

Modelagem Orientada a Agentes utilizando MESSAGE

Andrey M. Coppieters, Leandro Augusto J. Marzulo, Eduardo Kinder e Vera M. B. Werneck

UERJ - Universidade Estadual do Rio de Janeiro

Departamento de Informática e Ciência da Computação

Rua São Francisco Xavier 524, 6o Andar, Bloco B

Maracanã - 20 550-013 - Rio de Janeiro – Brasil

vera@ime.uerj.br

Resumo

A Modelagem Orientada à Agentes vem suprir a demanda pelo desenvolvimento de novas aplicações, que atendam aos requisitos e características das organizações sociais e de seus relacionamentos, de forma autônoma e integrada. O estudo deste novo paradigma é ainda recente, entretanto várias metodologias foram propostas que adotam este conceito para o desenvolvimento de sistemas. Neste trabalho é apresentada a metodologia MESSAGE, exemplificada na modelagem de um estudo de caso prático e complexo, expondo seus pontos fortes e suas desvantagens.

Abstract

The Agent-Oriented modeling comes to fill the growing demand for new application developments, which should sustain the social organizations requirements and characteristics, besides the relationships involved, in an autonomous and integrated way. The study of this new Agent-Oriented design approach is still recent, even though many systems development methodologies that adopt this concept have been proposed so far. This paper presents the MESSAGE methodology and study case modeling, exposing their strengths and disadvantages.

1. Introdução

O crescimento da demanda pelo desenvolvimento de novas aplicações, que atendam aos requisitos e características das organizações sociais e de seus relacionamentos, de forma autônoma e integrada, tem impulsionado a pesquisa por novos designs patterns de software que consigam suportar naturalmente estas propriedades. A Modelagem Orientada à Agentes vem suprir muitas destas necessidades, almejando somar-se às demais abordagens existentes, como Orientada à Objetos (OO), na construção de sistemas que tenham na autonomia, na mobilidade e na capacidade de coordenação

e adaptação, os aspectos fundamentais de seu funcionamento.

Este trabalho tem como objetivo apresentar os conceitos envolvidos numa modelagem orientada a agentes. Além dos conceitos da metodologia MESSAGE [1], [2], [3], [4], será apresentado um exemplo de estudo de caso prático, baseado num projeto denominado “Guardian Angel” [5]. Este consiste em um sistema de informação multi-agentes para a área de saúde, visando possibilitar a pacientes, portadores de condições crônicas como diabetes e pressão alta, armazenar consigo seus registros médicos e lidar de maneira mais independente com seu tratamento, ainda que devidamente monitorados.

Neste artigo apresentaremos na seção 2, os conceitos básicos, modelos e diagramas da metodologia MESSAGE. Na seção 3 o estudo de caso será descrito através de exemplos dos modelos e diagramas da metodologia. Na seção 4 será feita uma avaliação geral abordando as vantagens e desvantagens dessa experiência obtida no uso da metodologia MESSAGE. No capítulo 5, descreve-se as conclusões deste trabalho e seu contexto.

2. Método de Desenvolvimento MESSAGE

A tecnologia de Agentes está sendo aplicada como solução no domínio das aplicações de Telecomunicações. Com este propósito, um instituto de pesquisas da área de telecomunicações direcionou um dos seus projetos, o P907 [2], [3], [4], para a definição da metodologia MESSAGE para a análise e design baseados na orientação a agentes.

MESSAGE [1], [2], [3], [4], utiliza a UML (Unified Modeling Language) [6] como ponto de partida, adicionando conceitos de entidades e relacionamentos necessários à modelagem orientada a agentes. Esta modelagem busca descrever a forma como um sistema multi-agentes funciona para a realização de um objetivo coletivo, similarmente às organizações humanas e sociedades, além do comportamento cognitivo dos agentes, com o auxílio da Inteligência Artificial.

Tais conceitos lidam com idéias e estruturas em um nível conceitual maior que o da orientação a objetos,

referenciado na área de Inteligência Artificial como “Nível de Conhecimento”, estabelecendo um contraste entre o conhecimento e dados. A MESSAGE [1], [2], [3], [4], [7], [8] usa a linguagem padrão de Orientação a Objetos, UML (Unified Modeling Language) [6] como padrão de linguagem à “nível de dados”, acrescentando os conceitos ao “nível de conhecimento” que estão descritos em seu meta-modelo, que também provê uma interpretação de alguns conceitos da UML [6] usados para descrição do comportamento, dentre os quais a noção de “estado”.

2.1. Nível de Conhecimento

A maior parte dos conceitos de entidades, existentes no nível de conhecimento do MESSAGE [1], [2], [3], [4], [7], [8], podem ser agrupadas em três categorias: Entidade Concreta, Atividade e Entidade de Estado Mental. Na primeira categoria, estão contidas as noções de Agente, Organização, Papel e Recurso. A segunda contém as noções de Tarefa, Interação e Protocolo de Interação. A terceira tem um foco maior no conceito de Objetivo.

Agente é definido como uma entidade autônoma atômica, capaz de realizar alguma função útil. Esta competência funcional é extraída dos serviços do agente, tal que um serviço seja o nível de conhecimento análogo a uma operação de um objeto. A sua autonomia é referenciada como um atributo, chamado propósito, que poderá influenciar como um agente reage à um pedido para a execução de um determinado serviço, além da forma como a qual este será provido. Neste âmbito, os agentes de software e os agentes humanos são especializações do Agente.

Organização é considerada um grupo de agentes trabalhando em conjunto para o alcance de um propósito comum. Trata-se de uma entidade virtual no sentido de que não possui uma entidade computacional individual correspondente. Seus serviços são providos coletivamente pelos agentes que a constituem, que estão conectados por relacionamentos organizacionais, como os existentes em uma organização humana tal como o de superior-subordinado, por exemplo, por procedimentos de controle e gerenciamento, por fluxos de trabalho e interações.

O conceito de Papel permite que um agente em uma determinada situação esteja logicamente separado da identidade do próprio agente. A distinção entre Papel e Agente é análoga à existente entre a Classe de um objeto e sua interface. Um agente pode fazer diferentes papéis e vários agentes podem ser capazes de fazer um mesmo papel, visto que este descreve as características externas de um agente em um contexto particular.

Recurso representa entidades que não são autônomas, como bases de dados e programas externos que são utilizados pelos agentes. Os conceitos padronizados da orientação a objetos são apropriados para a modelagem dos recursos.

Tarefa é uma unidade de nível de conhecimento de atividade, com um único executor primário, possuindo estados definidos por suas pré-condições, seu progresso e pós-condições. Caso as pré-condições estejam válidas para a realização da tarefa, pode-se esperar que as pós-condições sejam satisfeitas após o término da mesma. As tarefas podem ser compostas por sub-tarefas que estão relacionadas entre si por relacionamentos lógicos e temporais. Os relacionamentos lógicos, também, são usados para associar as tarefas aos objetivos que visam atingir.

O conceito de Interação e Protocolo de Interação está fortemente relacionado ao introduzido pela metodologia Gaia. Uma interação é composta por um ou mais participantes e o propósito que estes pretendem atingir. O propósito geralmente é a obtenção de uma visão consistente de algum aspecto do domínio do problema, ou o estabelecimento dos termos e resultados de um ou mais serviços. Um protocolo de interação define um padrão para a troca de mensagens existente em uma interação.

O conceito Objetivo associa o agente a um estado, de maneira que se uma instância de um objetivo encontra-se na memória de trabalho do agente, então este pretende causar o estado relacionado àquele objetivo. Alguns objetivos são derivados do propósito do agente, intrínsecos à sua identidade, sendo persistentes por toda a vida útil deste. Há também os objetivos do tipo transientes, usados geralmente para expressar o propósito do agente por funções utilitárias que associam valores de excelência aos estados, tendo como alvo maximizar a utilidade.

O conceito de Mensagem também é importante na MESSAGE, representando um objeto comunicado de um agente à outro, diferentemente do apresentado pela UML [6], na qual trata-se de uma ligação causal em uma cadeia comportamental, onde determinada ação realizada por um objeto é provocada pela ação de outro. Outro conceito é o de Entidade de Informação que consiste na descrição do conteúdo de uma mensagem, que também é composta por atributos que especificam o seu remetente, o destinatário e sua categorização conforme o plano do remetente.

2.2. Visões do Modelo de Análise

O modelo de análise consiste em uma complexa rede de classes e instâncias inter-relacionadas, derivadas dos conceitos definidos no meta-modelo da metodologia. Devido à análise orientada a agentes apresentar um conjunto de conceitos muito rico, dificultando a compreensão de todos os aspectos envolvidos a partir de um único ponto de vista, torna-se necessário o uso de várias perspectivas do modelo [1], [2], [3], [4], [7], [8].

A estruturação do modelo em visões é amplamente usada pelas metodologias para descrever vários aspectos de um sistema, sendo de suma importância que se assegure

à consistência entre as diferentes visões para que esta técnica obtenha sucesso. MESSAGE utiliza as seguintes visões: Organizacional, de Objetivo/Tarefa, de Agente/Papel, da Interação e de Domínio.

A Visão Organizacional representa as entidades concretas no sistema e no seu ambiente, além de seus relacionamentos como os de agregação e de conhecimento, este último que indica que há pelo menos uma interação entre as entidades. Estas entidades são agentes, organizações, papéis ou recursos.

A Visão de Objetivo/Tarefa representa o objetivo, tarefas, estados e as suas dependências. É possível representar um objetivo como a composição de uma série de sub-objetivos, de um nível menor, e como as tarefas que propiciam o alcance daquele objetivo são executadas, além da dependência temporal. Para tal, emprega-se o Diagrama de Atividades da UML.

A Visão de Agente/Papel descreve para cada par agente/papel um diagrama que retrata os objetivos, os eventos percebidos, os recursos controlados, suas regras de comportamento e quais tarefas que sabe como realizar.

A Visão da Interação tem enfoque na interação entre agentes/papéis. Para cada interação mostra-se o seu iniciador, os colaboradores, o motivador (objetivo do iniciador), as informações importantes fornecidas e obtidas por cada participante da interação, os eventos que a disparam, dentre outros de seus efeitos relevantes.

A Visão de Domínio retrata as relações e os conceitos específicos do domínio do sistema que está em desenvolvimento.

2.3. O Processo de Análise

A Análise tem como propósito produzir uma especificação, que é um modelo do sistema a ser desenvolvido e do seu ambiente, resultado de um acordo entre o analista e o cliente. O modelo auxilia na comunicação entre a equipe de desenvolvimento e o cliente, provendo uma estrutura base a partir da qual a construção do software pode proceder de forma confiante.

Na metodologia MESSAGE [1], [2], [3], [4], [7], [8], assume-se que o ponto de partida para a análise é uma definição informal dos requisitos, feita pelo cliente, somada à uma documentação, na qual consta: o organograma, que costuma ter uma estrutura hierárquica e descreve como o negócio da empresa está estruturado em termos das pessoas e suas responsabilidades; a descrição dos objetivos da empresa, que pode consistir da declaração de qual é a missão da empresa, além de informações adicionais; os processos de negócio, que retratam a forma como a qual as atividades são executadas na organização, representados por fluxos de trabalho; e a descrição das atividades da empresa, que está relacionada aos processos de negócio, porém com ênfase na terminologia e na

natureza das informações que a empresa usa no seu dia-a-dia.

A MESSAGE emprega uma técnica de análise que consiste em um refinamento, em níveis, na qual estágios subsequentes do refinamento resultam na criação de modelos nos diferentes níveis, que devem ser consistentes entre si.

O nível mais alto para decomposição é denominado Nível 0, no qual procura-se definir o sistema a ser desenvolvido em relação aos seus interessados e ao seu ambiente, dando uma idéia geral sobre suas funcionalidades gerais. A granularidade desta fase tem enfoque na identificação das entidades e seus relacionamentos, conforme o meta-modelo. O sistema é visto como um conjunto de organizações que interage com recursos, outras organizações, ou atores, estes podendo ser humanos ou outros agentes já existentes. Neste nível, o processo de modelagem tem início na elaboração das visões de Objetivo/Tarefa e Organizacional, que são usadas como entradas para a criação das visões de Agente/Papel e de Domínio da Informação. Todas estas visões auxiliam a construção da visão de Interação.

No Nível 1, são definidos o comportamento e a estrutura interna das entidades identificadas, tais como agentes, organização, tarefas e objetivos. É possível adicionar níveis posteriores para a especificação de requisitos funcionais e não-funcionais como performance e segurança, porém, a MESSAGE só engloba os dois primeiros níveis.

Há várias maneiras de se fazer o refinamento dos modelos do Nível 0. A geração de várias visões do sistema, aplicada pela MESSAGE, propicia uma liberdade de escolha ao analista sobre a melhor estratégia a ser adotada. Porém, usualmente, procura-se utilizar uma combinação destas estratégias de refinamento.

Uma das abordagens é a Centralizada na Organização, que focaliza na análise das propriedades gerais como: a estrutura do sistema, os serviços oferecidos, objetivos e tarefas gerais, os papéis principais e os recursos. Durante o processo de refinamento, os agentes necessários à realização dos objetivos são identificados e então, a cooperação, possíveis conflitos e suas resoluções, podem ser analisados.

A abordagem Centralizada no Agente tem como foco a definição dos agentes necessários para que as funcionalidades do sistema possam ser providas, sendo que a organização mais apropriada é identificada de acordo com os requisitos do sistema.

A abordagem Orientada à Interação prega um refinamento progressivo dos cenários de interação, que caracterizam o comportamento interno e externo da organização e dos agentes. Estes cenários comportam-se como fontes para a caracterização das tarefas, objetivos, mensagens, protocolos e entidades do domínio.

A abordagem por Decomposição de Objetivo/Tarefa baseia-se na decomposição funcional. Os papéis, objetivos e tarefas do sistema são sistematicamente analisados para que sejam determinadas as condições de decisão, os métodos para resolução de problemas e tratamento de falhas. As pré-condições, estruturas, saídas e pós-condições das tarefas podem definir quais as entidades de domínio necessárias. Como um agente, fazendo um papel, realiza certas tarefas e atinge objetivos, a análise sobre a estrutura geral da Visão de Objetivo/Tarefa torna possível a tomada de decisões sobre os agentes, ou organização, mais adequados para execução daqueles objetivos e tarefas.

Uma das principais características do agente é a sua autonomia, ou seja, é capaz de decidir o que fazer de acordo com o seu propósito e sob dadas circunstâncias. No processo de análise da MESSAGE, considera-se uma abordagem top-down para a definição da estrutura do sistema multi-agentes. Para verificar se o comportamento destes agentes não apresenta desvios prejudiciais ao sistema como um todo, deve-se fazer uma análise bottom-up. Assim, o comportamento do sistema será avaliado em relação aos seus requisitos, de forma que, caso aspectos indesejáveis sejam encontrados, a fase top-down do ciclo deverá ser revisada.

2.4. O Processo de Projeto

O propósito do projeto é definir as entidades computacionais que representam o sistema multi-agentes (MAS) representado na fase de análise. Estes artefatos são transformados para possibilitar a implementação. A fase de projeto na metodologia MESSAGE [1], [2], [3], [4], [7], [8], é dividida em dois processos: projeto de alto nível e projeto detalhado.

No projeto de alto nível o modelo de análise é refinado produzindo uma versão inicial da arquitetura MAS. Essa atividade é realizada em quatro etapas: (i) atribuir papéis aos agentes e definir o diagrama de agentes; (ii) prover serviços às tarefas, detalhando-as através de um workflow (diagrama de atividades da UML); (iii) refinar os protocolos de interação através do uso de protocolos de interação da AUML e de diagramas de estado do UML; (iv) especificar o comportamento da interação do papel em diagramas de estado do UML.

O projeto detalhado pressupõe a definição das estruturas de implementação possíveis para o projeto dos agentes, sendo que o MESSAGE define duas abordagens. A primeira é direcionada por uma arquitetura agente e por uma organização multi-agentes, considerando que o agente é uma entidade mais abrangente que uma classe, sendo encarado como um sub-sistema com uma arquitetura interna que define os relacionamentos dos seus diferentes componentes.

A segunda abordagem é mais orientada à plataforma agente, levando-se em conta que cada agente pode ser mapeado para uma classe. Este conceito é extraído dos modelos suportados pela maioria das ferramentas para a implementação de agentes, como a JADE [9] e a FIPA-OS [10], nas quais há uma classe Agente a partir da qual tipos específicos de agentes são derivados.

O projeto detalhado não será escopo deste trabalho que pretende modelar somente até de projeto de alto nível.

3. Estudo de Caso Guardian Angel

Nesta seção, é apresentada a modelagem do estudo de caso Guardian Angel [5], utilizando a metodologia MESSAGE [1], [2], [3], [4], [7], [8].

Esse estudo de caso é um sistema de informação multi-agentes para apoiar o tratamento médico permitindo aos pacientes um controle maior sobre as suas informações de saúde e progressos dos tratamentos.

A idéia do Guardian Angel é o acompanhamento integral de pacientes com doenças crônicas, como diabéticos e hipertensos, através de dispositivos móveis (PDAs), com a vantagem de evitar burocracias, indisponibilidade dos médicos (telefones ocupados, agendas lotadas) e do paciente, além de diminuir custos.

Existem pelo menos três instâncias do Guardian Angel: a implementação pessoal (GA_PDA) para cada paciente, a implementação do software em um computador fixo do paciente (GA_Home) e o sistema utilizado pelo médico (GA_Hospital). Uma versão móvel para ambulâncias também pode ser cogitada.

Um paciente diabético precisa pelo menos três vezes por semana medir o nível de glicose no sangue e conversar com um médico para saber se é necessário fazer alguma alteração de medicamento ou tomar outra providência. Com o Guardian Angel, é possível fazer isto, sem sair de casa ou marcar consulta. O sistema permite monitorar o nível de açúcar com a ajuda de um medidor portátil, e enviar para o computador as informações obtidas. O Guardian Angel faz uma série de perguntas e interpreta suas respostas, cruzando as informações e o índice de açúcar com o perfil do paciente e os parâmetros fornecidos pelo médico. Nos casos de alteração grave ou moderada ou oscilações no nível de açúcar, os médicos responsáveis pelo tratamento recebem um e-mail ou outra forma de mensagem (através do Guardian Angel), e podem responder marcando uma consulta ou alterando o tratamento atual do paciente (medicamentos, alimentação, exercícios físicos, etc).

Os diagramas encontram-se em inglês, pois esse trabalho faz parte de um projeto de avaliação de metodologias orientadas a agentes [11], [12]. Este pretende submeter as avaliações e disponibilizar as modelagens ao autores das metodologias.

3.1. Modelos de Análise

Na fase de análise foram utilizados os níveis zero e um. O Nível 0 (zero) apresenta uma visão geral da organização e

os principais objetivos sendo construídos os Diagramas de Organização com relacionamento estrutural e relacionamento de conhecimento e Diagrama de Objetivos e Tarefas.

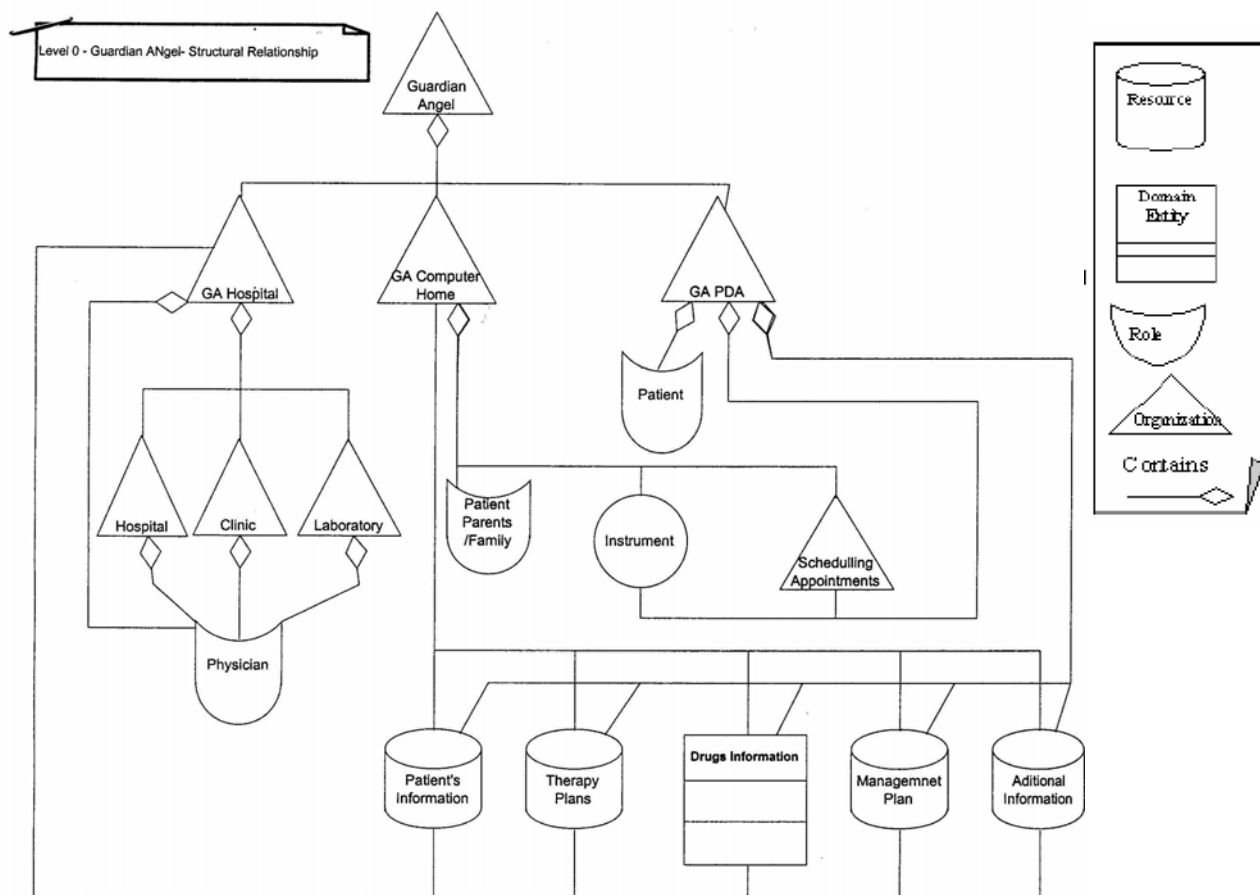


Figura 1 – Diagrama de Organização – Nível 0 – Relacionamento Estrutural

A figura 1 descreve o relacionamento estrutural no nível 0 do diagrama de organização. Nesse diagrama o Sistema Guardian Angel é dividido em 3 subsistemas cada um se comunicando com diferentes organizações (Hospital, Clínica, Laboratório, Marcador de Consultas (*Scheduling Appointments*), papéis (Médico, Pacientes, Pais/Família dos Pacientes) e recursos (Informação de Pacientes, Informação de Remédios, Plano do Tratamento, Gerenciamento Plano).

O objetivo do GA (Guardian Angel) é provê assistência médica através de 5 grandes objetivos: Assistência ao Paciente, Otimização do trabalho do Hospital, Assistir o médico, Assistir os Pais do Paciente e Prover Interfaces.

O objetivo do Guardian Angel em relação a Assistência ao Paciente se sub-divide nos seguintes sub-objetivos: Guardar Registros Médicos do Paciente, Customizar o Plano de Terapia, Monitorar o Tratamento seguido através

de Verificação do Tratamento Realizado e das Medidas Médicas do Paciente, Informação sobre Remédios e Similares e Gerenciamento de Consultas.

O objetivo Assistência ao Médico se sub-divide em: Definir Plano de Terapia, Administrar Agenda, Acessar Informações de Remédios, Acessar Informações de Doenças e Monitorar Tratamento de Pacientes.

A figura 2 apresenta parte do Diagrama dos Objetivos segundo a visão do nível 0.

Na figura 3 o diagrama de organização do nível 0 mostra os relacionamentos de conhecimento do sistema Guardian Angel. Este interage com quatro papéis: Paciente, Médico, Pais e Assistência Médica; e utiliza três recursos de base de dados de Consultas, Remédios/Doenças/Tratamentos e Planos/Registros.

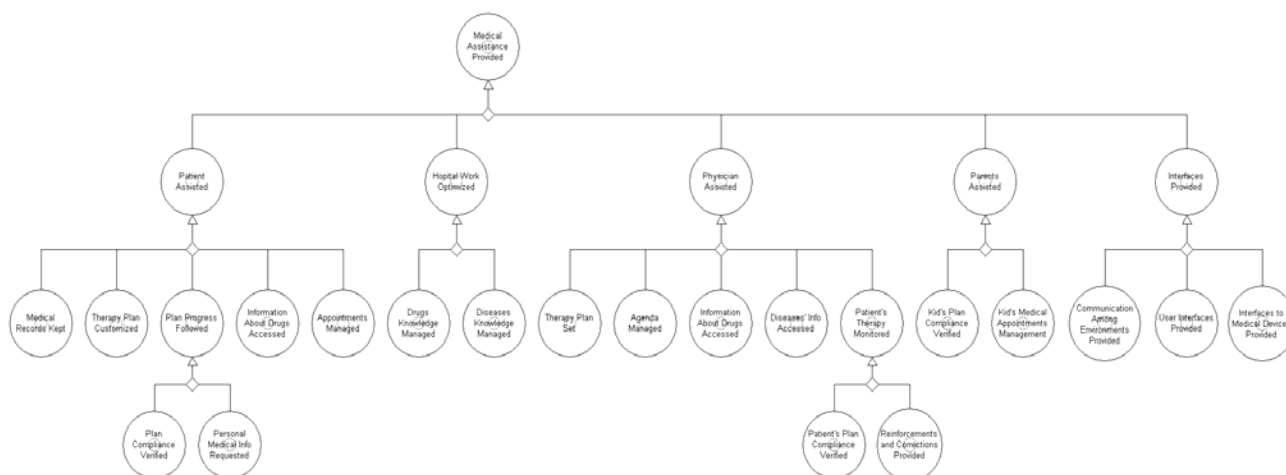


Figura 2 – Parte do Diagrama de Objetivos/Tarefas – Nível 0

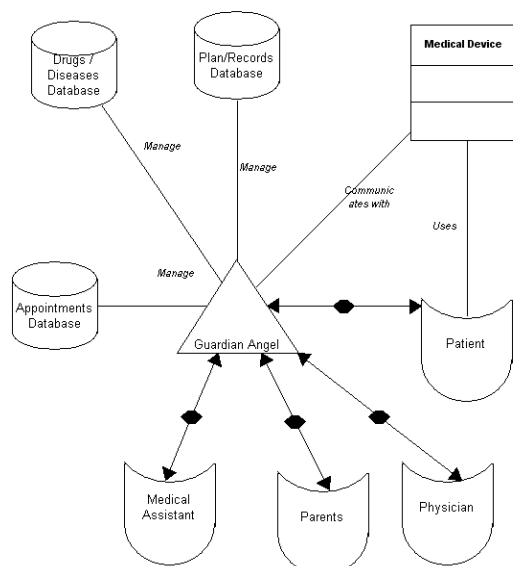


Figura 3 - Diagrama de Organização – Nível 0 – Relacionamento de Conhecimento

No nível 1 o sistema é detalhado segundo as visões da organização, e domínio, agentes/papéis e serviços. O diagrama da organização do nível 1 do conhecimento (figura 4) mostra os diferentes papéis, seus recursos e base de dados. Exemplos de papéis: analisador de plano, broker da interface com usuário, gerenciador de consultas.

O diagrama da delegação (figura 5) apresenta os objetivos relacionados aos diferentes papéis. A visão de interação do nível 1 detalha a comunicação entre 2 papéis e apresenta os recursos utilizados nessa interação. A figura 6 apresenta um exemplo de Diagrama de Interação.

O domínio pode ser descrito usando o diagrama de classes do UML para modelar classes, suas associações e seus atributos. A definição completa dessa modelagem estará disponível na página do projeto [12] e parcialmente

em [13].

O esquema agentes e papéis (Tabela 1) define para cada papel identificado no Diagrama de Organização (Figura 4) as capacidades, o conhecimento e os requisitos dos agentes.

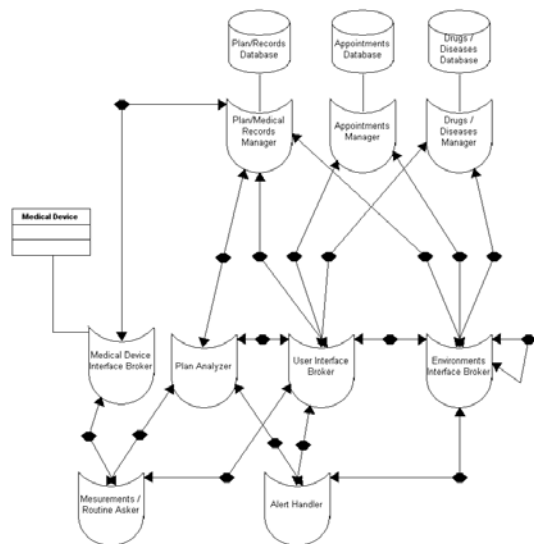


Figura 4 – Diagrama de Organização – Nível 1 – Relacionamento de Conhecimento

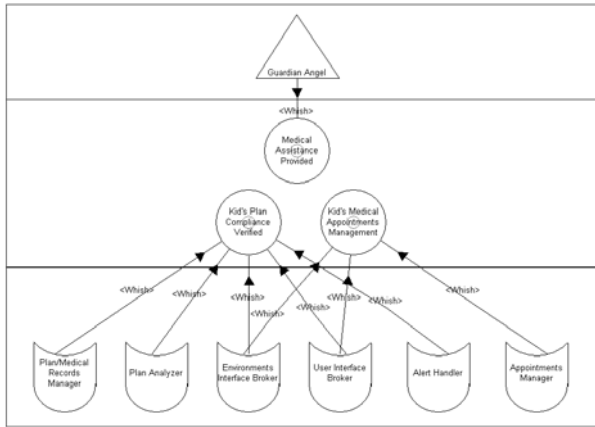


Figura 5 – Diagrama de Delegação – Nível 1

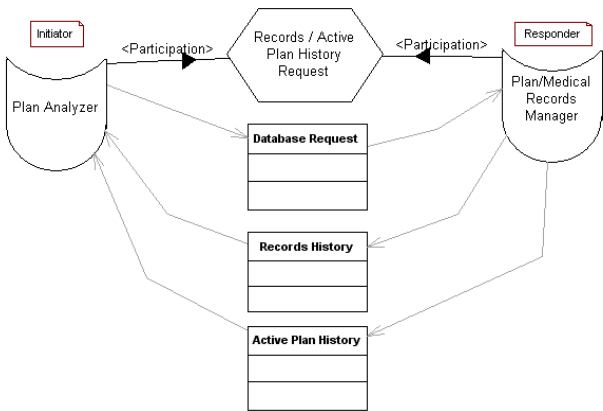


Figura 6 – Diagrama de Interação – Nível 1

Tabela 1 – Esquema Agente/ Papel

Role Schema	Plan Analyzer
Goals	Kids Plan Compliance Verified Therapy Plan Customized Plan Compliance Verified Personal Medical Info Requested Patient's Plan Compliance Verified
Capabilities	Compare collected data with plan's orientations; Identify deviation level; Build possible adapted plans; Send alert request; Check for missing medical info for questions elaboration;
Knowledge, Beliefs	How to make plan's comparisons; Classify the deviation level when existing to request an alert to be sent; How to assemble adapted plans based on comparison's results to satisfy the patient and guarantee his medical treatment success at the same time; Identify the relevant medical data that must be up to date, so future analysis could be correctly done;
Agent Requirements	This role will be played by the Plan Analyzer Agent who will provide plan analysis and control.

3.2. Modelos de Projeto

Na fase de projeto foram modelados somente parte dos modelos propostos, sendo focado principalmente na parte de customização, e definição do plano de tratamento e monitoramento do tratamento.No projeto de alto nível os agentes são identificados e descritos nos diagramas de agentes. Nesses diagramas temos todos os agentes, recursos, serviços e banco de dados que um determinado agente interage. A figura 7 apresenta um exemplo do diagrama de agentes Plan Analyser (Analisador de Plano).

Os serviços são detalhados em workflows nos diagramas de atividades da UML (Figura 8).

Os diagramas de estado definem a interação de comunicação dos papéis (Figura 9).

4. Avaliação da Metodologia MESSAGE

A metodologia MESSAGE [1], [2], [3], [4], [7], [8], foi experimentada em seguida, demonstrando ser de mais fácil compreensão, tendo em vista que sua documentação está bem definida e organizada.

Após a modelagem foi respondido o questionário contido em Yu e Cysneiros [11] para verificar as vantagens e deficiência de metodologias orientadas a agentes.

As maiores vantagens de MESSAGE são a simplicidade e clareza dos seus modelos, além da

abordagem de recursos críticos como as bases de dados e outros programas que podem ser usados pelos agentes do sistema. Entretanto, há a dificuldade no rastreamento de alterações nos requisitos, pois não há uma modelagem destes, pressupondo-se que serão disponibilizados pelo cliente do sistema em uma linguagem informal. A assimilação da MESSAGE é facilitada por estender as notações da UML, já familiares aos profissionais da área,

com conceitos de agentes à nível de conhecimento. O suporte ferramental [14] é excelente, produto de um projeto de uma instituição de pesquisa na área de telecomunicações, mas a versão demo só está disponível por um mês, dificultando o trabalho de modelagem e edição.

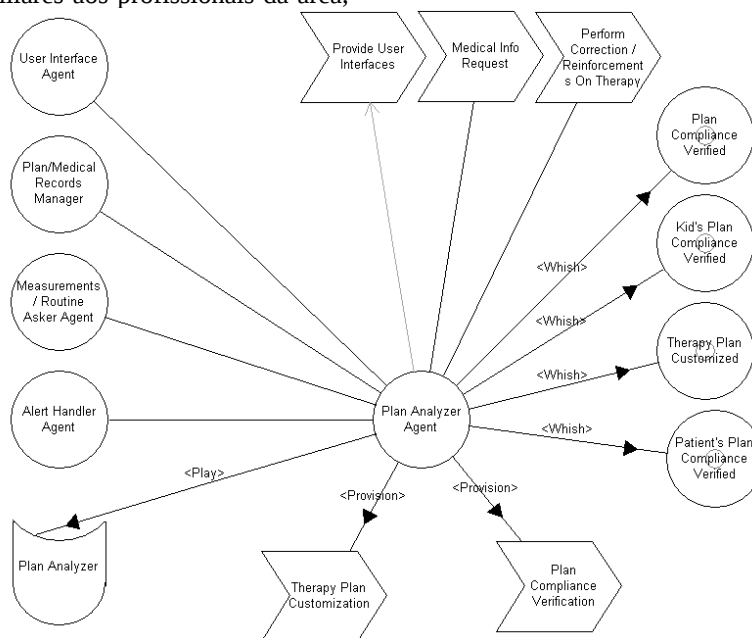


Figura 7 – Diagrama de Agentes do Plan Analyser – Fase de Projeto de Alto Nível

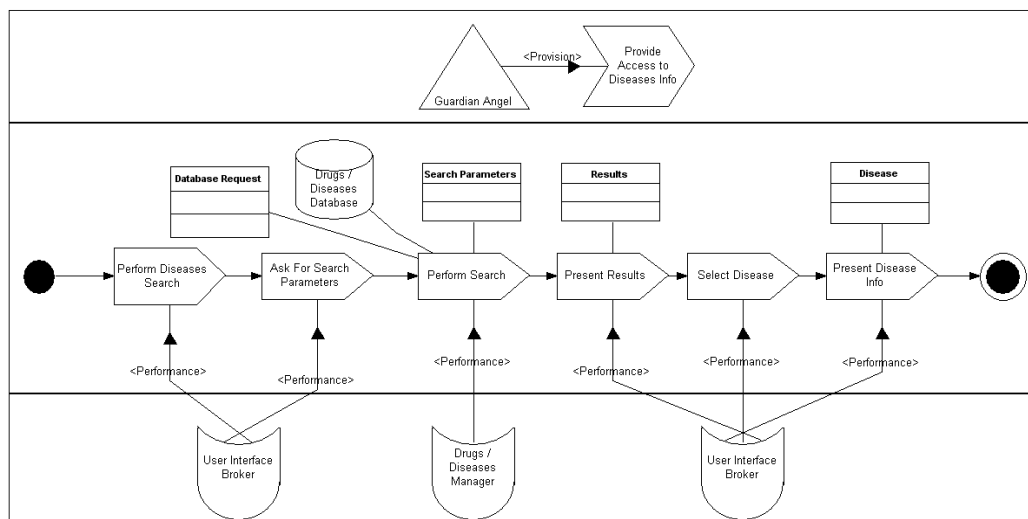


Figura 8 – Workflow do Serviço Provide Access to Diseases Info – Fase de Projeto de Alto Nível

5. Conclusão

A tecnologia de agentes não é um padrão isolado para o desenvolvimento de aplicações, neste caso, baseando-se

em entidades de software autônomas, interativas e adaptativas. A orientação a agentes deve ser integrada com as demais metodologias já existentes, visando explorar aspectos que estas não retratavam, como as características

presentes nas organizações humanas.

Esta abordagem se revela essencial quando há a necessidade de se desenvolver aplicações que requerem um maior nível de inteligência, na execução de tarefas ou disponibilização de serviços. Em relação à orientação a objetos, tecnologia mais usada atualmente, difere pela autonomia. Os agentes dispensam uma intervenção externa para agir, diferentemente dos objetos, que precisam ter seus métodos invocados para efetuar qualquer processamento.

Este novo paradigma ainda está restrito à área de pesquisa e há poucos relatos sobre casos de uso comercial. Além do fato de ser recente, outros fatores contribuem para que a tecnologia não desponte no mercado, tais como a falta de um consenso sobre o suporte à agentes e a existência de várias metodologias com processos de análise e design distintos.

Este trabalho revela a importância do desenvolvimento da tecnologia de agentes e os aspectos relevantes a serem discutidos, a medida que surgem novas necessidades na área de tecnologia de informação. A demanda por sistemas mais inteligentes e flexíveis impulsionará o amadurecimento das metodologias apresentadas, acelerando sua evolução e propiciando a sua utilização na criação de soluções reais para o mercado.

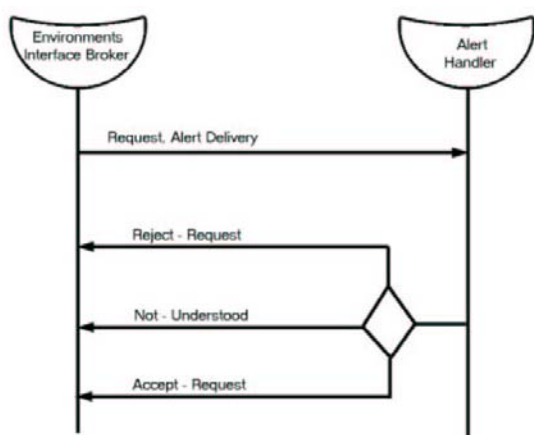


Figura 9 – Protocolo de Interação – Fase de Projeto de Alto Nível

Referências Bibliográficas

- [1] Caire, G., Coulier, W., Garijo, F., Gómez-Sanz, J., Pavón, J., Kearney, P. and Massonet, P., Message: A Methodology for Development of Agent-Based Applications, To appear at Methodologies And Software Engineering For Agent Systems, edited by Federico Bergenti, Marie-Pierre Gleizes and Franco Zambonelli, to be published by Kluwer Academic Publishing (New York), 2004.
- [2] Caire, G., Leal, F., Chainho, P., Evans, R., Gómez-Sanz, J., Pavón, J., Kearney, P., Stark, J. and Massonet, P., Project p907, deliverable 1: Initial methodology. Technical Information Final version, EURESCOM, July 2000.
- [3] Caire, G., Leal, F., Chainho, P., Evans, R., Gómez-Sanz, J., Pavón, J., Kearney, P., Stark, J. and Massonet, P., Agent oriented analysis using MESSAGE/UML. In Michael Wooldridge, Paolo Ciancarini, and Gerhard Weiss, editors, Second International Workshop on Agent-Oriented Software Engineering (AOSE-2001), 2001, pp 101-108.
- [4] Message, <http://www.eurescom.de/~public-webspace/P900-series/P907/index.htm>, May 23, 2000.
- [5] Szolovits, P., Doyle, J., Long, W.J. Guardian Angel: Patient-Centered Health Information Systems, Technical Report.
- [6] RUMBAUGH, J., JACOBSON, I. e BOOCH, G., The Unified Modeling Language Reference Manual, Second edition, Addison-Wesley, 2004.
- [7] Chainho, P., Evans, R., Kearney, P., Massonet, P., Milgrom, R., and Deville, Y., Project p907, deliverable 3: Final guidelines for the identification of relevant problem areas where agent technology is appropriate. Technical Information Final version, EURESCOM, Sept. 2001.
- [8] Evans, R., Kearney, P., Stark, J., Caire, Garijo, F. J., Gómez-Sanz, J., Pavón, J., Leal, F., Chainho, P. and Massonet, P., Project p907, deliverable 3: Methodology for agent-oriented software engineering. Technical Information Final version, EURESCOM, Sept. 2001.
- [9] JADE: Java Agent DEvelopment Framework, available at <<http://jade.cselt.it/>>, 2004.
- [10] FIPA-OS Developers Guide, available at <<http://fipa-os.sourceforge.net/>>, 2004.
- [11] Yu, E., Cysneiros, L.M., "Agent-Oriented Methodologies-Towards a Challenge Exemplar" in Proc of the 4th Intl. Bi-Conference Workshop on Agent-Oriented Information Systems (AOIS 2002) Toronto May 2002
- [12] <http://www.cs.toronto.edu/km/aometh/>
- [13] Coppieters, Andrey Matheus e Marzulo, Leandro Augusto Justen, Um Estudo sobre a Modelagem Orientada a Agentes e um Comparativo entre as Metodologias, Projeto Final de Curso, IME/Departamento de Informática e Ciência da Computação, Universidade do Estado do Rio de Janeiro.
- [14] Caire, G. and Leal, F., Project p907, deliverable 4: Recommendations on supporting tools. Technical Information Final version, EURESCOM, Sept. 2001.