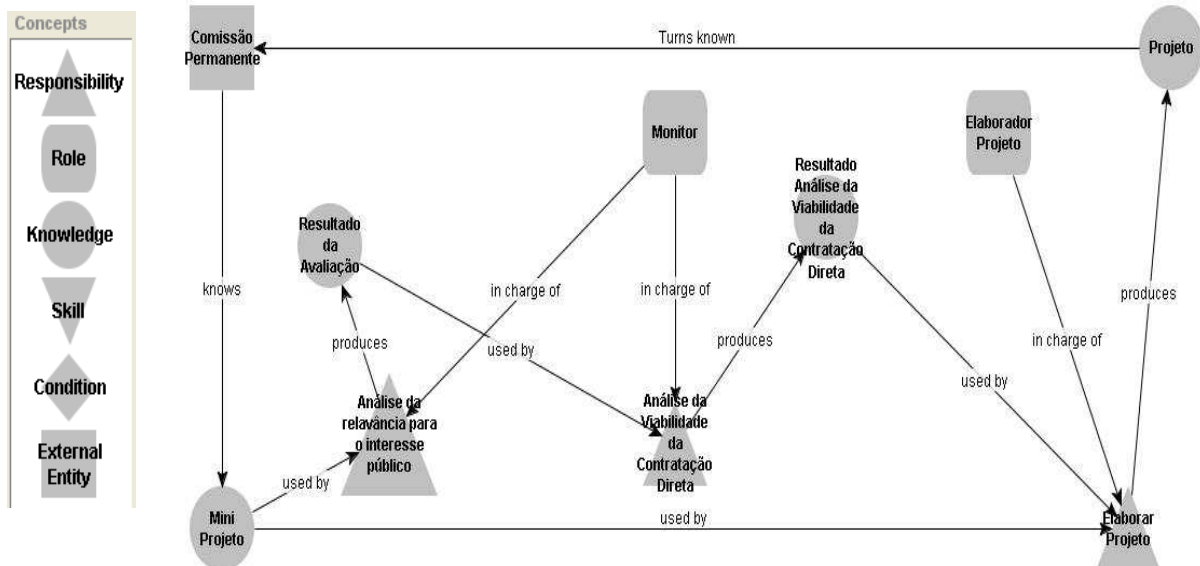


Figura 21 – Modelo de Papéis do SMPLP referente ao objetivo específico “Avaliar proposta de compra”

O terceiro papel, “Monitor”, é quem desempenha a responsabilidade de analisar a relevância para o interesse público quanto a ótica da oportunidade e conveniência para adquirir o bem ou serviço solicitado, utilizando o conhecimento miniprojeto recebido da entidade externa, “Comissão Permanente”.

Após cumprida a responsabilidade de analisar a relevância para o interesse público quanto a ótica da oportunidade e conveniência para adquirir o bem ou serviço solicitado, o papel “Monitor”, produz o conhecimento resultado da avaliação, que é fornecido a entidade externa, “Comissão Permanente”.

O quarto papel, é o “Elaborador do Projeto”, que desempenha a responsabilidade de elaborar um projeto, utilizando o conhecimento resultado da avaliação recebido da entidade externa, “Comissão Permanente”.

Após cumprida a responsabilidade de elaborar o projeto, o papel “Elaborador do Projeto”, produz o conhecimento projeto, que é fornecido a entidade externa, “Comissão Permanente”.

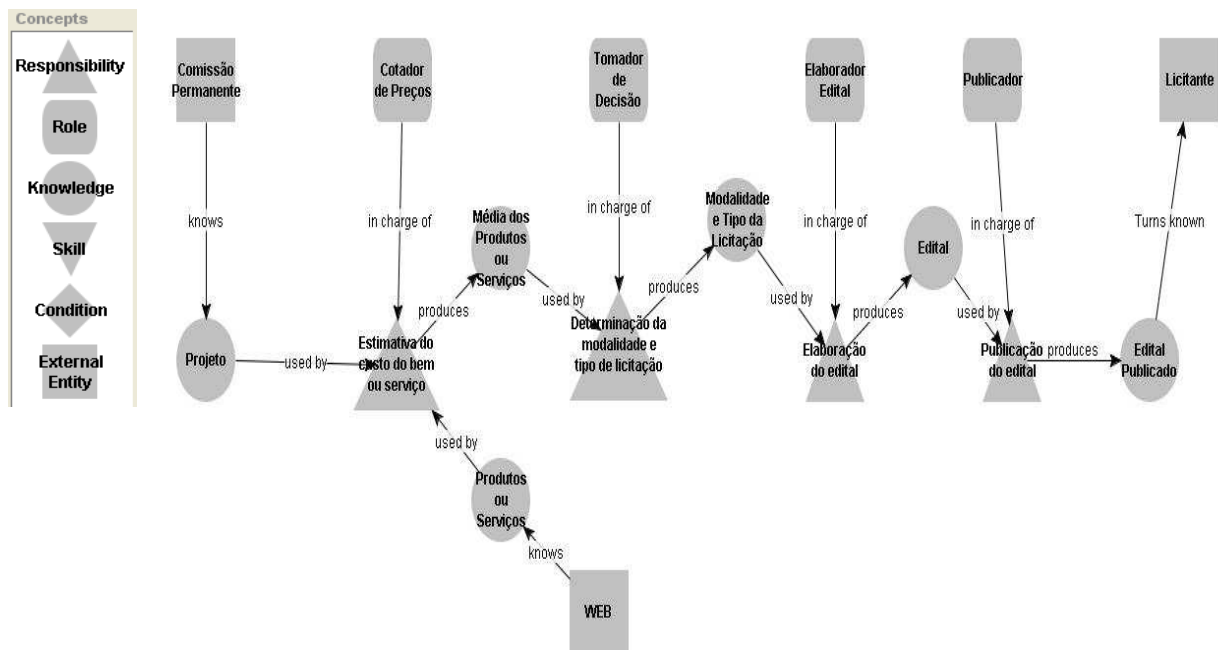


Figura 22 – Modelo de Papéis do SMPLP referente ao objetivo específico “Gerar Edital”

O quinto papel, “Cotador de Preços”, é quem desempenha a responsabilidade de fazer a estimativa do custo do bem ou serviço solicitado, utilizando o conhecimento projeto recebido da entidade externa, “Comissão Permanente”. Para realizar a responsabilidade de fazer a estimativa do custo do bem ou serviço solicitado, o papel “Cotador de Preços”, utiliza o conhecimento produtos ou serviços, que é fornecido da entidade externa, “WEB”.

Após cumprida a responsabilidade de fazer a estimativa do custo do bem ou serviço solicitado, o papel “Cotador de Preços”, produz o conhecimento média dos produtos ou serviços, que será usado pelo sexto papel, “Tomador de Decisão”, que tem a responsabilidade de determinar a modalidade e o tipo de licitação.

Uma vez cumprida a responsabilidade de determinar a modalidade e o tipo de licitação, pelo papel “Tomador de Decisão”, ele produz o conhecimento modalidade e tipo de licitação que será usado pelo sétimo papel, “Elaborador do Edital”, que tem a responsabilidade de elaborar o edital de licitação.

Tendo cumprida a responsabilidade de elaborar o edital de licitação, pelo papel “Elaborador do Edital”, ele produz o conhecimento edital, que será usado pelo oitavo papel, “Publicador”, que tem a responsabilidade de publicar o edital.

Ao cumprir a responsabilidade de publicar o edital, pelo papel “Publicador”, ele produz o conhecimento edital publicado que é fornecido a entidade externa, “Licitante”.

5.1.4 Modelagem de interações entre papéis

Após a modelagem dos papéis, é necessário que se defina o relacionamento entre eles, pois é através destas interações que os papéis conseguirão atingir o objetivo geral do sistema. Cada modelo de interação entre papéis está associado a um objetivo específico. As interações são numeradas de acordo com a ordem temporal em que acontecem [23].

A Figura 23 mostra o modelo de interações relacionado ao objetivo específico “Gerar Pedido”, onde qualquer setor da administração pública pode elaborar um pedido de compra, fornecendo as especificações do produto ou serviço desejado, como as quantidades dos mesmos.

Estas informações são interceptadas pelo papel “Elaborador do Pedido”, que é responsável por elaborar o pedido de compra e enviar ao Setor de Compras.

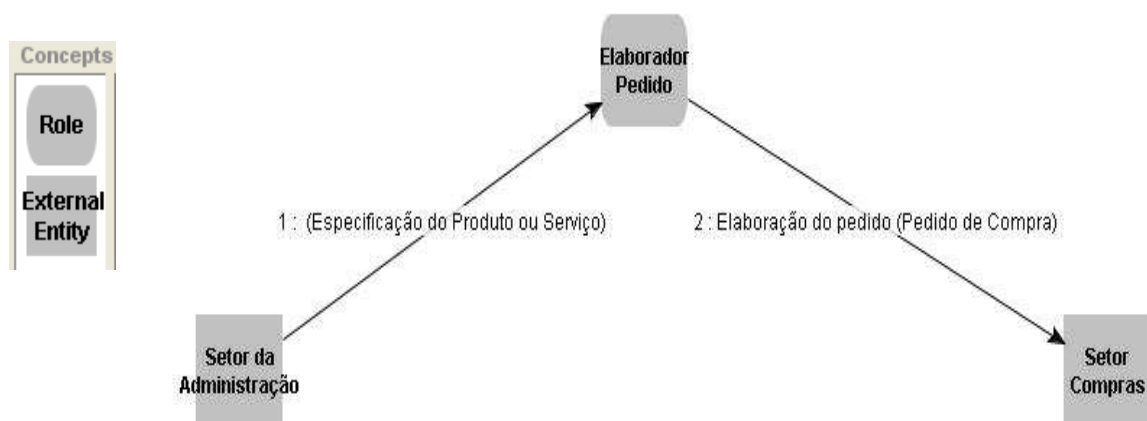


Figura 23 – Modelo de Interação entre Papéis do SMPLP referente ao objetivo específico “Realizar pedido de compra”

O próximo modelo de interação entre papéis está relacionado ao objetivo específico “definir objeto de compra”, visualizado na Figura 24. Após receber o pedido de compra do Setor de Compras, o papel “Elaborador do Miniprojeto”, elabora e submete o miniprojeto a Comissão Permanente.

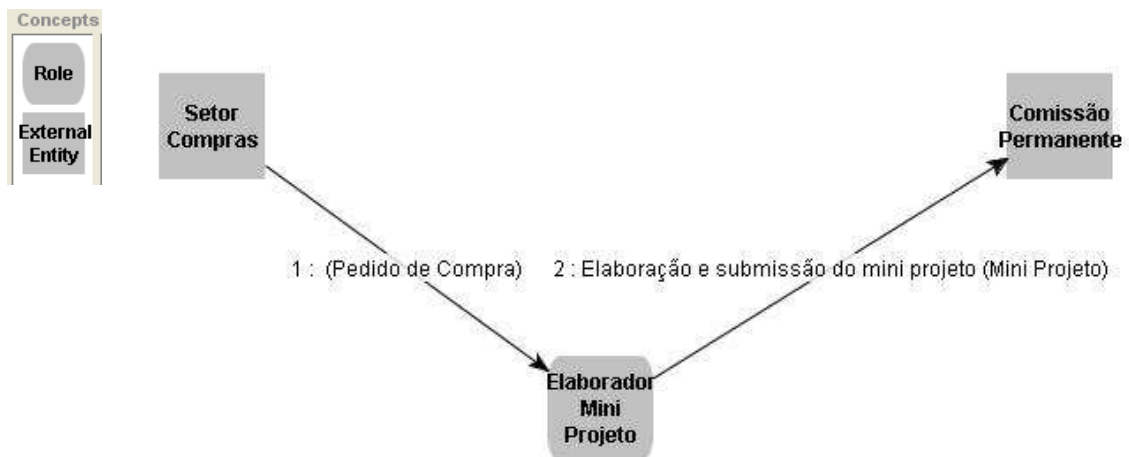


Figura 24 – Modelo de Interação entre Papéis do SMPLP referente ao objetivo específico “Definir objeto de compra”

O próximo modelo de interação entre papéis está relacionado ao objetivo específico “avaliar proposta de compra”, visualizado na Figura 25. Após receber o miniprojeto da Comissão Permanente, o papel “Monitor”, faz a análise da relevância para o interesse público, quanto a conveniência e a oportunidade. Uma vez realizada, ele envia o Resultado da Avaliação novamente a Comissão Permanente, e caso o mesmo tenha sido aprovado, ela encaminhará o miniprojeto ao papel “Elaborador do Projeto”, que elaborará um projeto final e enviará a Comissão Permanente.

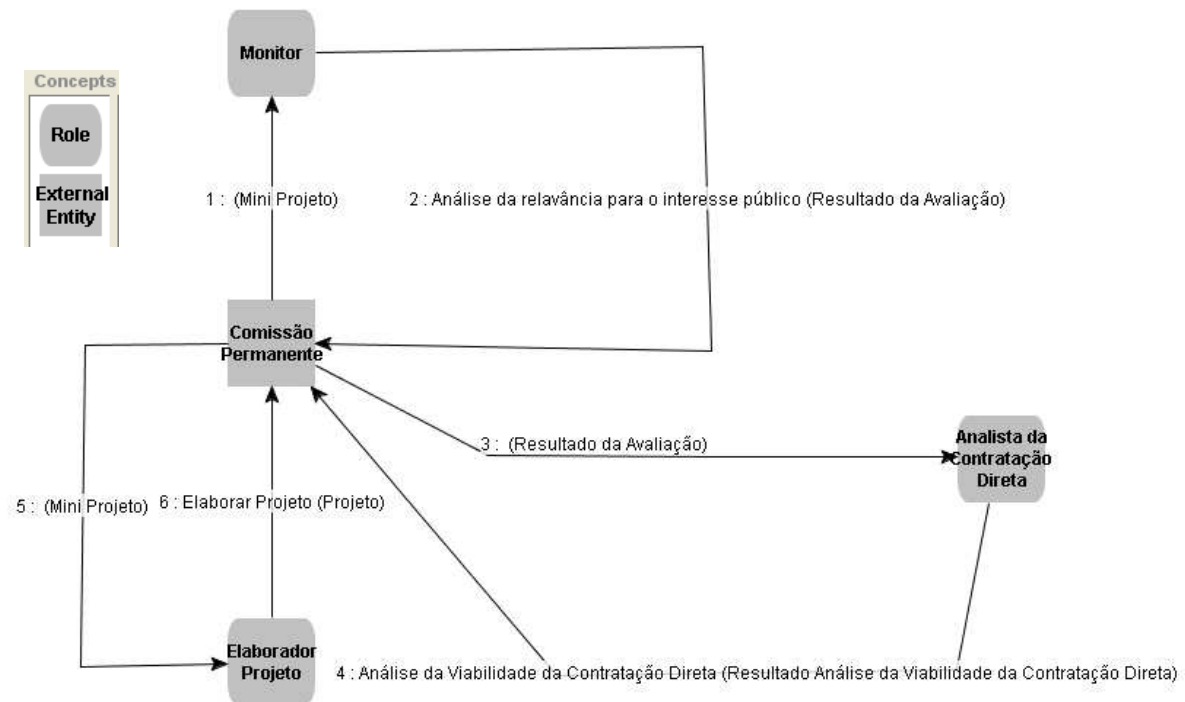


Figura 25 – Modelo de Interação entre Papéis do SMPLP referente ao objetivo específico “Avaliar proposta de compra”

O próximo modelo de interação entre papéis, visualizado na Figura 26, está relacionado ao objetivo gerar edital, o qual está relacionado diretamente com o objetivo geral do sistema, visualizado na Figura 18. Após receber o projeto da Comissão Permanente, o papel “Cotador de Preços”, faz a pesquisa de preços usando a Web. Após realizada, o papel gera a média de preço dos produtos ou serviços e envia para o papel “Tomador de Decisão”.

Baseado na média de preço e a quantidade de produtos ou serviços que serão licitados, o papel “Tomador de Decisão”, determina a modalidade e o tipo de licitação e encaminha essas informações para o papel “Elaborador do Edital”. Esse por sua vez tem a responsabilidade de elaborar o edital e encaminhar para o papel “Publicador”, o qual publicará o edital e o enviará para o Licitante.

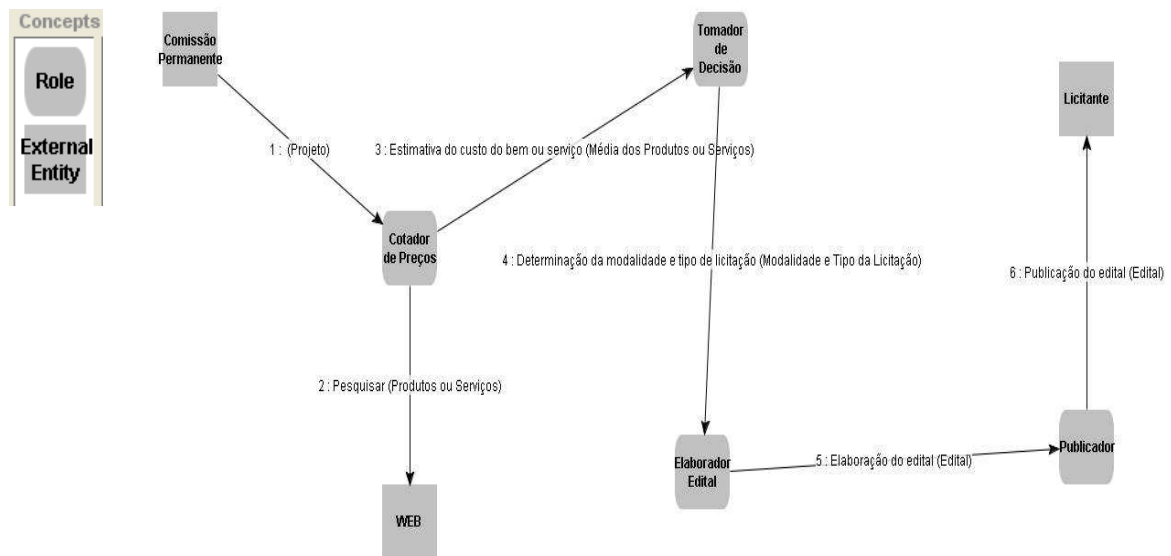


Figura 26 – Modelo de Interação entre Papéis do SMPLP referente ao objetivo específico “Gerar Edital”

5.1.5 Protótipo da Interface com o Usuário

O Protótipo da Interface com o Usuário representa as interações dos usuários com o sistema. Esse protótipo foi desenvolvida usando o software NetBeans IDE 7.1 [72].

O funcionamento do SMPLP dar-se-á da seguinte maneira:

1º - Algum setor do poder público, seja ele o financeiro, recursos humanos, administração, contabilidade ou qualquer outro setor, fará uma solicitação ao setor de compras, através da interface “Elaboração do Pedido”, onde será indicado o item, a quantidade, o solicitante, o responsável pelo setor e o setor.

2º - Após receber o pedido de compras, o setor de compras acessará a interface “Elaboração e Submissão do Miniprojeto”, onde será indicado o resumo do projeto, o detalhamento do objeto e os subsídios para a montagem do plano de licitação. Uma vez finalizado o Miniprojeto, o mesmo será encaminhado para a “Comissão Permanente”.

3º - A “Comissão Permanente” será responsável por analisar o “Miniprojeto” recebido do “Setor de Compras”. Após verificar quem solicitou e principalmente o resumo do projeto, a análise será feita para saber se o projeto tem relevância para o interesse público sobre a ótica da oportunidade e é conveniente. Caso o projeto não seja conveniente ou não seja interessante quanto a ótica da oportunidade, a “Comissão Permanente” reprovará o projeto, através da interface “Análise para Relevância para o Interesse Público”. Se ambas as opções forem positivas o sistema Multiagente passará a fazer análises internas baseado no que foi especificado no “Pedido” e no “Miniprojeto”. A primeira opção que será analisada é se os itens constantes nesse projeto podem ser adquiridos através da “Contratação Direta via

Dispensa”, baseado nos termos que a lei 8.666/93 estabelece. Caso não seja possível, a segunda opção é adquirir os bens ou serviços através da “Contratação Direta via Inexigibilidade de Licitação”, opção também regida pela lei 8.666/93. E caso as duas primeiras opções não tenham sido satisfeitas, então o sistema automaticamente autorizará o “Processo de Licitação”.

4º - Após aprovado o “Processo de Licitação”, a “Comissão Permanente” iniciará a próxima etapa e mais importante do sistema Multiagente. Através da interface “Estimativa do Custo do Bem/Serviço”, a “Comissão Permanente” recuperará os três menores preços do bem ou serviço solicitado. O sistema Multiagente acessará a internet e fará essa pesquisa. Após essa recuperação, o mesmo calculará a média de preço baseado nos três menores preços localizados.

Baseado na quantidade de bens ou serviços indicado no pedido e a média gerada, o sistema Multiagente sugerirá a “Modalidade de Licitação” baseada na lei 8.666/93. A “Comissão Permanente” poderá aceitar a sugestão dada pelo sistema como também poderá escolher outra modalidade, sempre levando em conta as restrições da lei 8.666/93 [58], isto é, ela não poderá escolher uma modalidade que seja inferior a aquisição que será realizada, pois o sistema Multiagente é responsável por monitorar o processo e seguir os ditames da lei.

Assim como a “Modalidade de Licitação”, o sistema Multiagente também sugerirá o “Tipo de Licitação” baseado no que foi apontado no projeto, podendo a “Comissão Permanente” aceitar ou não. Mas uma vez, ela não poderá escolher um tipo que seja diferente da aquisição que será realizada, pois o sistema Multiagente é responsável por monitorar o processo e seguir os ditames da lei.

Confirmado a “Modalidade de Licitação” e o “Tipo de Licitação”, a “Comissão Permanente” finalizará o processo, onde o sistema Multiagente gerará e publicará o edital, podendo o mesmo ser visualizado após sua geração.

A partir da Figura 27, é apresentada a interface do protótipo do SMPLP. A primeira interface, “Login”, será usada para o usuário acessar o sistema. Note que a primeira coisa que ele terá que escolher será o setor que ele vai usar no sistema, isto porque baseado em sua escolha, o sistema disponibilizará interfaces específicas. Após escolhido o setor, o mesmo deverá entrar com seu usuário e senha.

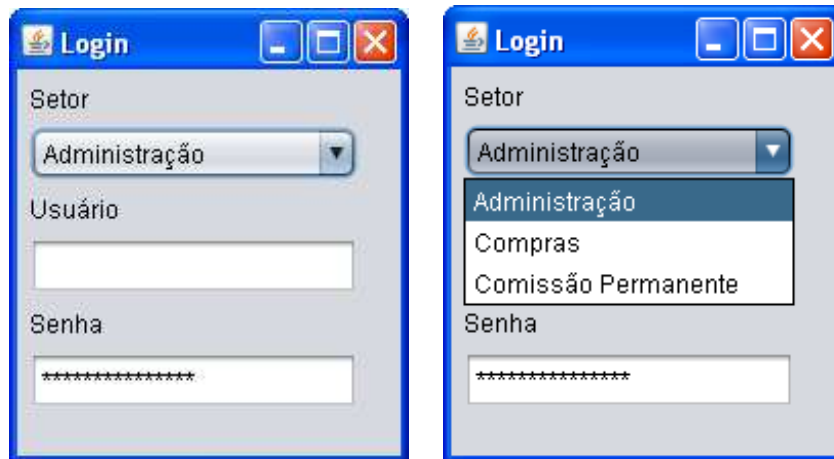


Figura 27 – Protótipo de Interface com o Usuário “Login”

Na Figura 28 é apresentada a interface que será usada por qualquer setor da Administração Pública para acessar a interface “Elaboração do Pedido”.



Figura 28 – Protótipo de Interface com o Usuário para o perfil “Setor da Administração”

Na Figura 29 é apresentada a interface “Elaboração do Pedido”, opção usada por qualquer usuário cadastrado em um “Setor da Administração”, para solicitar os produtos ou serviços a serem adquiridos, ação que é supervisionada pelo agente “Interface”. É importante ressaltar que no momento que o mesmo estiver fazendo a solicitação ele deve indicar qual o setor solicitante, quem o está fazendo e quem é o responsável.

Elaboração do Pedido

Inserir Finalizar Pedido Nº 1979 Data: 03/06/2012

Setor Administrativo Responsavel Lucy Solicitante Anna

Produto Condicionador de ar Split piso/teto com controle. 60.000 btus Qtd 50

Produto	QTD
Condicionador de ar Split piso/teto com controle. 60.000 btus	50

Figura 29 – Protótipo de Interface com o Usuário “Elaboração do Pedido”

Na Figura 30 é apresentada a interface que será usada apenas pelo “Setor de Compras” para acessar a interface “Elaboração e Submissão do Miniprojeto”, a qual é utilizada para elaborar o “Miniprojeto” em conformidade com o pedido solicitado.

Sistema Integrado de Licitações Públicas

Definir objeto de compra

Elaboração e submissão do mini-projeto

Figura 30 – Protótipo de Interface com o Usuário “Setor de Compras”

Na Figura 31 é apresentada a interface “Elaboração e Submissão do Miniprojeto” que será usada pelo “Setor de Compras” para elaborar o Miniprojeto em conformidade com os pedidos recebidos. É fundamental que seja informado o “Resumo do Projeto”, “Detalhamento

do Objeto” e o “Subsídios para Montagem do Plano de Licitação”, informações essas que serão de extrema relevância para aprovação ou não do projeto pela “Comissão Permanente”.

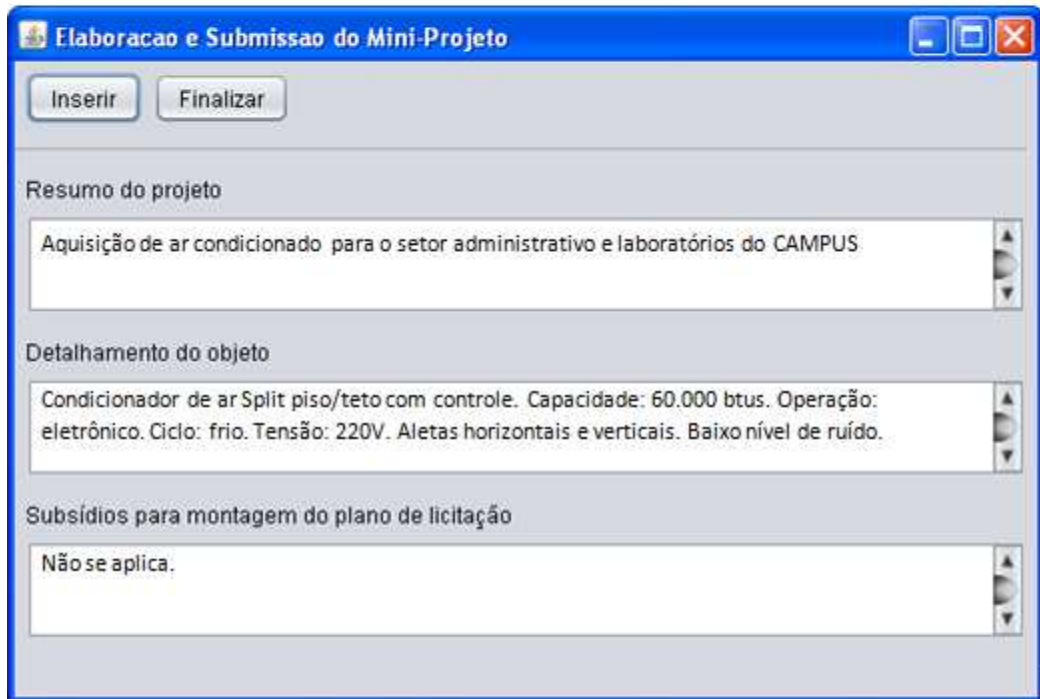


Figura 31 – Protótipo de Interface com o Usuário “Elaboração e Submissão do Miniprojeto”

Após o Setor de Compras elaborar o Miniprojeto, a “Comissão Permanente” usará a próxima interface, apresentado na Figura 32, para acessar a interface “Análise de Relevância para o Interesse Público”, utilizada para analisar o miniprojeto enviado pelo “Setor de Compras”.



Figura 32 – Protótipo de Interface com o Usuário – “Setor de Compras”

A interface “Análise de Relevância para o Interesse Público”, apresentada na Figura 33, será usada especificamente pela “Comissão Permanente” para verificar a viabilidade ou não de adquirir o produto ou serviço solicitado pelo “Setor de Compras”. Nessa interface, a Comissão visualizará o “Número do Projeto”, “Data”, “Solicitante” e o “Resumo do Projeto”. Para que o mesmo seja aprovado, a Comissão deverá verificar se o presente projeto tem relevância para o interesse público sobre a Ótica da Oportunidade e Conveniência. Caso uma das opções escolhidas seja negativa, o agente “Analista”, apresentado nos tópicos anteriores, negará a continuidade do processo.

Em contrapartida caso o agente “Analista” observe que o presente projeto tem relevância para o interesse público, então ele fará uma análise mais criteriosa, isto é, verificará a possibilidade de adquirir o bem ou serviço através da contratação direta, via dispensa ou inexigibilidade de licitação, apresentado nos capítulos anteriores. Caso não seja possível, então o Agente autorizará o Processo de Licitação.

O protótipo de interface é uma janela com o título "Análise da Relevância para o Interesse Público". No topo, há uma barra azul com o ícone de uma pessoa e o título. Abaixo, há uma tabela com três colunas: "Nº Projeto", "Data" e "Solicitante". A primeira linha da tabela contém os dados: "1982/2012", "03/06/2012" e "Anna". Abaixo da tabela, há um campo de texto rotulado "Resumo Projeto" com o conteúdo "Aquisição de ar condicionado para o setor administrativo e laboratórios do CAMPUS". Na parte inferior, há duas opções de seleção com radio buttons: "Ótica da oportunidade" e "Conveniência", ambas com "Sim" selecionado. Um botão "Finalizar" está localizado no canto inferior direito.

Nº Projeto	Data	Solicitante
1982/2012	03/06/2012	Anna

Resumo Projeto

Aquisição de ar condicionado para o setor administrativo e laboratórios do CAMPUS

O presente projeto tem relevância para o interesse público sobre:

Ótica da oportunidade Sim Não

Conveniência Sim Não

Finalizar

Figura 33 – Protótipo de Interface com o Usuário “Análise de Relevância para o Interesse Público”

Após aprovado o Processo de Licitação, a “Comissão Permanente” deverá acessar o menu “Estimativa do Custo do Bem ou Serviço”, apresentado na Figura 34, para dar continuidade ao processo.

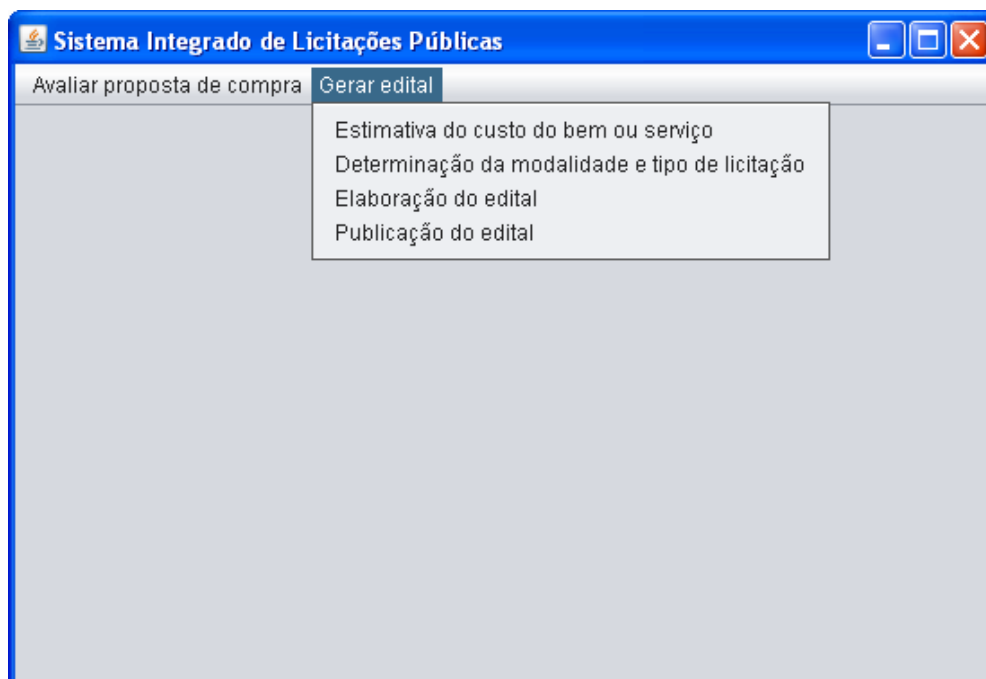


Figura 34 – Protótipo de Interface com o Usuário “Gerar Edital”

A interface “Estimativa do Custo do Bem”, apresentada na Figura 35, é a mais importante do “Sistema Multiagente”, pois após o usuário iniciar o processo, o agente “Licitador”, passará a exercer suas inúmeras responsabilidades e tarefas. Primeiro, ele fará a cotação de preços na internet dos produtos ou serviços solicitados. Após selecionar os três menores preços, ele calculará a média geral dos produtos e baseado na quantidade solicitada com a média geral do preço, ele sugerirá a “Modalidade de Licitação” mais apropriada, por exemplo, se o item selecionado era “Condicionador de Ar 30.000 BTU’s”, e o valor total cotado foi de R\$ 130.000,00 (cento e trinta mil reais), então o “Agente” sugerirá como modalidade de licitação: “Tomada de Preço”. Caso o usuário não aceite a sugestão, o mesmo poderá escolher outra modalidade, desde que essa não viole o teto da contratação, isto é, para o exemplo citado, ele não pode escolher a modalidade “Convite”, pois o valor cotado é superior ao que esta modalidade permite conforme a Lei 8.666/93 [58]. Logo o “Agente” não permitirá uma troca indevida, pois uma das responsabilidades que ele tem é de monitorar o processo e seguir os ditames da lei.

Além disso, o “Agente” também sugerirá o “Tipo de Licitação”, onde no exemplo citado, ele escolherá “Menor Preço”, visando adquirir o produto especificado com o menor preço possível. Para a opção “Tipo de Licitação”, o usuário também poderá mudar a opção sugerida pelo “Agente”, porém, assim como a “Modalidade de Licitação”, ele não poderá escolher um “Tipo” que não seja adequado ao que será licitado. O “Agente” sempre fará o bloqueio em caso de alguma tentativa dessa natureza.

Uma vez confirmada a “Modalidade” e o “Tipo de Licitação”, o “Agente” elaborará o edital e o publicará, possibilitando a participação dos licitantes.



Estimativa do Custo do Bem/Serviço

Iniciar Finalizar

Produto	QTD	Valor1	Valor2	Valor3	Media
Ar cond. 30 mil	50	2.600,00	2.500,00	2.700,00	2.600,00

Modalidade de licitação: Tomada de Preços

Tipo de licitação: Menor Preço

Figura 35 – Protótipo de Interface com o Usuário – “Estimativa do Custo do Bem/Serviço”

5.2 Fase de Projeto da Aplicação

A fase de projeto da Engenharia de Aplicações Multiagente consiste de três subfases, que são: o Projeto da Arquitetura, que define a visão global e externa da sociedade multiagente, sobretudo, passando pela escolha de seus mecanismos de cooperação e coordenação; o Projeto do Agente, que estabelece a visão detalhada e interna de cada agente, modelando sua estrutura e seu comportamento; e a Modelagem do Conhecimento da Sociedade Multiagente, que identifica os conceitos compartilhados por todos os agentes em sua comunicação [35][41][64].

Nesta fase, usaremos todos os modelos produzidos na fase anterior como base para a modelagem do conhecimento da sociedade multiagente e para o projeto da arquitetura do sistema SMPLP.

5.2.1 Modelagem do conhecimento da sociedade multiagente

No Modelo do Conhecimento da Sociedade Multiagente, os conceitos compartilhados pelos agentes da sociedade são identificados e representados em uma rede semântica [64].

Este modelo baseia-se no Modelo de Conceitos e Modelo de Interações de Papéis, modelos da fase de análise. A Figura 36 ilustra o modelo do conhecimento da sociedade multiagente do SMPLP.

Existem três conceitos principais utilizados durante a comunicação dos agentes: pedido de compra, miniprojeto, edital.

O conhecimento “Pedido de Compra” é uma abstração que é cadastrado no sistema. Este conhecimento é baseado na especificação do produto ou serviço e tem como atributos número, descrição do produto ou serviço, quantidade, setor, solicitante. A descrição do produto ou serviço é justamente o objeto que se pretende adquirir.

O conhecimento “Miniprojeto” é baseado no “Pedido de Compras”. Ele tem como atributo o desenvolvimento da solução, soluções técnicas globais e focalizadas, identificação dos tipos de serviços a executar e de materiais, informações que possibilitem o estudo, subsídios para montagem do plano de licitação, orçamento detalhado do custo global.

O conhecimento “Projeto” deriva do “Miniprojeto” e tem como atributo o nome, descrição e data da elaboração. A data da elaboração refere-se a data que o processo de licitação será iniciado. A descrição tem por objetivo conter os detalhes necessários para a realização do processo de licitação.

O conhecimento “Edital” possui os atributos: prazo entre a publicação e a realização da licitação, número e/ordem, nome da repartição, modalidade, objeto da licitação, critério de aceitabilidade, condições de pagamento, regime de execução, tipo da licitação, data e hora para abertura dos envelopes, local e hora para recebimento da documentação. A modalidade da licitação é um dos fatores cruciais no processo licitatório, pois ela que definirá se o processo licitatório será através da concorrência, tomada de preços ou convite.

Do conhecimento “Edital” deriva o conhecimento “Edital Publicado”, que possui o atributo data de publicação, o qual definirá a data que o edital foi publicado e será a data base para os licitantes se candidatarem ao processo licitatório.

O conhecimento “Local de Publicação” é baseado no conhecimento “Edital”, que tem como atributos: repartição, diário oficial da união, diário oficial do estado e jornal. A repartição serve para que o licitante possa tirar o edital diretamente no local.



Figura 36 – Modelo de Conhecimento da Sociedade Multiagente do Sistema SMPLP

5.2.2 Modelagem da sociedade multiagente

Nas Figuras 37 e 38 é apresentado o Modelo da Sociedade Multiagente, que é baseado no Modelo de Papéis. A maior diferença entre as duas abstrações é o fato de na fase de projeto já estarmos trabalhando no nível de agentes, que por sua vez pode assumir um ou mais papéis de acordo com a afinidade entre suas responsabilidades [10].

O Modelo da Sociedade Multiagente, é um refinamento do Modelo de Papéis. Destrezas específicas são adicionadas sendo descritas como soluções concretas para o problema atribuído ao agente respectivo [10].

Pelo fato dos papéis “Elaborador do Pedido” e o “Elaborador do Miniprojeto” possuírem um grau de afinidade alto devido as suas responsabilidades, “Elaboração do Pedido” e “Elaboração e Submissão do Miniprojeto”, ambos são atribuídas ao mesmo agente, o agente “Interface”. Pelo mesmo motivo, os papéis “Monitor” e “Elaborador do Projeto”

foram atribuídos ao agente “Analista”. Assim também se dar com os papéis “Cotador de Preços”, “Tomador de Decisão”, “Elaborador do Edital” e “Publicador”, atribuídos ao agente “Licitação”.

O agente “Interface” é responsável pelas interações do usuário com o sistema. Ele possui as responsabilidades “Elaboração do Pedido” e “Elaboração e Submissão do Miniprojeto”, onde através da interface “Elaboração do Pedido”, o usuário da administração pública insere um pedido de compra. Por sua vez, o usuário responsável pelo setor de compras, insere um miniprojeto através da interface “Elaboração e Submissão do Miniprojeto”. Após a inserção dessas informações, essas responsabilidades serão alcançadas, sendo que o agente “Interface” estará em constante monitoramento para observar a inserção de novos pedidos e miniprojetos.

Após essas etapas, o agente “Analista” é notificado pelo agente “Interface”, para que o mesmo possa analisar se o presente projeto pode ser aprovado. Uma vez aprovado, ele verificará se a aquisição do que foi solicitado será mediante a contratação direta através da dispensa ou da inexigibilidade de licitação.

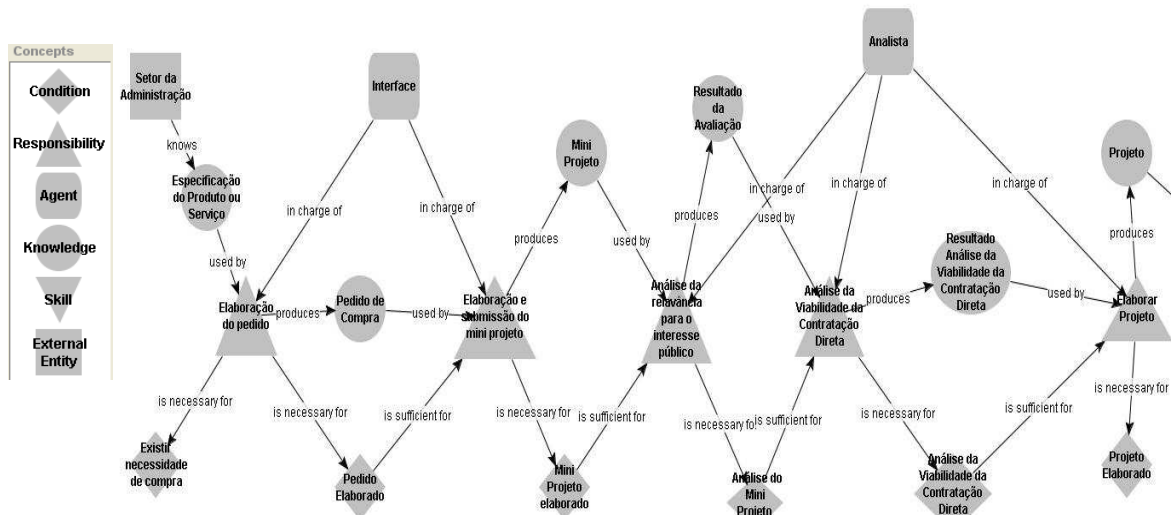


Figura 37 – Modelo de Sociedade Multiagente do SMPLP apresentando os agentes “Interface” e “Analista”

Caso o agente “Analista” resolva que não é possível realizar a contratação direta, ele elaborará um projeto com base no miniprojeto recebido do “agente Interface”, e o encaminhará ao agente “Licitação”, que é o responsável por conduzir o processo licitatório.

Com posse do projeto, o agente “Licitação” fará a estimativa do custo do bem ou serviço existente no projeto, usando a entidade externa WEB [63]. Ao adquirir a média dos produtos ou serviços o agente determinará a modalidade (concorrência, tomada de preços e convite) e o tipo de licitação (menor preço, melhor técnica, técnica e preço, maior lance).

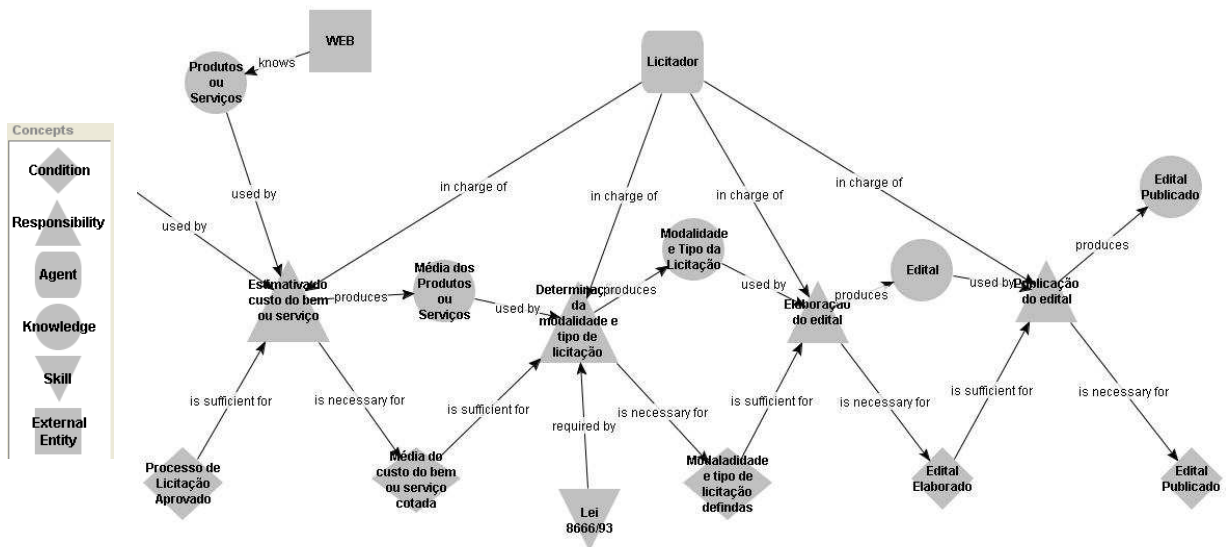


Figura 38 – Modelo de Sociedade Multiagente do SMPLP apresentando o agente “Licitador”

Tendo definido a modalidade e o tipo de licitação, o agente “Licitador” elaborará o edital que regerá o processo licitatório e publicará o mesmo possibilitando a participação dos interessados.

5.2.3 Modelagem das interações entre agentes

O modelo de interações entre agentes, apresentado na Figura 39, foi construído baseado no modelo de interações entre papéis da fase de análise. Neste modelo é representada a forma como os agentes trocam mensagens entre si e as notificações de eventos entre entidades externas e agentes. O padrão FIPA-ACL, o qual possui a definição do formato das mensagens trocadas entre os agentes, foi adotado como padrão para a especificação das mensagens entre agentes. São representadas pela performativa, escrita em letras maiúsculas, e seu respectivo conteúdo entre parênteses [10][27].

A medida que uma necessidade de compra é expressada pela entidade externa “Setor da Administração”, o agente “Interface” passa a tomar conhecimento de tal através de um novo pedido de compra que é inserida por esta entidade.

Uma vez elaborado o pedido de compra, a entidade externa “Setor de Compras” toma conhecimento dos itens que precisam ser adquiridos e retorna para o agente “Interface” a necessidade do miniprojeto, o qual será elaborado usando este agente, onde uma vez finalizado será encaminhado para o agente “Analista”.

Ao receber a mensagem do miniprojeto do agente “Interface”, o agente “Analista” analisará se o presente projeto tem relevância para o interesse público quanto a ótica da

oportunidade para aquisição dos itens solicitados e se é conveniente adquiri-los. Se aprovado o projeto, ele verificará se o mesmo poderá ser adquirido através da contratação direta ou através de um processo de licitação. Independente do resultado dessa análise, ele gerará um projeto detalhado baseado no miniprojeto o que permitirá que se adquiram os bens ou serviços solicitados [32][99].

Caso não seja possível adquirir os bens ou serviços através da contratação direta, o agente “Analista” enviará uma mensagem com o projeto para o agente “Licitador”, o qual fará a cotação dos preços através da entidade externa “WEB” [63]. Ao receber os três menores preços e efetuar a média entre eles, o agente “Licitador” definirá a modalidade e o tipo de licitação, elaborará o edital de licitação e o publicará, onde o agente encaminhará uma mensagem à entidade externa “Licitante” avisando-o dessa nova publicação.

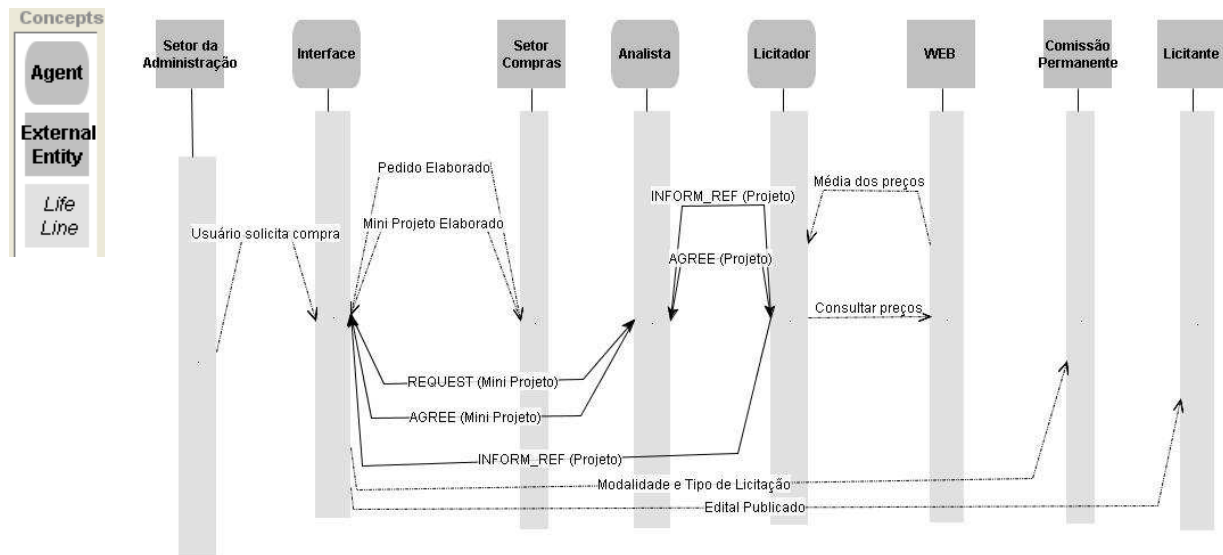


Figura 39 - Modelo das Interações entre Agentes do Sistema SMPLP

5.2.4 Modelagem dos mecanismos de cooperação e coordenação

O SMPLP possui uma arquitetura de duas camadas – Processamento do Usuário e Processamento de Licitação, as quais estão relacionadas diretamente com as quatro entidades externas que o sistema possui [10].

A camada level #2 é que tem a responsabilidade de atingir o objetivo específico Processar Licitação, composta pelo agente “Licitador”, a qual necessita dos serviços da camada abaixo, level #1, para que satisfaça seus objetivos. Por sua vez, a responsabilidade da camada de processamento de usuário é relacionada ao objetivo de compra de cada novo pedido inserido.

A Figura 40 ilustra o modelo dos mecanismos de cooperação e coordenação do SMPLP, que a camada de processamento do usuário interage com o setor da administração, o qual fornece o pedido de compra ao setor de compras responsável por elaborar o miniprojeto dos itens solicitados. Esta camada é composta do agente “Analista” e o “Interface”, os quais trabalharão em conjunto para realizar todo o processamento do usuário.

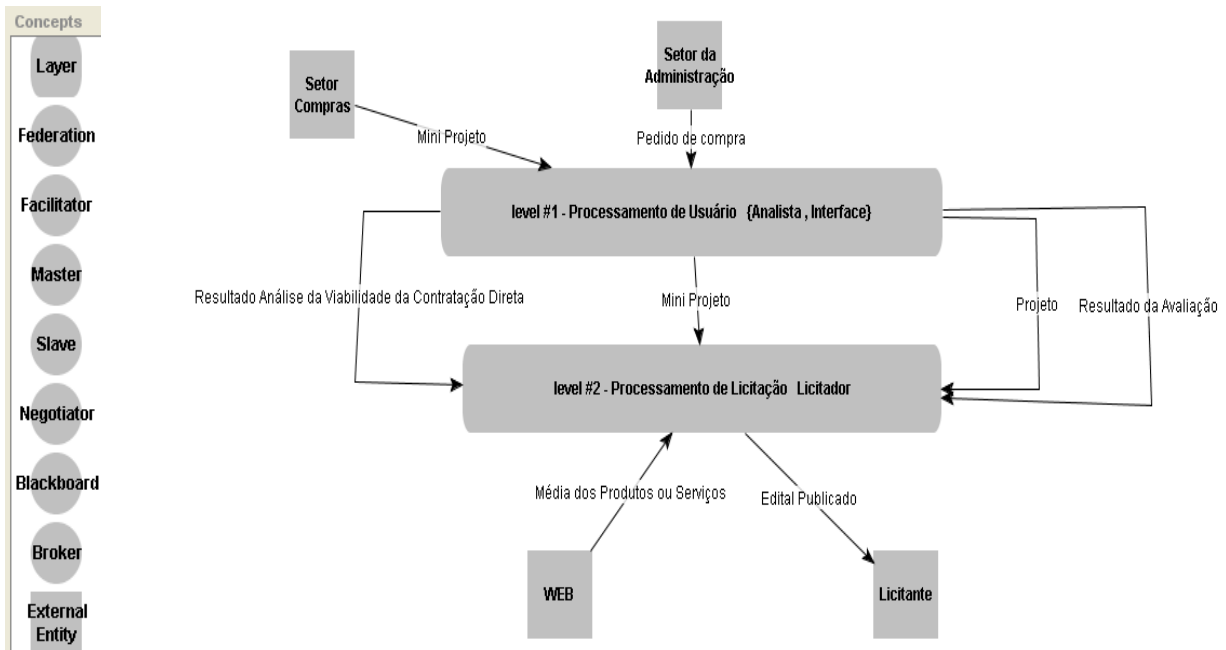


Figura 40 – Modelo dos Mecanismos de Cooperação e Coordenação do Sistema SMPLP

5.2.5 Modelagem do conhecimento do agente

O Modelo do Conhecimento do Agente representa o conhecimento particular de cada agente da sociedade. Ele é construído com base no Modelo de Interações entre Papéis, o qual provê o conhecimento compartilhado entre os papéis durante as suas interações, e no Modelo do Conhecimento da Sociedade Multiagente, do qual se obtém os conhecimentos específicos de cada agente (Figura 36). Para cada Agente do Modelo da Sociedade Multiagente é criado um modelo do conhecimento do agente, apresentado a partir da Figura 41 [10].

No Modelo do Conhecimento do agente “Interface” podemos visualizar claramente os conhecimentos que são derivados do Modelo da Sociedade Multiagente do agente “Interface”, como o conhecimento “Pedido de Compra” e o conhecimento “Miniprojeto” [10].

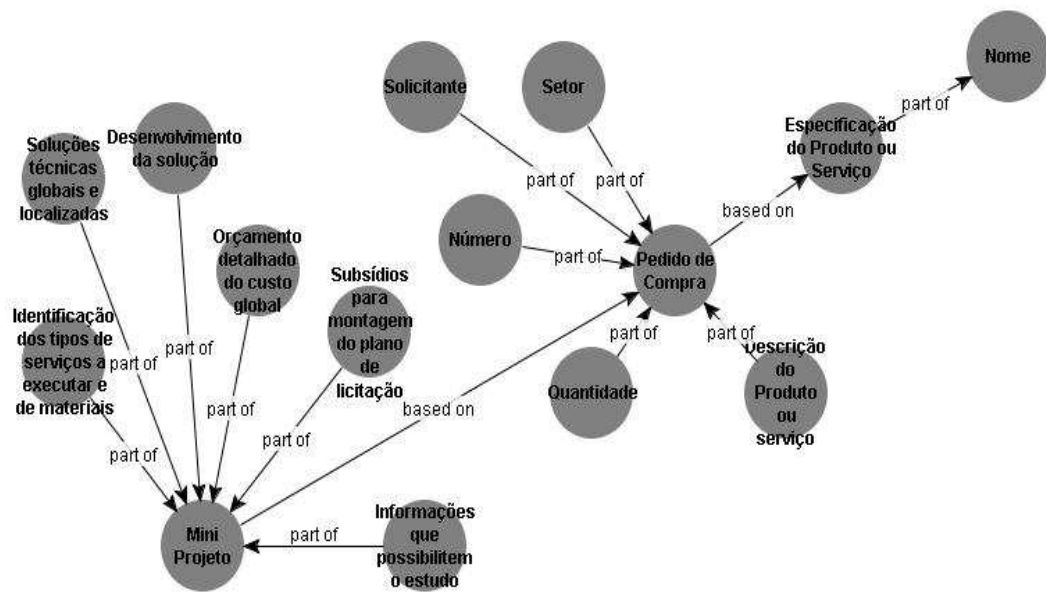


Figura 41 – Modelo do Conhecimento do agente “Interface”

Do conhecimento “Pedido de Compra”, observa-se os conhecimentos que o compõem: setor, solicitante, número, quantidade, descrição do produto ou serviço. Esse conhecimento é baseado na especificação inicial do produto ou serviço.

O conhecimento “Miniprojeto” é composto dos seguintes conhecimentos: identificação dos tipos de serviços a executar e de materiais, soluções técnicas globais e localizadores, desenvolvimento da solução, orçamento detalhado do custo global, subsídios para montagem do plano de licitação e informações que possibilitem o estudo.

No Modelo do Conhecimento do agente “Analista” podemos visualizar os conhecimentos que são derivados do Modelo da Sociedade Multiagente do agente “Analista”, como o conhecimento “Resultado da Avaliação”, “Resultado da Análise da Viabilidade da Contratação Direta” e o conhecimento “Projeto”, apresentado na Figura 42.

Os conhecimentos “Resultado da Avaliação”, “Resultado da Análise da Viabilidade” e o “Projeto” são baseados no conhecimento “Miniprojeto”. O conhecimento “Projeto” é composto pelos conhecimentos: nome, descrição, data de elaboração.

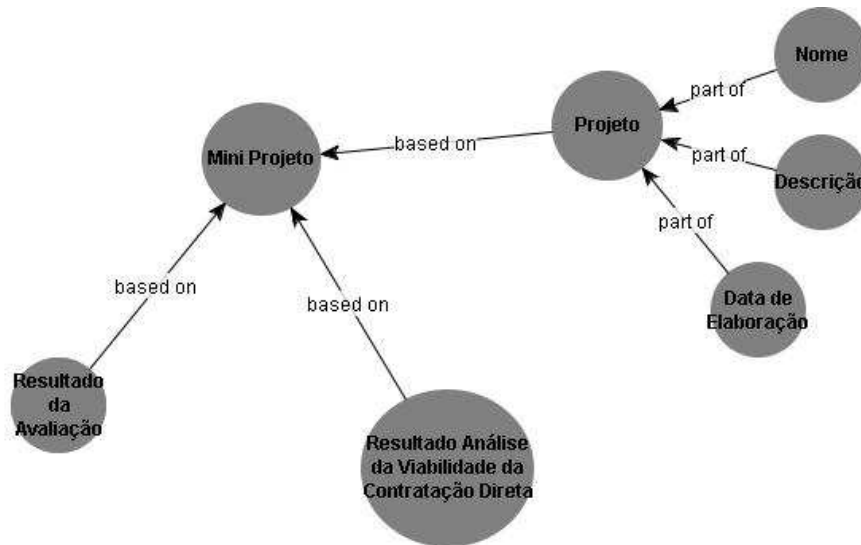


Figura 42 – Modelo do Conhecimento do agente “Analista”

No Modelo do Conhecimento do “agente Licitador” podemos visualizar os conhecimentos que são derivados do Modelo da Sociedade Multiagente do “agente Licitador”, como o conhecimento “Edital” e “Edital Publicado”, apresentado na Figura 43.

O conhecimento “Edital” é derivado do conhecimento “Projeto”, composto pelos seguintes conhecimentos: número/ordem, nome da repartição, modalidade, objeto da licitação, critério de aceitabilidade, condições de pagamento, regime de execução, tipo da licitação, data e hora para abertura dos envelopes, local e hora para recebimento da documentação e prazo entre a publicação e a realização da licitação.

Derivando do conhecimento “Edital”, o conhecimento “Edital Publicado” é composto do conhecimento data de publicação, e faz parte do conhecimento “Local de Publicação”. Este conhecimento é composto dos seguintes conhecimentos: repartição, jornal, diário oficial da união e diário oficial do estado.

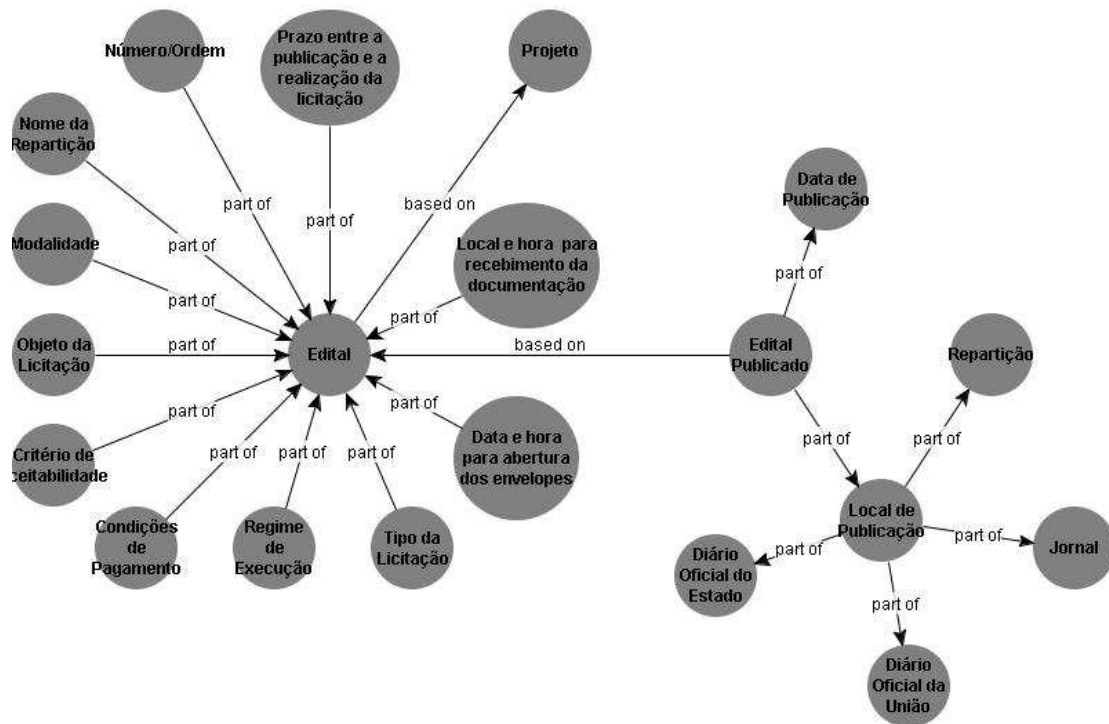


Figura 43 – Modelo do Conhecimento do agente “Licitador”

5.2.6 Modelagem das ações do agente

O Modelo de Ações do Agente representa as ações que o agente terá no seu ambiente de acordo com as responsabilidades de que tiver sido encarregado. Nesse modelo são representadas as percepções que o agente tem do ambiente: um evento ou as possíveis mensagens que ele poderá receber de outros agentes ou de entidades externas que irão disparar as suas ações. Existem duas formas de mapeamento dessas percepções para ações. A primeira é quando uma percepção é mapeada diretamente para uma ação. Nesse caso, o agente é considerado do tipo reativo. A outra forma é quando ocorre um planejamento na busca da ação adequada de acordo com a percepção do agente para atingir um objetivo. Nesse caso, o agente é considerado do tipo deliberativo e usa um motor de inferência no seu processo de raciocínio. Para realizar uma ação o agente pode necessitar de uma destreza (uma destreza é uma habilidade que o agente deverá possuir para a execução de suas responsabilidades) e também atender a uma determinada pré-condição. Ao término de uma ação, uma pós-condição também pode ser necessária. Aqui os pontos de variação dos agentes em relação a responsabilidades variantes são os mesmos definidos no Modelo da Sociedade Multiagente [10].

Um Modelo de Ações do Agente é construído para cada agente do Modelo da Sociedade Multiagente, o qual é usado como base para a criação desse modelo. Os conceitos

de entidade externa, destreza, conhecimento e condição (não mais relacionados com as responsabilidades, mas sim com as ações) são obtidos do Modelo da Sociedade Multiagente. Os conceitos de “mensagens” e “eventos” buscam-se no modelo de interações entre agentes. Os conceitos conhecimento e condição extraem-se também do Modelo da Sociedade Multiagente, que aqui não estarão mais relacionados com responsabilidades, mas sim com ações. Os demais relacionamentos serão informados pelo usuário [10].

Na Figura 44 é apresentado o Modelo das Ações do agente “Interface”, onde o agente percebe o evento “Especificação do Produto ou Serviço” que é gerado pela entidade externa “Setor da Administração”. Esse evento dispara automaticamente a ação “Obter Especificação do Pedido”, a qual só pode ser realizada mediante a destreza “Elaboração do Pedido”.

Ao executar essa ação, o agente produz o conhecimento “Pedido de Compra”, o qual é usado para executar a próxima ação, “Elaborar Miniprojeto”. Assim como a ação “Obter Especificação do Pedido” só pode ser realizada mediante uma destreza, da mesma forma a ação “Elaborar Miniprojeto” precisa da destreza “Elaboração e Submissão do Miniprojeto”.

Após executar essa ação, o agente produz agora um conhecimento “Miniprojeto”, o qual é usado para executar a próxima ação, “Submeter Miniprojeto”. Mais uma vez, essa nova ação pra ser executada precisa de uma destreza, isto é, a “Elaboração e Submissão do Miniprojeto”.

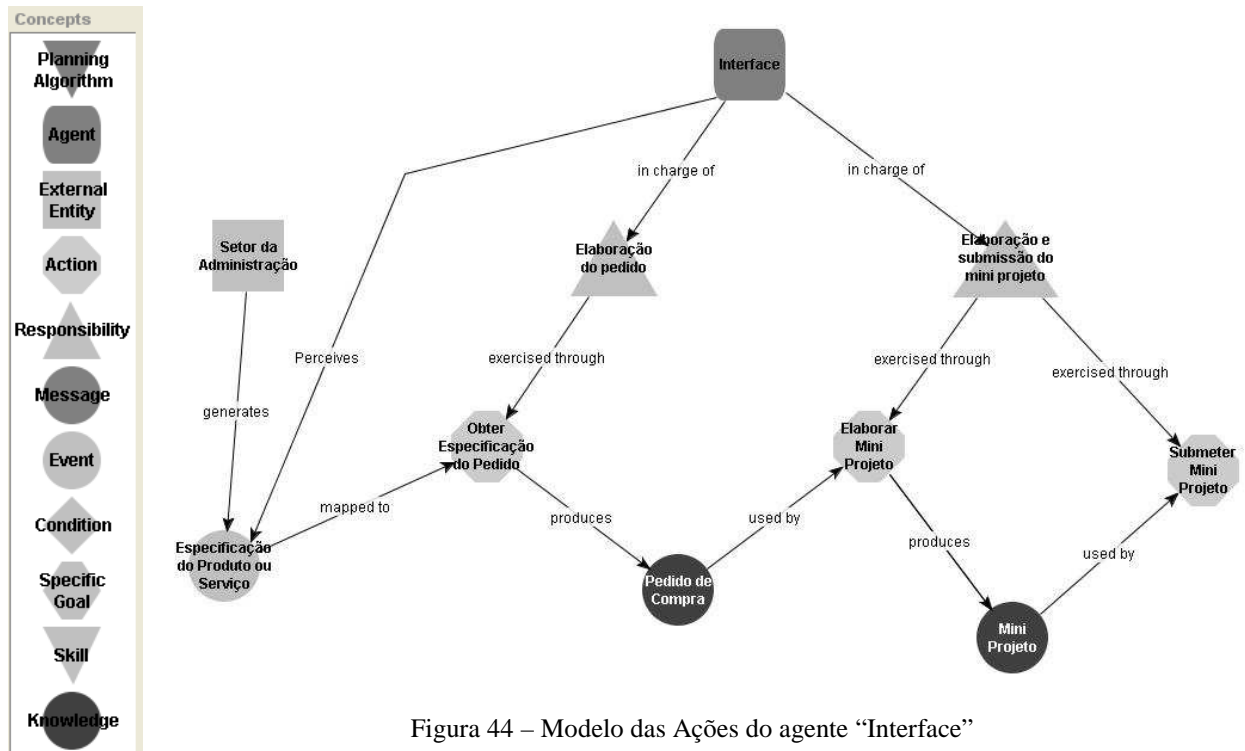


Figura 44 – Modelo das Ações do agente “Interface”

Na Figura 45 é apresentado o Modelo das Ações do agente “Analista”, onde o agente recebe o conhecimento “Miniprojeto” do “agente Interface”, o qual é usado para executar a ação “Consultar Produtos e Serviços”. Esta ação precisa do conhecimento “Produtos e Serviços Relevantes” que é gerado pela “entidade externa Ontol – Produtos e Serviços Relevantes”. Além disso, esta ação necessita da destreza “Análise da Relevância para o Interesse Público”.

Uma vez executada essa ação, ela produzirá o conhecimento “Resultado da Avaliação”, que juntamente com o conhecimento “Miniprojeto”, executará duas novas ações, “Verificar Dispensa” e “Verificar Inexigibilidade de Licitação”. Como as outras ações, essas também dependem de uma destreza anterior, nesse caso, a “Análise da Viabilidade da Contratação Direta”. Podemos observar através do modelo que a ação “Verificar Inexigibilidade de Licitação” para ser executada precisa do conhecimento “Produtos e Serviços” que é fornecido pela “entidade externa WEB” [63].

Essas duas últimas ações produzirão um novo conhecimento, “Resultado da Análise da Viabilidade da Contratação Direta”, o qual será usado para executar uma nova ação, “Gerar Projeto”. Mais uma vez, essa ação também precisa de uma destreza anterior, “Elaborar Projeto”. Ao ser executada, ela produzirá um novo conhecimento, “Projeto”.

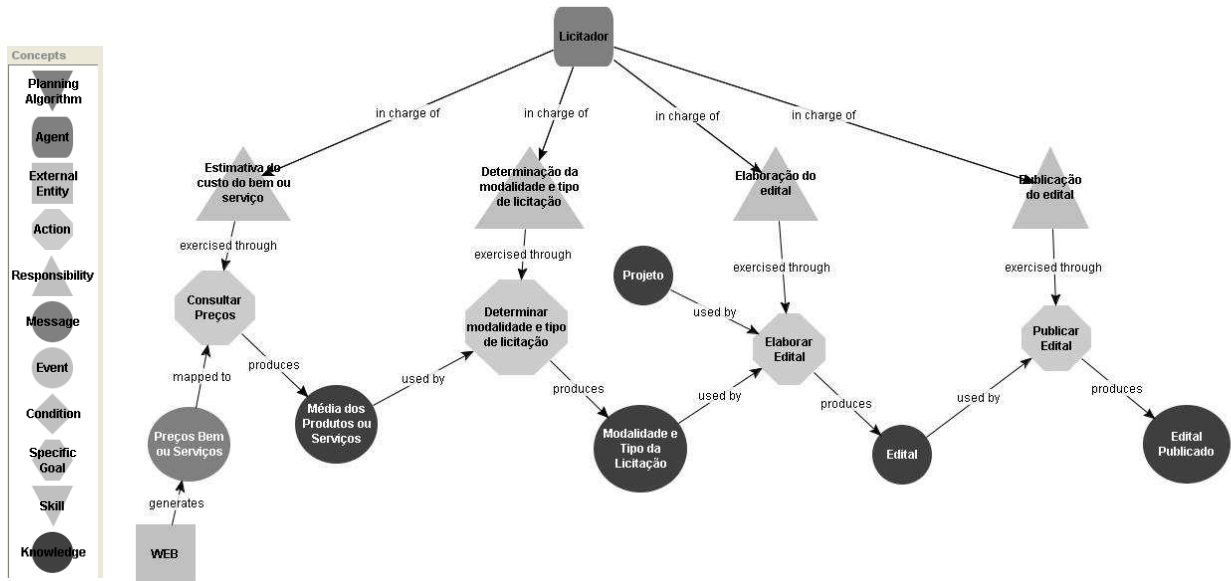


Figura 45 – Modelo das Ações do agente “Analista”

Na Figura 46 é apresentado o Modelo das Ações do agente “Licitador”, que é o principal agente da nossa modelagem, o qual recebe o conhecimento “Preços Bem ou Serviços” da “entidade externa WEB”, usado para executar a ação “Consultar Preços”. Esta ação necessita da destreza “Estimativa de Custo do Bem ou Serviço”.

Após a realização dessa ação, é produzido o conhecimento “Média dos Produtos ou Serviços”, o qual será usado para produzir uma nova ação, “Determinar Modalidade e Tipo de Licitação”. Obviamente que essa ação também precisa uma destreza para ser executada, isto é, a “Determinação da modalidade e tipo de licitação”.

Uma vez realizada essa ação, um novo conhecimento é produzido, isto é, “Modalidade e Tipo de Licitação”, que juntamente com o conhecimento “Projeto”, que foi produzido pelo agente “Analista”, servirá para a execução de uma nova ação, “Elaborar Edital”. Esta ação também necessita de uma destreza para ser executada, a “Elaboração do Edital”.

Esta última ação produzirá o conhecimento “Edital” que servirá para executar a última ação desse modelo, “Publicar Edital”. Da mesma forma que as ações anteriores, essa ação também necessita de uma destreza para ser executada, isto é, a “Elaboração do Edital”. Sendo executada essa última ação, um último conhecimento é produzido, o “Edital Publicado”.

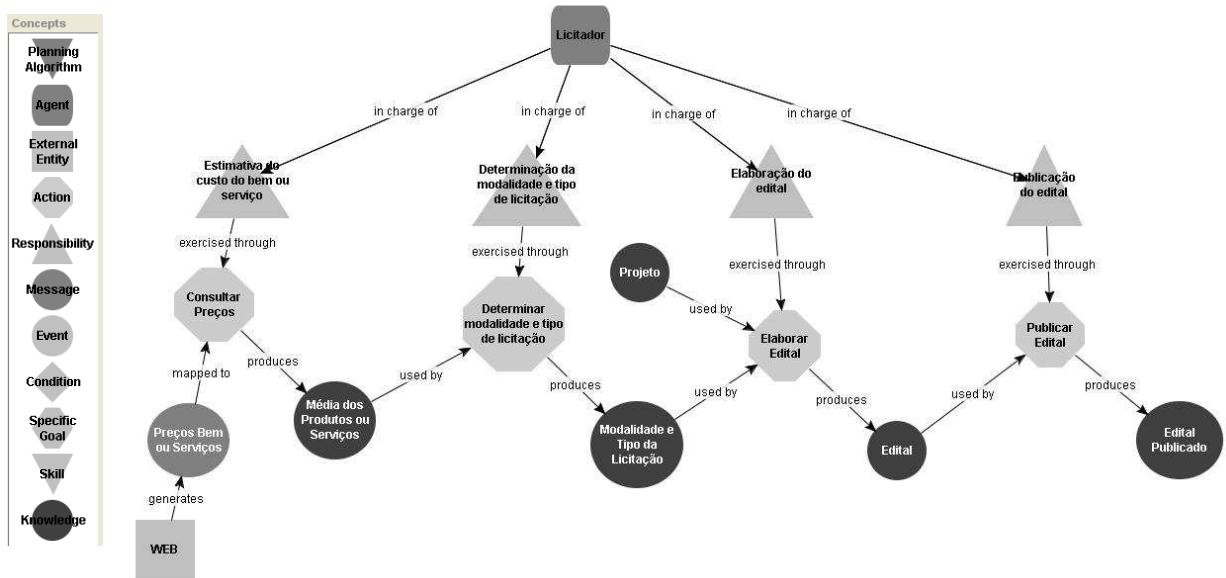


Figura 46 – Modelo das Ações do agente “Licitador”

5.3 Fase de Implementação da Aplicação

Na fase de implementação da Engenharia de Aplicações Multiagente, almeja-se produzir modelos cujos conceitos representados equivalham às principais classes utilizadas na plataforma JADE [48] para escrever uma aplicação multiagente, o que é feito por meio da linguagem orientada a objetos Java [50][51], a qual dá suporte àquela plataforma. Tal modelo, desse modo elaborado, poderia até ser submetido a um processo automático para geração de um esqueleto do código-fonte da aplicação, que deveria apenas ser complementado pelo programador [64].

Dessa maneira, a tarefa fundamental do desenvolvedor é mapear os conceitos presentes nos modelos do projeto da aplicação em termos de agentes, comportamentos e atos de comunicação, que são os conceitos envolvidos na tecnologia de implementação adotada [64].

O reuso, por conseguinte, situa-se não propriamente na elaboração dos modelos de implementação, que resultam do simples mapeamento de conceitos, mas sim na complementação do código gerado, para a qual o programador pode se valer de agentes de software anteriormente implementados, cuja seleção ocorre conforme a semelhança de comportamentos e comunicações entre si que apresentem, sendo a adaptação necessária em função dos detalhes que diferem uma aplicação de outra, as quais sempre procedem da composição das implementações de toda sua sociedade multiagente [64].

Nas subseções subsequentes, a descrição do Mapeamento de Agentes do Projeto à Implementação e de Responsabilidades em Comportamentos e do Mapeamento de Interações

entre Agentes em Atos de Comunicação é exemplificada com o Modelo de Implementação da Aplicação multiagente SMPLP [64].

5.3.1 Modelagem de comportamentos

A modelagem de comportamentos é representada na Figura 47, a qual é composta do conceito “JADE Agent”, que representa um agente JADE; “JADE Behaviour”, que representa um comportamento JADE e “Method JADE Behaviour”, que representa um método do comportamento JADE [48].

O agente “JADE Interface”, possui dois comportamentos, “Elaboração do Pedido” e a “Elaboração e Submissão do Miniprojeto”. O comportamento “Elaboração do Pedido”, possui apenas um método do comportamento JADE, “Obter Especificação do Pedido”. Já o comportamento “Elaboração e Submissão do Miniprojeto” possui dois métodos do comportamento JADE, o método “Elaborar Miniprojeto” e “Submeter Miniprojeto”.

O agente “JADE Analista”, possui três comportamentos, “Análise da Relevância para o Interesse Público”, “Análise da Viabilidade da Contratação Direta” e “Elaborar Projeto”. O comportamento “Análise da Relevância para o Interesse Público” possui apenas um comportamento, “Consultar Produtos e Serviços”. Já o comportamento “Análise da Viabilidade da Contratação Direta”, possui dois comportamentos, “Verificar Dispensa” e “Verificar Inexigibilidade de Licitação”. O último comportamento, “Elaborar Projeto”, possui apenas um método, “Gerar Projeto”.

No caso do agente “JADE Licitador”, o principal da sociedade MultiAgente, possui quatro comportamentos, “Estimativa do Custo do Bem ou Serviço”, “Determinação da Modalidade e Tipo de Licitação”, “Elaboração do Edital” e “Publicação do Edital”. Uma particularidade desses comportamentos é que todos possuem apenas um método. No caso do comportamento “Estimativa do Custo do Bem ou Serviço”, ele possui apenas um método, “Consultar Preços”. Já o comportamento “Determinação da Modalidade e Tipo de Licitação”, possui o método “Determinar Modalidade e Tipo de Licitação”. O comportamento “Elaboração do Edital” possui o método “Elaborar Edital”. E por último, o comportamento “Publicação do Edital” possui o método “Publicar Edital”.

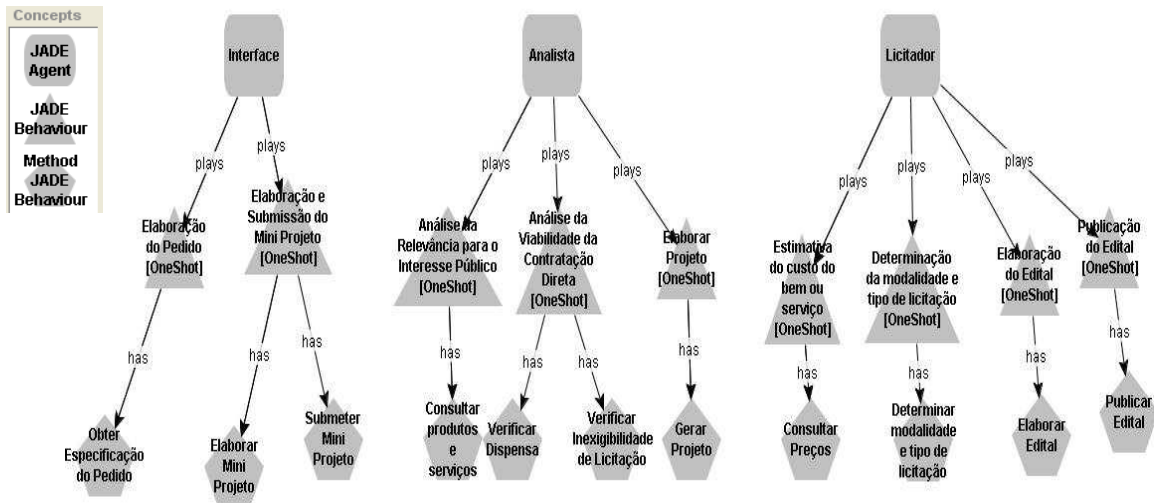


Figura 47 – Modelo de Comportamentos

5.3.2 Modelagem de atos de comunicação

A modelagem de atos de comunicação é representada na Figura 48, a qual é composta do conceito “JADE Agent”, que representa um agente JADE; “FIPA-ACL [27] Message” representa uma performativa presente na especificação da linguagem de comunicação entre agentes; “JADE Behaviour”, que representa um comportamento JADE e “Method JADE Behaviour”, que representa um método do comportamento JADE [48].

As mensagens trocadas entre agentes presentes no Modelo de Interações entre Agentes são mapeadas para a linguagem FIPA-ACL [27]. As interações identificadas no Modelo de Interações entre Agentes são mapeadas para uma linguagem de comunicação entre agentes [64].

Na modelagem de atos de comunicação, o agente “JADE Interface”, possui o comportamento “Elaboração e Submissão do Miniprojeto”. Esse comportamento possui um método do comportamento JADE, “Elaborar Miniprojeto”, o qual produzirá a mensagem “Miniprojeto” que será enviada para o agente “JADE Analista”.

O agente “JADE Analista” recebe a mensagem “Miniprojeto” via método “Consultar Produtos e Serviços”. Esse método faz parte do comportamento “Análise da Relevância para o Interesse Público”. Esse agente possui também o comportamento “Elaborar Projeto”, o qual possui o método “Gerar Projeto” que produzirá a mensagem “Projeto” para o agente “JADE Licitador”.

O agente “JADE Licitador”, recebe essa mensagem através do método “Consultar Preços”, o qual faz parte do comportamento “Estimativa do Custo do Bem ou Serviço”.

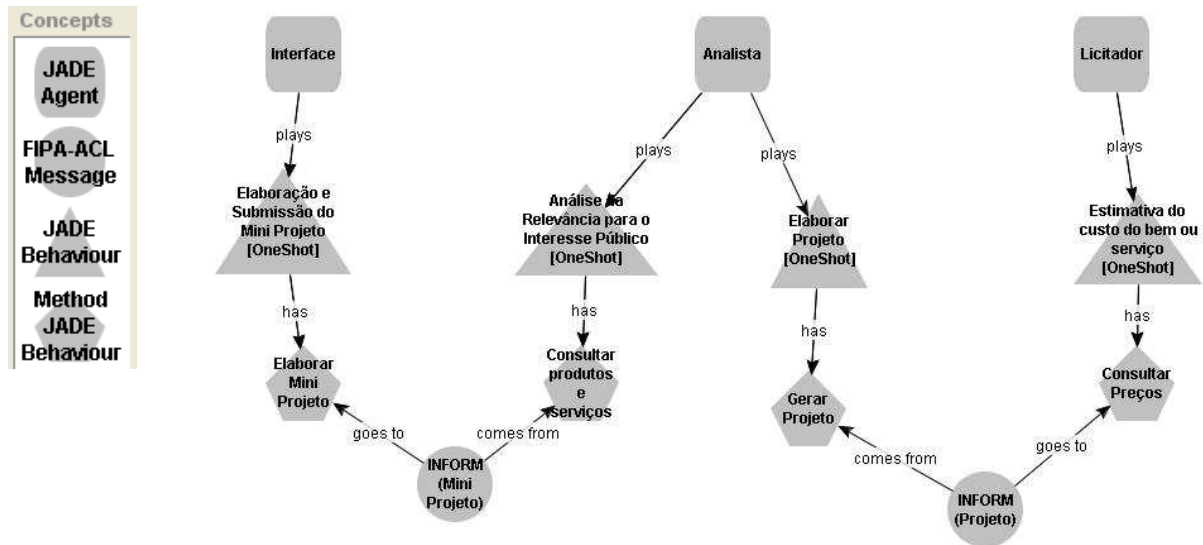


Figura 48 – Modelo de Atos de Comunicação

5.3.3 Implementação do SMPLP no JADE

A plataforma de implementação adotada para o SMPLP foi o ambiente Java JADE (Java Agent DEvelopment Framework), utilizando como motor de inferência o PROLOG.

Ao ser iniciado o ciclo de vida do agente “Interface”, ele passa a monitorar constantemente o ambiente, percebendo quando que um novo pedido é inserido no SMPLP. A Figura 49 apresenta parcialmente como é feito esse monitoramento. Para uma verificação completa, consulte o Apêndice A.

```

Class.forName("com.mysql.jdbc.Driver");
Connection con =
DriverManager.getConnection("jdbc:mysql://localhost/proc_licitatorio","root","123");
ResultSet rs = stmt.executeQuery();
stmt = con.prepareStatement("select count(*) as quant from pedidos");
rs = stmt.executeQuery();
if (rs.next()) {
    quantPed = rs.getInt("quant");
}

```

Figura 49 – Parte do código do agente “Interface”

Na Figura 50 é apresentado o ponto inicial da troca de mensagens que será realizada entre os agentes, apresentado na Figura 39 no Modelo de Interações entre Agentes. Nesse momento, apenas o agente “Interface” monitora o ambiente aguardando um novo pedido, razão pela qual não existe nenhuma troca de mensagem.

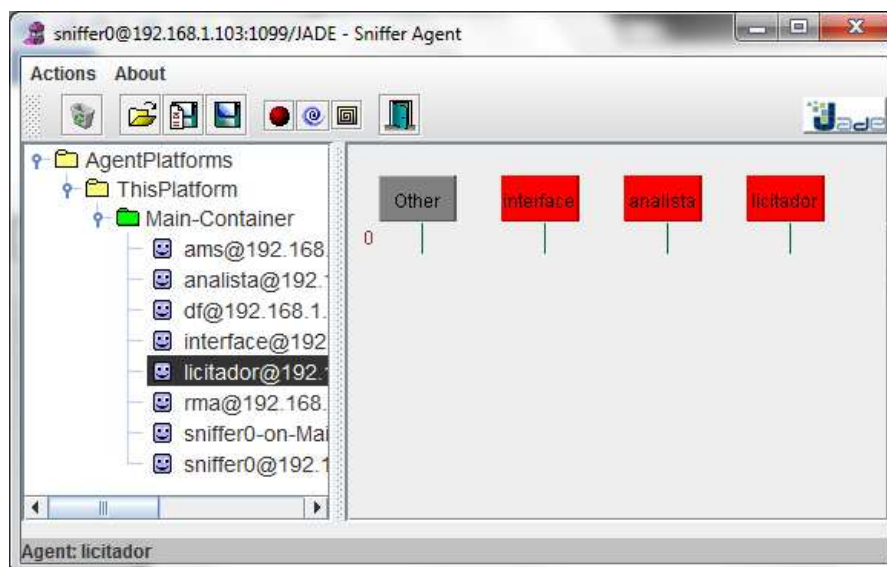


Figura 50 – Comunicação Entre Agentes

O agente “Interface” possui o conhecimento da quantidade de pedidos existentes na base. Quando ele percebe que um novo pedido foi inserido, ele dispara uma mensagem para a entidade externa “Setor de Compras”, informando que a mesma deve acessar o sistema e criar um miniprojeto para o pedido inserido, apresentado na Figura 51.

```

stmt = con.prepareStatement("select count(*) as quant from pedidos");
rs = stmt.executeQuery();
if (rs.next()) {
    novoPed = rs.getInt("quant");
}
rs.close();
stmt.close();
if (novoPed > quantPed) {
    stmt = con.prepareStatement("select numero from pedidos order by numero desc");
    rs = stmt.executeQuery();
    System.out.println(2);
    rs.next();
    int codpedido = rs.getInt("numero");
    System.out.println("Agente Interface: setor de compras, logue no sistema e insira um
projeto para o pedido " + codpedido);
    rs.close();
    stmt.close();
    quantPed++;
}

```

Figura 51 – Parte do código do agente “Interface”

Quando a entidade externa “Setor de Compras” cria o miniprojeto, o agente “Interface”, dispara uma mensagem para o agente “Analista” (Figura 52), para que o mesmo possa analisar o projeto, dando suporte à entidade externa “Comissão Permanente”.

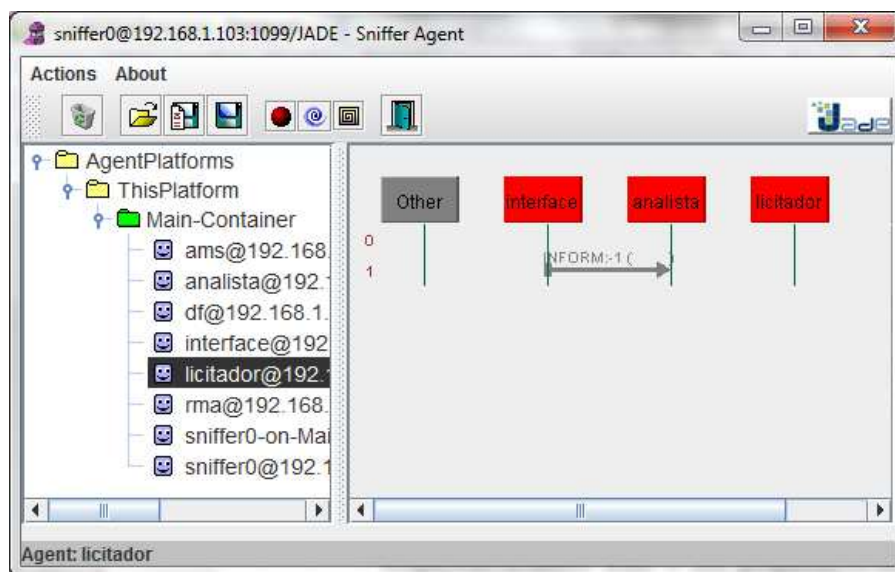


Figura 52 – Agente “Interface” transmitindo uma mensagem ao agente “Analista”

A Figura 53 apresenta a troca de mensagens entre os agentes “Interface”, “Analista” e “Licitador”. Quando a entidade externa “Comissão Permanente” analisa e aprova o miniprojeto, o agente “Analista” percebe o ocorrido e imediatamente informa para o agente “Licitador”, para que o mesmo possa seguir com o processo licitatório. Além disso, o agente “Analista” confirma o recebimento do miniprojeto para o agente “Interface”. Para uma verificação completa, consulte o Apêndice A e B.

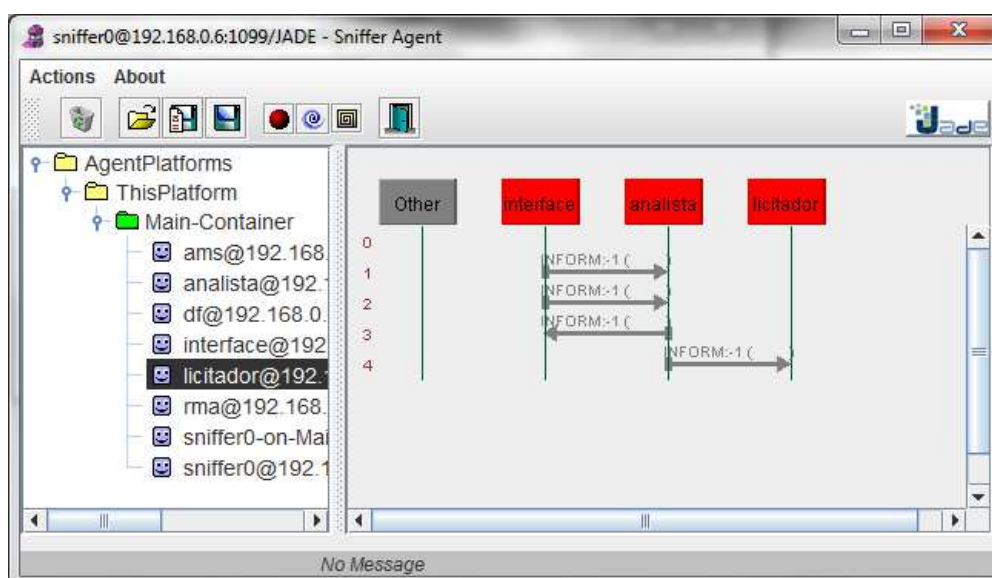


Figura 53 – Troca de mensagens entre o agente “Interface”, “Analista” e “Licitador”

A Figura 54 apresenta parcialmente as ações do agente “Licitador”, principal agente do SMPLP, o qual fará a cotação de preços na entidade externa “WEB” para obter o valor médio dos itens a serem licitados. Ao realizar a cotação, ele utilizará a base de

conhecimento criada no Prolog (Figura 55), para sugerir a modalidade e o tipo de licitação mais apropriada para o processo de licitação. Para uma verificação completa do agente “Licitador”, consulte o Apêndice C.

```

ACLMessage msg = receive();
if (msg != null)
System.out.println("Agente Licitador: Recebido projeto numero " + msg.getContent());
String consulta = "consult('C:/Documents and Settings/FabioVieira/Meus
documentos/NetBeansProjects/Proc_Licitatorio/src/prolog/regras_licitacao.pl)";
Query q1 = new Query(consulta);
q1.hasSolution();
Query obraEngenharia = new Query("assert(obraEngenharia("+engenharia+"))");
obraEngenharia.hasSolution();
Query obraValor = new Query("assert(obraValor("+valorMedio+"))");
obraValor.hasSolution();
Query cbLicitacao_pl = new Query("cbLicitacao(X)");
Query cbTipo_pl = new Query("cbTipo(X)");
System.out.println(Integer.parseInt(cbLicitacao_pl.oneSolution().get("X").toString()));
System.out.println(Integer.parseInt(cbTipo_pl.oneSolution().get("X").toString()));
block();

```

Figura 54 – Parte do código do agente “Licitador”

```

%obraEngenharia('N').
%obraValor(2000000).

modalidade(W) :- obraEngenharia(Y),(Y,'S'),obraValor(Z),=<(Z,150000),W = 'Convite',!.
modalidade(W) :- obraEngenharia(Y),(Y,'S'),obraValor(Z),>(Z,150000),=<(Z,150000),W =
'Tomada de preço',!.
modalidade(W) :- obraEngenharia(Y),(Y,'S'),obraValor(Z),>(Z,150000),W = 'Concorrência',!.

modalidade(W) :- obraEngenharia(Y),(Y,'N'),obraValor(Z),=<(Z,80000),W = 'Convite',!.
modalidade(W) :- obraEngenharia(Y),(Y,'N'),obraValor(Z),>(Z,80000),=<(Z,650000),W =
'Tomada de preço',!.
modalidade(W) :- obraEngenharia(Y),(Y,'N'),obraValor(Z),>(Z,650000),W = 'Concorrência',!.

tipoLicitacao(X) :- obraEngenharia(Y),(Y,'S'),X = 'Técnica e Preço',!.
tipoLicitacao(X) :- obraEngenharia(Y),(Y,'N'),X = 'Menor Preço',!.

cbLicitacao(X) :- modalidade(W),(W,'Convite'),X = 3,!.
cbLicitacao(X) :- modalidade(W),(W,'Tomada de preço'),X = 2,!.
cbLicitacao(X) :- modalidade(W),(W,'Concorrência'),X = 1,!.

cbTipo(X) :- tipoLicitacao(W),(W,'Técnica e Preço'),X = 3,!.
cbTipo(X) :- tipoLicitacao(W),(W,'Menor Preço'),X = 1,!.

```

Figura 55 – Base de conhecimento no Prolog

5.4 Considerações Finais

Neste capítulo apresentou-se inicialmente a fase de análise da Engenharia de Aplicações Multiagente, que consiste em elaborar a especificação dos requisitos de uma determinada aplicação, partindo de modelos de domínio, que são artefatos de software reutilizáveis elaborados na fase correspondente da Engenharia de Domínio Multiagente.

Em seguida apresentou-se a fase de projeto da Engenharia de Aplicações Multiagente consiste de três subfases, que são: o Projeto da Arquitetura, que define a visão global e externa da sociedade multiagente, sobretudo, passando pela escolha de seus mecanismos de cooperação e coordenação; o Projeto do Agente, que estabelece a visão detalhada e interna de cada agente, modelando sua estrutura e seu comportamento; e a Modelagem do Conhecimento da Sociedade Multiagente, que identifica os conceitos compartilhados por todos os agentes em sua comunicação.

Por fim, foi realizada a implementação dos agentes “Interface”, “Analista” e “Licitação” através do JADE, um framework para o desenvolvimento de agentes, devido ao fato do mesmo ser a plataforma de desenvolvimento utilizado pelo MADAE-Pro, um processo para o desenvolvimento e reuso de famílias de sistemas multiagente desenvolvido pelo grupo GESEC. Além disso, desenvolveram-se as regras em PROLOG para a escolha da modalidade e tipo de licitação, conforme descrito na seção 3.2.11.

6 CONCLUSÕES

Este trabalho apresentou um Sistema Multiagente para apoiar o Processo de Licitação Pública, que tem como principal objetivo auxiliar o agente público nos momentos de tomada de decisão do processo licitatório. Foram também analisados conceitos fundamentais do processo de licitação pública e outros sistemas multiagente para a licitação pública.

O SMPLP foi desenvolvido seguindo as diretrizes do MADAE-Pro, um processo para o desenvolvimento de aplicações e de famílias de sistemas multiagente. A modelagem do sistema foi realizada utilizando a ONTORMAS, uma ferramenta dirigida por ontologias que guia a especificação dos requisitos e o projeto de sistemas multiagente.

6.1 Resultados e Contribuições da Pesquisa

As principais contribuições desta pesquisa foram:

- Realização de análise comparativa de sistema multiagente no domínio da Licitação Pública. Para tanto, foram definidos três critérios comparativos: Linguagem e processo de Modelagem, tais como, UML, MADAE-ML e AUML; Paradigma de desenvolvimento, tal como, baseado em agentes; Modalidades de Licitação, tais como, Convite, Pregão, Tomada de Preço e Concorrência. Os sistemas multiagente analisados foram: Cnet, INFOLICT, SMA Pregoeiro e SMPLP.
- Compilação de documentos no domínio da licitação pública para uma linguagem adequada para desenvolvedores de software. O conhecimento no domínio da licitação pública encontra-se documentado em linguagem jurídica, o que dificulta o processo de desenvolvimento de software.
- Elicitação dos requisitos gerais no domínio da Licitação Pública. Baseado na Lei n° 8666/93, descreveram-se detalhadamente todas as fases internas e externas do processo licitatório que devem ser satisfeitos por quaisquer sistemas desenvolvidos nesse domínio.
- Modelagem do sistema SMPLP utilizando a ferramenta ONTORMAS aplicada no domínio da licitação pública, o qual permite o reuso para futuros trabalhos nesse domínio;

- Avaliação experimental do processo para o desenvolvimento de sistemas multiagente, MADAE-Pro. MADAE - Pro foi desenvolvido pelo Grupo de Pesquisa em Engenharia de Software e Engenharia do Conhecimento – GESEC. Esse processo tem sido avaliado experimentalmente no desenvolvimento de outros sistemas multiagente em subáreas do domínio jurídico, como no Direito de Família. De maneira geral, o processo se mostrou adequado ao desenvolvimento do sistema SMPLP, uma vez que diminuiu a distância cognitiva entre os requisitos iniciais e o sistema final.
- Evidenciou-se a necessidade da aplicação do paradigma baseado em agentes no desenvolvimento de software no domínio da Licitação Pública. O paradigma baseado em agentes proporciona um nível de abstração superior ao paradigma orientado a objetos, o que é adequado para lidar com ambientes dinâmicos e complexos, como a licitação pública.

Como principais resultados, temos:

- Um protótipo do Sistema SMPLP, o qual guia todas as etapas do processo licitatório facilitando a tomada de decisões do agente público, tais como: aprovar o processo licitatório ou dispensá-lo; definir a modalidade e o tipo de licitação. Isto se dá devido a efetividade proporcionada pelos agentes no processo de tomada de decisão, tornando o processo licitatório mais padronizado e transparente;
- Base de conhecimento em Prolog disponibilizada em arquivo externo, contendo as regras para as definições das modalidades e tipos de licitação. O fato de o conhecimento estar armazenado externamente ao sistema permite sua alteração em tempo de execução;
- Submissão do artigo intitulado “Um Estudo de Caso na Aplicação do Processo Madae-Pro na Modelagem dos Requisitos de um Sistema Multiagente de Licitações Públicas” a Revista do Centro de Ciências da Economia e Informática (URCAMP).

6.2 Trabalhos Futuros

Pela sua complexidade, o escopo desta pesquisa foi limitado em vários aspectos que deverão ser abordados em futuros trabalhos, entre eles:

- O SMPLP suporta as modalidades e tipos de licitações comuns normatizadas pela lei nº 8666/93, entretanto, ele ainda não contempla o pregão eletrônico, concurso e o leilão. Para as versões futuras, as modalidades de licitações ainda não contempladas, serão incluídas.
- Efetuar uma revisão dos modelos da Fase de Projeto de Aplicação do MADAE-Pro, correspondente ao projeto do agente (modelo de ações do agente e modelo do conhecimento do agente), quanto a escolha do tipo do agente (reativo ou deliberativo) estabelecendo critérios específicos e claros;
- Aprimorar a ferramenta ONTORMAS quanto a modelagem, pois a mesma possui algumas limitações quanto a alteração do modelo já criado, fazendo com que o mesmo perca toda a configuração inicial criada. Logo, sugerimos a correção da mesma ou a construção de uma outra ferramenta para a construção dos modelos;
- Efetuar uma revisão e expansão da linguagem MADAE-ML, devido ao excesso de figuras que se repetem em seus vários modelos;
- Disponibilizar uma nova versão do MADAE-Pro que ofereça apoio a outras plataformas de desenvolvimento de agentes tais como o JADEX, Chimera e JACK a outras linguagens de comunicação de agentes como a KQML.

REFERÊNCIAS

- [1] ALMEIDA, A.T. RAMOS, F.S. Gestão da Informação na Competitividade das Organizações. Editora Universitária da UFPE. Recife. 2002.
- [2] ALONSO, F.; ANTONIO, A.; GONZÁLEZ, A. L.; FUERTES, J. L.; MARTINEZ, L. Towards a Unified Methodology for Software Engineering and Knowledge Engineering. Proceedings of the IEEE International Conference on Systems, Man, and Cybernetics, 1998, p. 4890-4895.
- [3] ANGELE, J.; FENSEL, D.; LANDES, D.; STUDER, R. Developing Knowledge-based Systems with MIKE. Springer Netherlands, ISSN 0928-8910, Vol. 5, Number 4, October, 1998, p. 389-418.
- [4] ANSI / IEEE Std 100-1984, IEEE Standard Dictionary of Electrical and Electronics Terms.
- [5] BARR, A.; FEIGENBAUM, E. The Handbook of Artificial Intelligence. Los Altos, California: William Kaufmann Inc., 1981. v.I-II.
- [6] BERNSTEIN, P. L. Desafio aos Deuses: a fascinante história do risco. Rio de Janeiro: Campus, 1997.
- [7] BIDGOLI, H. Decision Support System - Principles and Practice. West Publishing Company. New York. 1989.
- [8] BUCHANAN, B.G., and E.H. SHORTLIFFE. Rule-Based Expert Systems: The MYCIN Experiments of the Stanford Heuristic Programming Project. Addison-Wesley, Reading, MA, 1984.
- [9] CAVALCANTE, U. A. de S. MADAE-IDE: Um ambiente de desenvolvimento de software baseado no conhecimento para o reuso composicional no desenvolvimento de sistemas multiagente. Dissertação (Mestrado em Engenharia de Eletricidade – Área de Ciência da Computação) - Universidade Federal do Maranhão, São Luís, 2009.
- [10] CAVALCANTE, U. GIRARDI, R. An Overview of the MADAE-IDE Multi-Agent System Development Environment. Proceedings of the 7th International Conference on Information Technology: New Generations. Las Vegas. IEEE, 2010. pp. 968-973.
- [11] CHANDRASEKARAN, B. J. (1999). What Are Ontologies, and Why Do We Need Them? IEEE Intelligent Systems, 14 (1) , pp. 20–26.
- [12] CHIMERA. Chimera Programming Guide and Tech Ref. Disponível em: <http://www.lpa.co.uk/dow_doc.htm>. Acessado em: 10 dez. 2012.
- [13] CONSTITUIÇÃO DA REPÚBLICA FEDERATIVA DO BRASIL DE 1988. Presidência da República. Casa Civil. Subchefia para Assuntos Jurídicos. Disponível em: <http://www.planalto.gov.br/ccivil_03/constituicao/constituicao.htm>. Acesso em: 12 set. 2012.
- [14] CORREA, P. J. M. G.; MARQUES, D. B.; FARIAS, O. da S.; CAVALCANTE, G. H. R.; MARQUES, A. L. B.; LABIDI, S.; BARROS, A. K. D. Sistema Multiagente de Apoio à Tomada de Decisão para Controle da Qualidade de Combustíveis. In: 5

- PDPETRO - Congresso Brasileiro de Pesquisa e Desenvolvimento em Petróleo e Gás, 2009, Fortaleza-CE. 5 Congresso Brasileiro de Pesquisa e Desenvolvimento em Petróleo e Gás, 2009.
- [15] CORREIA, J. dos S. S. Um Processo Para a Aquisição de Relações Taxonômicas de uma Ontologia. Dissertação (Mestrado em Engenharia de Eletricidade – Área de Ciência da Computação) - Universidade Federal do Maranhão, São Luís, 2011.
- [16] COSTA, E.B.; LOPES, M.A.; FERNEDA, E. “MATHEMA: A Learning Environment Based on a Multi-Agent Architecture”, Proceedings of the 12th Brazilian Symposium on Artificial Intelligence, Campinas, Brazil, pp. 141-150, Wainer J.; Carvalho A. (eds), Volume 991 of Lecture Notes in Artificial Intelligence, Springer-Verlag, outubro 1995.
- [17] COSTA, S. S. S.; RABELO, R. J. Supporting the Creation of Virtual Enterprises using Mobile Agents. Anais of PRO-VE'2002. 3rd. IFIP Working Conference on Infrastructures for Virtual Enterprises. Lisboa, Portugal, 2002.
- [18] CRUCIOL, L. L. B. Modelagem de Apoio à Decisão para o Problema de Espera no Ar Utilizando Sistemas Multiagente e Aprendizagem por Reforço. Dissertação (Mestrado em Informática) - Universidade de Brasília, Brasília, 2012.
- [19] DE GEUS, A. P. A Empresa Viva. Rio de Janeiro: Editora Campus, 1998.
- [20] DECRETO Nº 5.450, DE 31 DE MAIO DE 2005. Presidência da República. Casa Cível. Subchefia para Assuntos Jurídicos. Disponível em: <http://www.planalto.gov.br/ccivil_03/_ato2004-2006/2005/decreto/d5450.htm>. Acesso em: 01 mar. 2012.
- [21] DIAS, M.M.; PACHECO, R. C. S. Uma visão geral de metodologias para desenvolvimento de sistemas baseados em conhecimento. Revista de Ciência da Informação – v. 10 nº 5. 2009.
- [22] DIVERIO, T.; MENEZES, P. Teoria da Computação: máquinas universais e computabilidade. 2.ed. Porto Alegre: Sagra-Luzzatto, 2000. (Livros Didáticos, v.5).
- [23] DURANS, C. L. T., Um Sistema de Recomendações de Licitações Públicas. Monografia (Graduação em Ciência da Computação) - Universidade Federal do Maranhão, São Luís, 2008
- [24] EMENDA CONSTITUCIONAL Nº 19, DE 04 DE JUNHO DE 1998. Presidência da República. Casa Civil. Subchefia para Assuntos Jurídicos. Disponível em: <http://www.planalto.gov.br/ccivil_03/constituicao/Emendas/Emc/emc19.htm>. Acesso em: 12 set. 2012.
- [25] FIGUEIREDO, J.R. Licitações Públicas para Principiantes. Florianópolis: Editora Insular, 2002.
- [26] FININ, T., LABROU, Y., and MAYFIELD, J. KQML as an agent communication language, Computer Science Department. University of Maryland Baltimore County. Baltimore, USA, 1993.
- [27] FIPA. ACL Message Structure Specification. Disponível em: <<http://www.fipa.org/specs/fipa00061/SC00061G.html>>. Acesso em: 01 mar. 2012.

- [28] FIPA. Foundation for Intelligent Physical Agents. Disponível em: <<http://www.fipa.org/>>. Acessado em: 25 jul. 2012.
- [29] FLORES, C. D. Fundamentos dos Sistemas Especialistas. In: BARONE, D. A. C. (Ed.). *Sociedades Artificiais: a nova fronteira da inteligência nas máquinas*. Porto Alegre: Bookman, 2003. p.332.
- [30] FONTES, L. M. O.; MENDES NETO, F. M.; PONTES, A. A. A. ; Campos, G. A. L. de. An Agent-Based Architecture for Supporting the Workgroups Creation and the Detection of Out-of-Context Conversation on Problem-Based Learning in Virtual Learning Environments. In: ACM Symposium on Applied Computing, Track on Intelligent, Interactive and Innovative Learning Environments, 2011, TaiChung, Taiwan. Proceedings of the 2011 ACM Symposium on Applied Computing. New York: ACM, 2011.
- [31] FRANÇA, M. A. de C. *Comentário à Lei de Licitações e Contratos da Administração Pública*. 6ª ed. Editora Saraiva, 2010.
- [32] FRANKLIN, S.; GRAESSER, A. “Is it an Agent, or just a Program?: A Taxonomy for Autonomous Agents”, Proceedings of the Third International Workshop on Agent Theories, Architectures, and Languages, Springer-Verlag, 1996.
- [33] GABINESKI, R. ; LORENZI, F. Sistema Multiagente Baseado em Casos de Apoio a Gerencia de Projetos. In: VIII Salão de Iniciação Científica e Trabalhos Acadêmicos, 2007, Guaíba. VIII Salão de Iniciação Científica e Trabalhos Acadêmicos. Guaíba: Ulbra, 2007.
- [34] GIGERENZER G.; TODD P. *Simple Heuristics That Make Us Smart*. Oxford Univ: Press, 1999.
- [35] GIRARD, J.; GAUTHIER, G.; LEVESQUE, S. Une architecture multiagent. Proceedings of the International Conference on Intelligent Tutoring Systems - ITS'92, Montreal (Canadá), junho, 1992.
- [36] GIRARDI, R. Engenharia de Software baseada em Agentes, Anais do IV Congresso Brasileiro de Ciência da Computação (CBCOMP 2004), Ed. UNIVALI, Itajaí, Santa Catarina, Brasil, pp. 913-937, 2004.
- [37] GIRARDI, R. Guiding Ontology Learning and Population by Knowledge System Goals. International Conference on Knowledge Engineering and Ontology Development. Valencia. 2010.
- [38] GIRARDI, R., DRUMOND, L. (2008). Multiagent Legal Recommender System. *Artificial Intelligence and Law (Dordrecht)*, March, pp. 175-207.
- [39] GIRARDI, R., FARIA, C., MARINHO, L. Ontology-based Domain Modeling of MultiAgent Systems. Proceedings of the Third International Workshop on Agent-Oriented Methodologies at International Conference on Object-Oriented Programming, Systems, Languages and Applications (OOPSLA 2004), Ed. Cesar Gonzalez-Perez, pp. 51--62. Vancouver, Canada. October 24th to 28th, 2004.
- [40] GIRARDI, R., LEITE. A Knowledge-based Tool for Multi-Agent Domain

- Engineering, Knowledge-Based Systems Journal, v. 21, Ed. Elsevier, pp. 604-611, 2008.
- [41] GIRARDI, R., MARINHO, L., & RIBEIRO, I. (2005). A System of Agent-based Patterns for User Modelling based on Usage Mining. *Interacting with Computers*, v. 17, n. 5, pp. 567-591.
- [42] GODET, M. *Manual de Prospectiva Estratégica. Da antecipação à ação*. Lisboa: Publicações Dom Quixote, 1993.
- [43] GOMES, L.F.A.M.; GOMES, C.F.S. ALMEIDA, A.T. *Tomada de Decisão Gerencial: Enfoque Multicritério*. Editora Atlas. São Paulo. 2002
- [44] GRUBER, T. Toward principles for the design of ontologies used for knowledge sharing. *International Journal of Human-Computer Studies*, p. 907-928, 1995.
- [45] HAPPEL, H. SEEDOF, S. Applications of Ontologies in Software Engineering, 2nd Int. Workshop on Semantic Web Enabled Software Engineering (SWESE 2006).
- [46] HOWDEN, N., RÖNNQUIST, R., HODGSON, A., LUCAS, A. JACK intelligent agents - Summary of an agent infrastructure. *Proceedings of the 5th International Conference on Autonomous Agents*, 2001.
- [47] IGLESIAS, C. A.; GARIJO, M.; GONZÁLEZ, J. C.; VELASCO, J. R. Analysis and Design of Multiagent Systems using MAS-CommonKADS. *Lecture Notes in Computer Science*, Springer Berlin/Heidelberg, ISSN 0302-9743, Vol. 1365, 1998, p. 313-327.
- [48] JADE. Java Agent Development Framework. Disponível em: <<http://jade.tilab.com/>>. Acesso em: 10 abr. 2012.
- [49] Jansen, M. & Girardi, R. (2006). GENMADEM: A Methodology for Generative Multiagent Domain Engineering. In: *Proceedings of the 9th International Conference on Software Reuse*, Torino. *Lecture Notes in Computer Science (LNCS)*, v. 4039. Berlin: Springer-Verlag, pp. 399-402.
- [50] JAVA SOFTWARE. Disponível em: <http://www.java.com/pt_BR/>. Acesso em: 23 mar. 2012.
- [51] JAVA TECHNOLOGY. Disponível em: < <http://java.sun.com/>>. Acesso em: 23 mar. 2012.
- [52] JENNINGS, N. R.. On Agent-based Software Engineering. *Artificial Intelligence*: 117, p. 277-296. 2000.
- [53] KENDAL, SIMON; CREEN, MALCOM. *An Introduction to Knowledge Engineering*. Londres: Springer Verlag, 2007. 287 p.
- [54] KNAPIK, M., and JOHNSON, J. *Developing Intelligent Agents for Distributed Systems*, Computing McGraw-Hill, NY:McGraw-Hill, 1998.
- [55] KRUEGER, C. W.. Software Reuse. *ACM Computing Surveys*, 24(2): 131-183, June 1992.
- [56] LEI Nº 10.520, DE 17 DE JULHO DE 2002. Presidência da República. Casa Civil. Subchefia para Assuntos Jurídicos. Disponível em:

- <http://www.planalto.gov.br/CCIVIL_03/LEIS/2002/L10520.htm>. Acesso em: 30 ago. 2012.
- [57] LEI Nº 8.433, DE 16 DE JULHO DE 1992. Presidência da República. Casa Civil. Subchefia para Assuntos Jurídicos. Disponível em: <http://www.planalto.gov.br/ccivil_03/leis/l8443.htm>. Acesso em: 01 jul. 2012.
- [58] LEI Nº 8.666, DE 21 DE JUNHO DE 1993. Presidência da República. Casa Civil. Subchefia para Assuntos Jurídicos. Disponível em: <http://www.planalto.gov.br/ccivil_03/leis/l8666cons.htm>. Acesso em: 01 mar. 2012.
- [59] LEITE, A. C., MADAE-Pro: Um processo baseado no conhecimento para engenharia de domínio e de aplicações multiagente. Dissertação (Mestrado em Engenharia de Eletricidade – Área de Ciência da Computação) - Universidade Federal do Maranhão, São Luís, 2009.
- [60] LEITE, A., GIRARDI, R. A Process For Multi-Agent Domain And Application Engineering: The Domain Analysis And Application Requirements Engineering Phases, Proceedings of the 11th International Conference on Enterprise Information Systems, Ed. INSTIIC. Milan, Italy, 2009.
- [61] LEITE, A., GIRARDI, R., & CALVALCANTE, U. (2008a). MAAEM: A Multiagent Application Engineering Methodology. Proceedings of the 20th International Conference on Software Engineering and Knowledge Engineering (pp. 735-740). Redwood City: Knowledge Systems Institute.
- [62] LEITE, A., GIRARDI, R., & CAVALCANTE, U. (2008b). An Ontology for MultiAgent Domain and Application Engineering. Proceedings of the 2008 IEEE International Conference on Information Reuse and Integration (IEEE IRI-08) (pp. 98-103). Las Vegas: IEEE.
- [63] LEMOS, A. J. D., ABARFI: Uma arquitetura baseada em agentes para a recuperação e filtragem de informação. Dissertação (Mestrado em Engenharia de Eletricidade – Área de Ciência da Computação) - Universidade Federal do Maranhão, São Luís, 2001.
- [64] LINDOSO, A. N., GIRARDI, R. The SRAMO Technique for Analysis and Reuse of Requirements in Multi-agent Application Engineering, IX Workshop on Requirement Engineering, Cadernos do IME, v. 20, Ed. UERJ, pp. 41-50. Rio de Janeiro. 13 a 14 de julho de 2006
- [65] MACCRIMMON, K. R. WEHRUNG, D. A. Taking Risks: the management of uncertainty. New York: Free Press. 1986
- [66] MARCIAL, E.C; GRUMBACH, R. J. S. Cenários Prospectivos: como construir um futuro melhor. 3.ed. Rio de Janeiro: FGV, 2005.
- [67] MEDAUAR, O. Direito administrativo moderno. 6ª ed. rev. E atual. São Paulo: Revista dos Tribunais, 2002.
- [68] MEDIDA PROVISÓRIA Nº 2.026, DE 04 DE MAIO DE 2000. Presidência da República. Casa Civil. Subchefia para Assuntos Jurídicos. Disponível em: <http://www.planalto.gov.br/ccivil_03/mpv/Antigas/2026.htm>. Acesso em: 01 mar.

2012.

- [69] MEIREILLES, Hely Lopes. Direito administrativo brasileiro. 19ª ed. Atual. São Paulo; Malheiros, 1994.
- [70] MELLO, C. A. B. de, Curso de direito administrativo, 12ª ed. rev. ampl. E atual. São Paulo: Malheiros, 2000.
- [71] MOORE, P. G. The Bussiness of Risk. Cambridge: Cambridge University Press. 1997
- [72] NETBEANS. NetBeans IDE 7.1.2. Disponível em: < <http://netbeans.org/>>. Acesso em: 15 mai. 2012.
- [73] NILSSON, N. Principles of Artificial Intelligence, Springer-Verlag, 1982.
- [74] NORVIG, P, RUSSELL, S. Inteligência Artificial. Ed. Campus. São Paulo, 2004.
- [75] OLIVEIRA, R. N., Aprendizagem de Máquina em um Ambiente para Negociações Automatizadas. Dissertação (Mestrado em Engenharia de Eletricidade – Área de Ciência da Computação) - Universidade Federal de Campina Grande, Campina Grande, 2006.
- [76] OSÓRIO, F. M.. Existe uma supremacia do interesse público sobre o privado no direito administrativo brasileiro?. Revista de direito administrativo, Rio de Janeiro, v. 220, p. 69-107, abr./jun. 2000.
- [77] P. H. Winston and B. K. P. Horn. LISP. Addison-Wesley. Reading, MA. 1981.
- [78] P. H. Winston. Artificial Intelligence. Addison-Wesley. Reading, MA. 1984.
- [79] PEREZ, A.L.F.; POZZEBON, E.; BITTENCOURT, G. Um Ambiente Interativo Multiagente para Licitação Pública. Florianópolis. 2004
- [80] POKAHR, A., BRAUBACH, L., LAMERSDORF, W. Jadex: Implementing a BDI-Infrastructure for JADE Agents, 2003,pp.76-85.
- [81] PRIETO-DÍAZ, R. 1993. Status Report: Software Reusability. IEEE Softw. 10, 3 (May. 1993), 61-66. DOI= <http://dx.doi.org/10.1109/52.210605>.
- [82] PROTÉGÉ. Disponível em: <<http://protege.stanford.edu>>, Acessado em: 01 ago. 2012.
- [83] RUIZ, F. H. Using Ontologies in Software Engineering and Technology, chapter 2. In: Ontologies for Software Engineering and Software Technology. Springer. 2006.
- [84] RUSSEL, S. NORVIG, P. Inteligência Artificial. 2ª Ed. Rio de Janeiro: Elsevier, 2004.
- [85] RUSSEL, S., NORVIG, P., Artificial Intelligence: A Modern Approach, Prentice-Hall, 2009.
- [86] SCHREIBER, GUUS et al. Knowledge Engineering and Management: The CommonKADS Methodology. Massachussets: MIT Press, 2002. 401 p.
- [87] SCHWARTZ, P. A Arte da Previsão: planejando o futuro em um mundo de incerteza. São Paulo: Editora Página Aberta, 1991.
- [88] SCHWARTZ, P. A Arte da Visão de Longo Prazo. 3. ed. Rio de Janeiro: Best Seller, 2004.

- [89] SHADBOLT, N.; MOTTA, E.; ROUGE, A. Constucting Knowledge-Based Systems. IEEE Software, November, 1993, p. 34-38.
- [90] SHIMIZU, T. Decisão nas Organizações. São Paulo: Atlas, 2001.
- [91] SHORTLIFFE, E. Computer-Based Medical Consultations: mycin. New York: American Elsevier, 1976.
- [92] SILVA JUNIOR, G. B. Padrões Arquiteturais para o Desenvolvimento de Aplicações Multiagente. Dissertação (Mestrado em Engenharia de Eletricidade – Área de Ciência da Computação), Universidade Federal do Maranhão (CPGEE/UFMA). São Luís-MA. 2003.
- [93] SPRAGUE Jr, R.H.; WATSON, H.J. Decision Support Systems – Putting Theory into Practice. Prentice-Hall. Inc. 1989
- [94] SYCARA, K. P. Multiagent Systems. AI Magazine 19(2): 79-92. 1998.
- [95] TELES, R. M.; IVAMOTO, M.; MELLO, L. H. S.; GRACIANO NETO, V. V.; CARVALHO, C. L. Um Sistema de Apoio à Decisão Baseado em Agentes para Tratamento de Ocorrências no Setor Elétrico. Revista de Informática Teórica e Aplicada, Porto Alegre, v. VIII, n. 3, p. 73-87, 2001.
- [96] TURBAN, E. MEREDITH, J. R. Fundaments of management science. 6ª ed. Boston: Irwin. 1994
- [97] USCHOLD, M. Building Ontologies: Towards A Unified Methodology. Proceedings Expert Systems 96. Cambridge, 1996.
- [98] WOOLDRIDGE, M. JENNINGS, N. R. Agent-Oriented Software Engineering. In 9th European Workshop on Modelling Autonomous Agents in a Multi-Agent World. 1999.
- [99] WOOLDRIDGE, M., JENNINGS, N.. Intelligent Agents: Theory and Practice. The Knowledge Engineering Review, v. 10 (2), p. 115-152, 1995.

APÊNDICES

APÊNDICE A – Implementação do Agente “Interface” em JADE

```

import jade.core.Agent;
import jade.core.AID;
import jade.lang.acl.*;
import jade.core.behaviours.*;
import java.sql.*;

public class Interface extends Agent {
    Connection con = null;
    boolean erro = false;
    int novo = 0, novoPed = 0, novoProj = 0, quant = 0, quantPed = 0, quantProj = 0, codprojeto = 0;

    public void setup() {

        try {
            Class.forName("com.mysql.jdbc.Driver");
            Connection con =
                DriverManager.getConnection("jdbc:mysql://localhost/proc_licitatorio","root","123");
            PreparedStatement stmt = con.prepareStatement("select count(*) as quant from
miniprojeto where aprovado = 'S'");
            ResultSet rs = stmt.executeQuery();

            if (rs.next()) {
                quant = rs.getInt("quant");
            }
            rs.close();
            stmt.close();
            stmt = con.prepareStatement("select count(*) as quant from pedidos");
            rs = stmt.executeQuery();

            if (rs.next()) {
                quantPed = rs.getInt("quant");
            }

            rs.close();
            stmt.close();

            stmt = con.prepareStatement("select count(*) as quant from miniprojeto");
            rs = stmt.executeQuery();

            if (rs.next()) {
                quantProj = rs.getInt("quant");
            }

            con.close();
            System.out.println("Agente Interface aguardando novo pedido.");
        }
        catch (ClassNotFoundException e) {
            System.out.println("erro: "+e.getMessage());
        }
        catch (SQLException e) {
            System.out.println("erro: "+e.getMessage());
        }

        addBehaviour(new Behaviour(this) {

            public void action() {

```

```

try {
    ACLMessage msg = receive();
    if (msg != null)
        System.out.println(msg.getContent());

    Class.forName("com.mysql.jdbc.Driver");
    Connection con =
DriverManager.getConnection("jdbc:mysql://localhost/proc_licitatorio","root","123");
    PreparedStatement stmt = con.prepareStatement("select count(*) as quant from
miniprojeto where aprovado = 'S'");
    ResultSet rs = stmt.executeQuery();

    if (rs.next()) {
        novo = rs.getInt("quant");
    }

    rs.close();
    stmt.close();

    if (novo > quant) {
        stmt = con.prepareStatement("select numero from miniprojeto where
aprovado = 'S' order by numero desc");
        rs = stmt.executeQuery();
        rs.next();
        codprojeto = rs.getInt("numero");
        msg = new ACLMessage(ACLMessage.INFORM);
        msg.setContent(String.valueOf(codprojeto).trim());
        msg.addReceiver(new AID("analista", AID.ISLOCALNAME));
        send(msg);
        System.out.println("Agente interface: Miniprojeto " + codprojeto + "
enviado para agente analista.");
        rs.close();
        stmt.close();
        quant++;
    }

    stmt = con.prepareStatement("select count(*) as quant from pedidos");
    rs = stmt.executeQuery();

    if (rs.next()) {
        novoPed = rs.getInt("quant");
    }

    rs.close();
    stmt.close();

    if (novoPed > quantPed) {
        stmt = con.prepareStatement("select numero from pedidos order by numero
desc");

        rs = stmt.executeQuery();
        rs.next();
        int codpedido = rs.getInt("numero");
        System.out.println("Agente Interface: setor de compras, logue no sistema e
insira um projeto para o pedido " + codpedido + ".");
        rs.close();
        stmt.close();
        quantPed++;
    }

    stmt = con.prepareStatement("select count(*) as quant from miniprojeto");

```

```

rs = stmt.executeQuery();

if (rs.next()) {
    novoProj = rs.getInt("quant");
}

rs.close();
stmt.close();

if (novoProj > quantProj) {
    stmt = con.prepareStatement("select numero from miniprojeto order by
numero desc");

    rs = stmt.executeQuery();
    rs.next();
    int codprojeto = rs.getInt("numero");
    System.out.println("Agente Interface: comissao permanente, logue no sistema e
analise o miniprojeto " + codprojeto + ".");
    msg = new ACLMessage(ACLMessage.INFORM);
    msg.setContent("0Miniprojeto " + codprojeto + " recebido.");
    msg.addReceiver(new AID("analista", AID.ISLOCALNAME));
    send(msg);
    System.out.println("Enviei mensagem para agente analista:" + "Miniprojeto "
+ codprojeto + ".");

    rs.close();
    stmt.close();
    quantProj++;
}

con.close();

try {
    Thread.sleep(500);
}
catch(InterruptedException e){}
}
catch (ClassNotFoundException e) {
    System.out.println("erro 1: "+e.getMessage());
    e.printStackTrace();
}
catch (SQLException e) {
    System.out.println("erro 2: "+e.getMessage());
    e.printStackTrace();
}
}

public boolean done() {
    return false;
}
});
}
}

```

APÊNDICE B – Implementação do Agente “Analista” em JADE

```

import jade.core.Agent;
import jade.core.AID;
import jade.lang.acl.*;
import jade.core.behaviours.*;
import java.sql.*;

public class Analista extends Agent {

    public void setup() {

        addBehaviour(new CyclicBehaviour(this) {

            public void action() {

                ACLMessage msg = receive();
                int codprojeto = 0;

                if (msg != null) {

                    String mensagem = msg.getContent();

                    if (!mensagem.substring(0,1).equals("0"))
                    {
                        codprojeto = Integer.parseInt(msg.getContent());
                        msg = new ACLMessage(ACLMessage.INFORM);
                        msg.setContent("Mini projeto recebido: " + codprojeto);
                        msg.addReceiver(new AID("interface", AID.ISLOCALNAME));
                        send(msg);

                        try {
                            Class.forName("com.mysql.jdbc.Driver");
                            Connection con =
DriverManager.getConnection("jdbc:mysql://localhost/proc_licitatorio","root","123");
                            PreparedStatement stmt = con.prepareStatement("select *
from miniprojeto where numero = " + codprojeto);
                            ResultSet rs = stmt.executeQuery();
                            rs.next();
                            if (rs.getString("carater").equals("S")) {
                                System.out.println("Agente Analista: Projeto numero "
+ codprojeto+ " aprovado!");

                                msg = new ACLMessage(ACLMessage.INFORM);
                                msg.setContent(String.valueOf(codprojeto));
                                msg.addReceiver(new AID("licitador",
AID.ISLOCALNAME));

                                send(msg);
                                block();
                            }
                            rs.close();
                            stmt.close();
                            con.close();
                        }

                        catch (ClassNotFoundException e) {
                            System.out.println("erro: "+e.getMessage());
                            e.printStackTrace();
                        }
                    }
                    catch (SQLException e) {
                        System.out.println("erro: "+e.getMessage());
                    }
                }
            }
        });
    }
}

```

```
                e.printStackTrace();
            }
        }
        else
            System.out.println("Agente analista: " + mensagem.substring(1));
    }
}
});
}
```

APÊNDICE C – Implementação do Agente “Licitação” em JADE

```

import java.sql.*;
import jade.core.Agent;
import jade.core.AID;
import jade.lang.acl.*;
import jade.core.behaviours.*;
import jpl.Query;

public class Licitador extends Agent {
    public void setup() {
        addBehaviour(new CyclicBehaviour(this) {
            String codprojeto = "";
            public void action() {
                ACLMessage msg = receive();
                if (msg != null) {
                    System.out.println("Agente Licitador: Recebido projeto numero " +
msg.getContent());
                    codprojeto = msg.getContent();
                    block();
                }
                try {
                    if (!codprojeto.equals("")) {
                        Class.forName("com.mysql.jdbc.Driver");
                        Connection con =
DriverManager.getConnection("jdbc:mysql://localhost/proc_licitatorio","root","123");
                        PreparedStatement stmt = con.prepareStatement("select * from
miniprojeto where numero = " + codprojeto + " and valor > 0");
                        ResultSet rs = stmt.executeQuery();
                        if (rs.next()) {
                            if (rs.getDouble("Valor") > 0) {
                                String engenharia =
rs.getString("engenharia");
                                double valorMedio = rs.getDouble("valor");
                                String consulta =
"consult('E:/bck_netbook/Mestrado/Dissertação/Proc_Licitatorio/src/prolog/regras_licitacao.pl)";
                                Query q1 = new Query(consulta);
                                q1.hasSolution();
                                Query obraEngenharia = new
Query("assert(obraEngenharia("+engenharia+"))");
                                obraEngenharia.hasSolution();
                                Query obraValor = new
Query("assert(obraValor("+valorMedio+"))");
                                obraValor.hasSolution();
                                Query cbLicitacao_pl = new
Query("cbLicitacao(X)");
                                Query cbTipo_pl = new Query("cbTipo(X)");
                                String modalidade = "";
                                String tipo = "";
                                if
(cbLicitacao_pl.oneSolution().get("X").toString().equals("1"))
modalidade = "Modalidade de
Licitacao: Concorrencia";
                                else if
(cbLicitacao_pl.oneSolution().get("X").toString().equals("2"))
modalidade = "Modalidade de
Licitacao: Tomada de preco";
                                else if

```



```

(cbLicitacao_pl.oneSolution().get("X").toString().equals("3"))
Licitacao: Convite";
Precos";
(cbTipo_pl.oneSolution().get("X").toString().equals("1"))
Precos";
(cbTipo_pl.oneSolution().get("X").toString().equals("3"))
Precos";

modalidade = "Modalidade de
tipo = "Tipo de Licitacao: Menor
tipo = "Tipo de Licitacao: Tecnica e

System.out.println(modalidade);
System.out.println(tipo);
codprojeto = "";

}
try {
Thread.sleep(1000);
}catch(InterruptedExceção e){}
}
}
catch (ClassNotFoundException e) {
System.out.println("erro: "+e.getMessage());
e.printStackTrace();
}
catch (SQLException e) {
System.out.println("erro: "+e.getMessage());
e.printStackTrace();
}
});
}
}

```