

MAI - UM MODELO DE ARQUITETURA DE AGENTES PARA SISTEMAS DE MÚLTIPLOS DOMÍNIOS DE APLICAÇÃO NA INTERNET

Eliane Moreira Sá de SOUZA, M.Eng

Universidade Federal de Santa Catarina - UFSC
Sociedade Lageana de Educação - SLE - FACIC
Av. Trompowsky, nº 265/701-B. Florianópolis, SC
Brasil, CEP 88015-300
eliane@eps.ufsc.br

Fernando Ostuni GAUTHIER, D.Eng

Universidade Federal de Santa Catarina – UFSC
Campus Universitário, UFSC-CTC
Coordenação da Pós-Graduação em Ciência da Computação
CEP 88015-040
gauthier@eps.ufsc.br

Otávio Roberto Martins de SOUZA, Mestrando Eng

Universidade Federal de Santa Catarina – UFSC
Sociedade Lageana de Educação – SLE – FACIC
Av. Trompowsky, nº 265/701-B. Florianópolis, SC
Brasil, CEP 88015-300
otavio@eps.ufsc.br

ABSTRACT

This paper refers to the architecture of an attendance and advisory generic system at Internet, using the technology of intelligent agents.

The developed model to accomplish such objective is compounded by a society of agents that interact among themselves becoming a multi-agents system.

The proposed architecture for the system is compounded of three classes of agents: Server Agent, that manages the system processes in general; Client Agent that administers each client's process and the Applicative Agents, responsables for fulfilling specific tasks of the advisory domain, made available to the user.

To demonstrate the applicability and functionability of the proposed model, were developed two systems in Java language: APCP - Advisory in Planning and Production Control at Internet and SAIP – Advisory in Identification of Profile Psychologic.

PALAVRAS-CHAVE

Sistemas para Internet; Agentes Inteligentes; Sistemas Multi-Agentes; Assessoria na Internet; Atendimento na Internet; Planejamento e Controle de Produção; Perfil Psicológico.

RESUMO

Este artigo refere-se a arquitetura de um sistema genérico para atendimento e assessoria em múltiplos domínios na Internet, utilizando a tecnologia de agentes inteligentes.

O modelo desenvolvido para tal fim é composto por uma sociedade de agentes que interagem entre si, formando um sistema multi-agentes.

A arquitetura proposta para o sistema é composta por três classes de agentes: Agente Servidor, que gerencia os processos do sistema em geral; Agente Cliente, que administra os processos de cada cliente e Agentes Aplicativos, os quais são responsáveis pela execução das tarefas específicas do domínio da aplicação disponibilizada ao usuário.

Para demonstrar a aplicabilidade e funcionalidade do modelo proposto, foram implementados dois sistemas na linguagem Java: APCP – Assessoria, Planejamento e Controle da Produção na Internet e SAIP – Sistema de Identificação de Perfil Psicológico.

1 INTRODUÇÃO

O cenário dinâmico no qual a sociedade está inserida exige um tratamento rápido e eficiente das informações bem como processos cada vez mais otimizados.

O número crescente destas informações impõe um novo modelo na informática: a computação centralizada na rede Internet. Nele, a computação, comunicação, informação e conhecimento convergem e a rede torna-se o computador.

Com este novo ambiente, o canal para a informação apresenta-se como uma explosão de novas tarefas e serviços, porém sua complexidade exige um novo estilo de interação homem-máquina, onde o computador torna-se um colaborador inteligente, ativo e personalizado [MAE95].

A busca de novas soluções para satisfazer as necessidades dos usuários, intensificou o interesse pela pesquisa e disseminação de um novo paradigma da computação: a tecnologia de agentes inteligentes, a qual objetiva assistir usuários e executar tarefas em favor destes, gerenciando assim a complexidade do ambiente.

Neste novo contexto de utilização do computador, surge por parte das empresas, a necessidade de atendimento e assessoramento ao cliente através da Internet.

Para tanto, apresentamos a modelagem de um sistema multi-agentes, que assessoria clientes e execute tarefas em favor destes, a partir de um servidor e um protocolo de comunicação, sem que haja necessidade de alteração da arquitetura do modelo, quando a implementação de aplicações em outro domínio de conhecimento seja solicitada.

2 CONCEITOS BÁSICOS

Alguns conceitos são importantes para o entendimento da teoria de agentes:

a) Definição de Agente - Agente inteligente pode ser entendido com um *software* que utiliza a comunicação para negociar e coordenar a transferência de informações, agindo autonomamente sobre si próprio ou sobre o ambiente, buscando realizar os objetivos e tarefas para os quais foi designado.

b) Funções dos agentes - O agente é visto como um assistente pessoal do usuário, sendo a ele atribuídas quatro funções: desempenhar tarefas em favor do usuário; treinar ou ensinar o usuário; ajudar diferentes usuários colaboradores e monitorar eventos e processos.

c) Propriedades dos Agentes - O estudo das propriedades apresentadas pelos agentes são de grande importância para o desenvolvimento deste paradigma, pois são responsáveis pelo comportamento que os agentes apresentam. Entre as propriedades relacionadas na literatura, selecionamos abaixo aquelas que consideramos mais importantes:

Autonomia - é a habilidade apresentada pelo agente de executar o controle sobre suas próprias ações, permitindo-lhe operar isoladamente e decidir sobre o que fazer enquanto opera neste estado;

Mobilidade - é a capacidade do agente de transportar-se de um ambiente a outro para executar suas tarefas;

Comunicabilidade - é a capacidade do agente de comunicar-se com outros agentes ou usuários para troca de informações relacionadas ao mundo real, às preferências e aos objetivos a conquistar;

Discurso - é a característica que propicia um *feedback* em dois sentidos entre usuário e agente, possibilitando um maior conhecimento sobre suas habilidades e intenções.

Inteligência - é a capacidade do agente de aprender e desenvolver raciocínio, habilitando-o a negociar com ambigüidades e determinar a ação mais adequada a uma situação apresentada; a inteligência está associada à diversas propriedades, podendo ser verificada na aprendizagem, comunicação, mobilidade e outras;

Reatividade - é a propriedade que permite aos agentes perceberem seus ambientes e responderem adequadamente às mudanças neles ocorridas;

Habilidade Social - é a característica dos agentes que os habilita a interagir no momento adequado com outros agentes ou pessoas, objetivando concluir suas tarefas ou ajudar outros agentes;

Pró-Atividade - é a capacidade do agente de exibir um comportamento direcionado a objetivos;

Persistência - é a propriedade apresentada pelo agente de manter um estado interno conciso e adequado através do tempo, sem alterá-lo ao acaso;

Cooperatividade - é a habilidade dos agentes de trabalharem juntos para concluírem tarefas mutuamente benéficas e complexas;

Comportamento Adaptativo - é a capacidade do agente de modificar seu comportamento em função de experiências anteriores;

Aprendizagem - é a habilidade apresentada pelo agente de acumular conhecimento baseado em experiências anteriores, e conseqüentemente, modificar seu comportamento em resposta à novas situações [BEL95].

d) Agência - Denomina-se agência ao conjunto de características que definem um software como agente e o diferencia das demais aplicações.

Na análise das definições de agência verificadas na literatura identifica-se propriedades imprescindíveis à existência dos agentes, denominadas propriedades fundamentais; e propriedades desejáveis, denominadas auxiliares, que alteram o comportamento destes tornando-os mais ou menos robustos e complexos.

Wooldridge e Jennings [WOO96] definem como atributos fundamentais da agência: a autonomia, a habilidade social, a capacidade de resposta e a pró-atividade; Coen [COE94] define a agência através das características fundamentais: persistência, autonomia, reatividade e comunicabilidade; e através das características auxiliares: pró-atividade, mobilidade, raciocínio, aprendizado, comportamento adaptativo e

habilidade para planejar; Nissen [NIS95] utiliza seis características fundamentais: autonomia, comunicabilidade, cooperatividade, raciocínio, comportamento adaptativo e confiabilidade.

e) Sistemas Multi-Agentes - Um ambiente multi-agente pode ser definido como um sistema no qual diversos agentes interagem. Este conjunto de agentes muitas vezes apresenta-se heterogêneo com respeito a alguns aspectos, tais como: conhecimento, critério de avaliação de sucesso, objetivos, linguagens, algoritmos, hardware e requisitos. Nos sistemas multi-agentes o foco principal consiste em coordenar o comportamento inteligente dos agentes autônomos integrantes, objetivando solucionar um problema apresentado

3 MODELO PROPOSTO

O modelo proposto consiste na arquitetura geral de um sistema multi-agentes, que disponibilize e gerencie atendimento e assessoria em um domínio específico para usuários conectados remotamente a um servidor na Internet.

A arquitetura desenvolvida utiliza uma comunidade de agentes que colaboram e comunicam-se entre si, de tal forma a permitir a execução dos processos necessários aos clientes e ao próprio servidor.

Neste ambiente multi-agentes, o foco principal reside em coordenar os diversos agentes nas tarefas cuja realização se façam necessárias, a partir de solicitação dos clientes, ou a partir da determinação dos próprios agentes.

A ênfase do Modelo de Atendimento e Assessoria na Internet (MAI) está voltada para gerenciamento, arquitetura e comunicação do sistema.

3.1 Gerenciamento do Sistema

O modelo proposto apresenta um gerenciamento dinâmico do sistema, onde os agentes controladores desempenham multi-tarefas, executando processos paralelos, minimizando desta forma o tempo de espera dos usuários.

Outro aspecto referente ao gerenciamento é a característica de descentralização do controle, o qual é distribuído entre dois agentes principais: um agente gerenciador dos aspectos gerais do sistema, residente no servidor, e outro, gerenciador dos processos locais executados no cliente.

3.2 Arquitetura

A arquitetura do MAI apresenta três tipos de agentes, classificados conforme as funções que desempenham:

AgServer - agente gerenciador do processo geral do sistema;

AgClient - agente gerenciador dos processos disponíveis para o cliente;

AgAplicativos - agentes executores de aplicações específicas.

O sistema consiste na execução contínua do *AgServer*, residente em *background* no servidor, bem como na execução do *AgClient*, cujo processamento é ativado quando um cliente acessa a *Home Page* do sistema. Os agentes componentes do MAI detêm objetivos distintos e preestabelecidos apresentando, em diferentes graus, as propriedades de autonomia, comunicabilidade, representatividade, inteligência e mobilidade. O modelo apresenta uma arquitetura projetada para permitir a inclusão de módulos aplicativos, sem que estes alterem os processos gerais existentes ou exijam novos controles para o processamento.

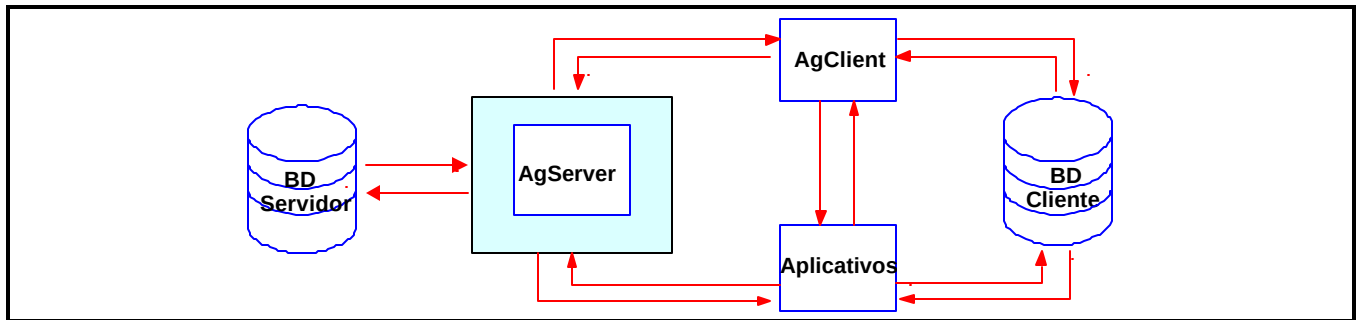


Figura 3.1 - Arquitetura geral MAI

A arquitetura dos agentes componentes do MAI é formada por diversos módulos conforme tabela abaixo:

	AgServer	AgClient	AgAplicativos
MGM: Módulo Gerenciador de Memória e Mensagens	X	X	X
MGP: Módulo Gerenciador de Processos	X	X	X
MQN: Módulo de Estrutura de Quadro Negro	X	X	X
MIL: Módulo de Interface com Usuário	X	X	X
MGA: Módulo Gerenciador de Arquivos	X		X
MGQ: Módulo Gerenciador do Quadro Negro	X		
PCA: Protocolo de Comunicação entre Agentes	X	X	X
TCP: Protocolo TCP/IP	X	X	X

MGM - tem por objetivo administrar as mensagens gravadas no Quadro Negro e efetuar o gerenciamento dinâmico dos recursos de memória;

MGP - é responsável pela seleção dos procedimentos a serem executados;

MGA - tem por função consultar e atualizar bases de dados residentes no servidor e no ambiente do cliente;

MIU - busca informar as conexões e processos ativos, bem como possibilitar o acesso à informações necessárias à execução do sistema;

MQN - armazena as mensagens emitidas pelos agentes. Associada ao módulo MGQ, possibilita o gerenciamento de processos por cliente.

MGQ - tem como meta a verificação constante das mensagens contidas na estrutura e seleção daquelas endereçadas ao AgServer, possibilitando porém o processo paralelo dos outros módulos pertencentes ao agente;

PCA - tem como processos principais a codificação e decodificação de mensagens, permitindo assim a comunicação entre agentes;

TCP - permite a troca de mensagens entre os agentes bem como a transferência destes disponibilizando-os no ambiente do cliente.

3.3 Agente AgServer

O agente AgServer é um agente estático, cujo principal objetivo é o controle geral dos processos ativos no sistema. O grau de inteligência deste agente é função do paradigma para inferência a ele atribuído, podendo ser de baixo nível, se for constituído de regras fixas, ou de médio a alto nível, se for constituído de algum paradigma da Inteligência Artificial (IA), como por exemplo, sistemas especialistas, redes neurais ou sistemas

híbridos. A arquitetura do AgServer é composta por todos os módulos projetados no modelo, conforme é apresentado na figura abaixo.

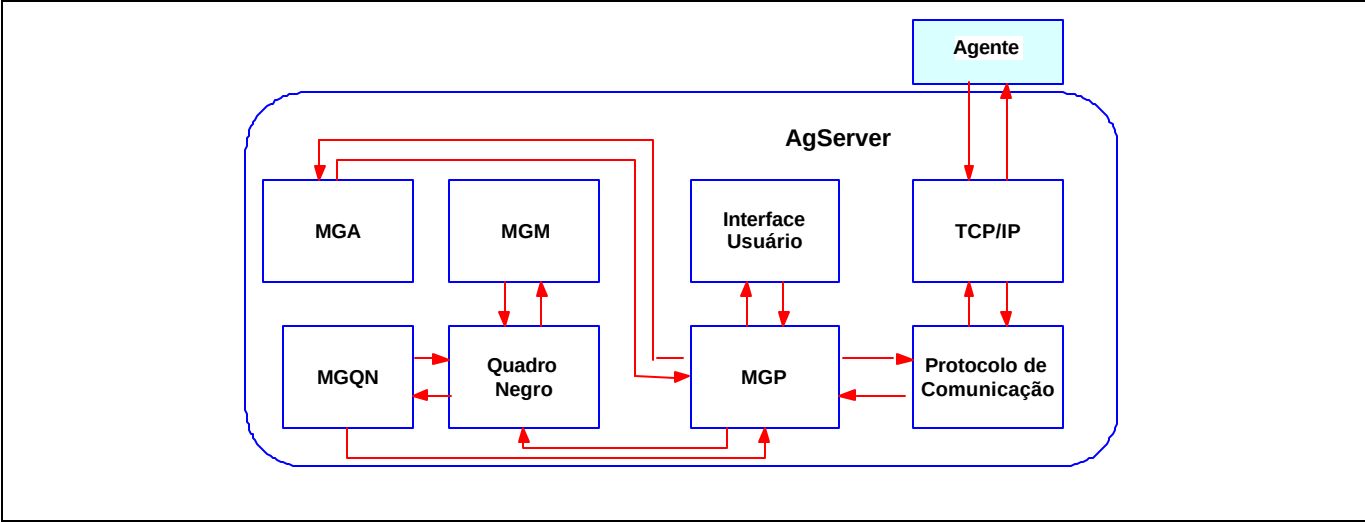


Figura 3. - Arquitetura do agente AgServer

3.4 Agente AgClient

O AgClient é um agente móvel, centralizador do controle dos processos disparados para cada cliente, o qual é transferido do servidor à máquina do cliente, para executar tarefas a ele atribuídas. Sua arquitetura, conforme fig. 3, o caracteriza como um agente que poderá atingir alto nível de inteligência em função dos requisitos e técnicas de IA utilizadas em sua implementação, bem como um agente pró-ativo, visto que identifica e disponibiliza tarefas para execução sem exigir a intervenção do usuário para o qual atua.

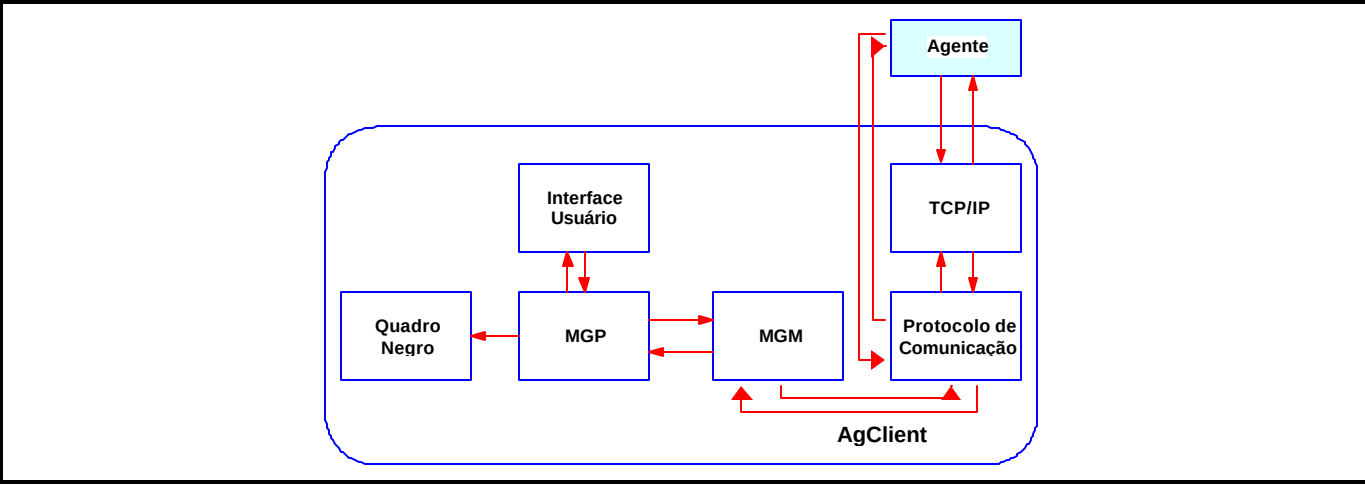


Figura 4. - Arquitetura do Agente AgClient

O módulo MGM apresenta duas funções básicas: criticar e selecionar as informações recebidas para enviá-las ao MGP, e agrupar e enviar as informações necessárias ao desempenho de outros processos, bem como repassá-las ao protocolo de comunicação.

O MGP deste agente tem por função principal identificar os processos a serem executados para o cliente e solicitar ao AgServer autorização para processamento.

No MQN são armazenadas mensagens para os demais agentes, incluindo-se entre elas, aquelas referentes ao estado dos processos em andamento, os quais são constantemente atualizados.

3.5 Agentes Aplicativos

Os Agentes Aplicativos executam as tarefas específicas do domínio da assessoria implantada, sendo ativados quando o AgClient os considerar necessário ou quando houver interferência direta do cliente ou da empresa. Estes agentes também apresentam a propriedade de mobilidade, pois são enviados à memória da máquina do cliente quando solicitada sua execução. Da mesma forma apresentada para os agentes AgServer e AgClient, os AgAplicativos podem apresentar grau variado de inteligência, conforme sua especificação.

Na arquitetura apresentada na fig. 4, os módulos MIU e MQN manipulam informações de acordo com a tarefa específica atribuída ao agente Aplicativo. Por outro lado, a existência do Módulo de Gerenciamento de Arquivos está condicionada à necessidade ou não de armazenamento e recuperação de informações em arquivos de dados.

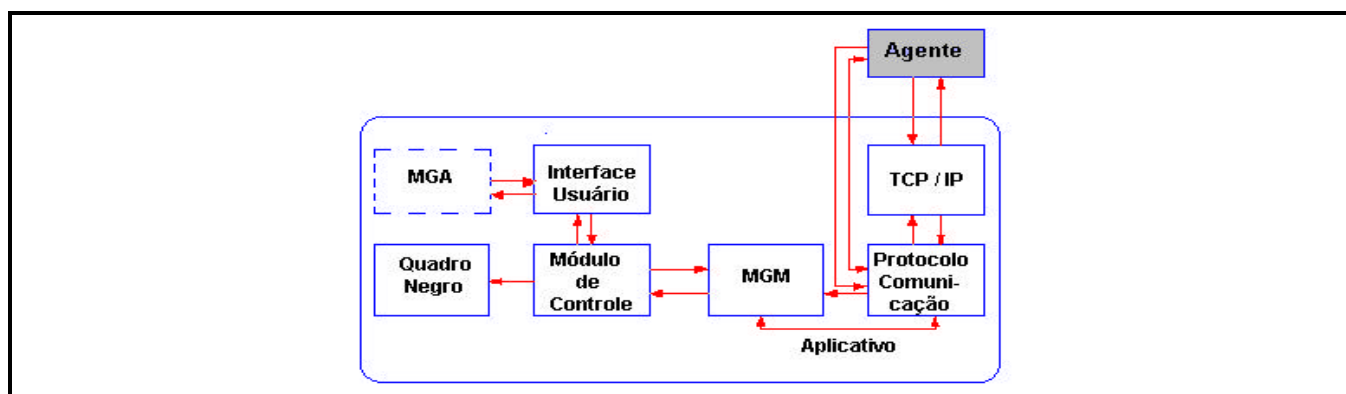


Figura 4. Arquitetura de um Agente Aplicativo

3.6 Comunicação

O MCI apresenta uma estrutura de comunicação direta e indireta entre os agentes, com padrão próprio reconhecido pelo protocolo de comunicação do sistema.

Os agentes integrantes do modelo comunicam-se através de mensagens, utilizando tanto a comunicação direta, através de troca de mensagens, quanto a comunicação indireta, por meio de uma arquitetura de Quadro Negro, residente no servidor.

3.7 Estrutura da Mensagem

Conforme demonstrado na fig. 5, a estrutura da mensagem proposta para comunicação direta entre os agentes é composta de uma parte obrigatória e outra opcional, a qual informa o texto da mensagem. Os itens

obrigatórios componentes da mensagem são:

Identificação da Mensagem, composta por três caracteres que identificam as mensagens do sistema, evitando que informações alheias sejam processadas;

Identificação do Emissor, composto pelo código do agente emissor da mensagem;

Identificação do Destinatário, o qual informa o código do agente a que se destina a mensagem;

Evento ocorrido, que informa o motivo da mensagem.



Figura 5 - Estrutura de mensagem

3.8. Estrutura de Quadro Negro

A estrutura de Quadro Negro, cujo esquema é apresentado na figura 6, foi projetada em níveis hierárquicos para cada cliente, de tal forma a permitir o gerenciamento dos diferentes processos dos usuários.

As mensagens armazenadas na estrutura de Quadro Negro possuem, além de informações pertinentes à conexão *socket* de cada cliente, outros dados, tais como: agente emissor, agente destinatário, evento, endereço IP do agente emissor e informações não padronizadas componentes das mensagens.

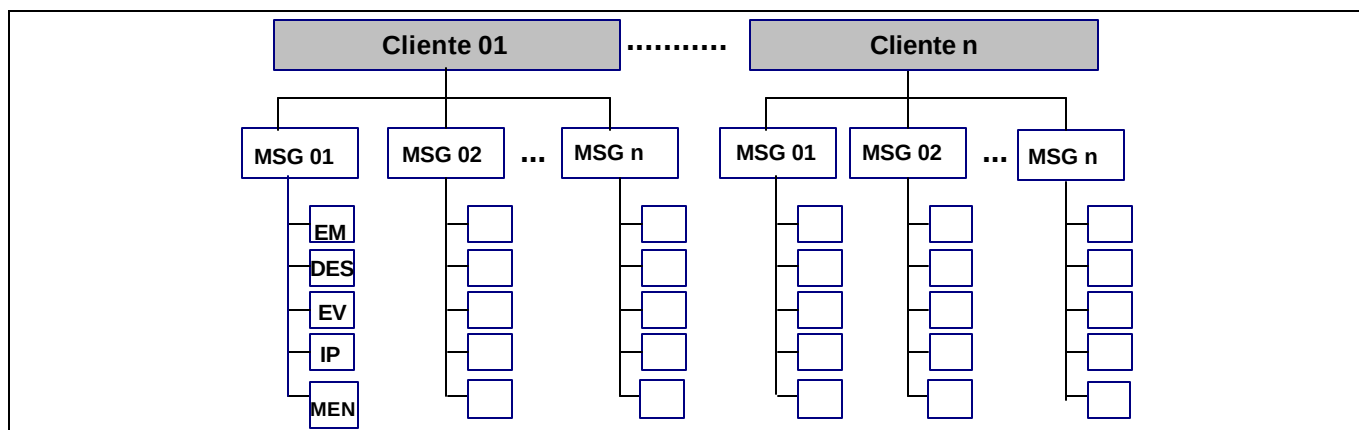


Figura 6 - Estrutura de Quadro Negro

4 APLICAÇÃO PRÁTICA

A aplicabilidade da arquitetura MAI foi testada em Sistemas de Assessoria em Planejamento e Controle da Produção – APCP e Assessoramento em Identificação de Perfil Psicológico - SAIP, ambos desenvolvidos com o objetivo de controlar e processar as tarefas requisitadas pelos clientes através da Internet.

4.1 Ambiente de Desenvolvimento do Sistema

No desenvolvimento dos agentes dos sistemas desenvolvidos, utilizou-se a linguagem Java em virtude de algumas características nela presentes, principalmente aquelas relacionadas à adequação da linguagem às aplicações na rede Internet, como portabilidade, processamento distribuído e tratamento de protocolos.

Sendo a Internet uma rede heterogênea onde plataformas diferenciadas comunicam-se através de protocolos apropriados, a portabilidade intrínseca à linguagem foi um dos principais fatores para a escolha do ambiente Java. Da mesma forma, como as aplicações desenvolvidas caracterizam-se por serem sistemas distribuídos, exigindo alta capacidade de compartilhamento entre os objetos (remotos ou locais) do sistema, a linguagem Java também apresentou-se adequada à esta implementação. Outro fator decisório na escolha do ambiente de desenvolvimento foi a facilidade de tratamento de protocolos através desta linguagem, visto que os sistemas baseiam-se em conexões *sockets*, utilizando o protocolo TCP/IP.

4.2 Implementação do Sistema

Na implementação do MAI nos sistemas SAIP e APCP demonstrou-se que a partir da estrutura formada pelos Agentes AgServer, AgClient e pela Estrutura de Mensagens, diferentes aplicações poderiam ser acopladas aos sistemas, através dos AgAplicativos, contribuindo para maior satisfação das necessidades das empresas e de seus clientes.

Para tanto, os agentes AgServer e AgClient foram programados com as seguintes características:

- O AgServer configura-se como um Servidor Socket que aguarda conexões na porta lógica 100, controlando vários clientes simultâneos, cada qual podendo manter ativo lotes de mensagens na estrutura de Quadro Negro;
- O AgClient desenvolvido conecta-se com o AgServer através de *socket* e mantém ativo o sistema de mensagens, controlando o desempenho dos agentes aplicativos.

Os agentes aplicativos do sistema SAIP foram implementados em sua integridade, possuindo diferentes graus de inteligência, de acordo com as tarefas a eles designadas. Integram o sistema SAIP, entre outros agentes aplicativos:

- Agente Orientador Vocacional;
- Agente Identificador de Perfil Personológico;
- Agente Identificador de Capacidade de Memorização;
- Agente Identificador de Capacidade de Raciocínio Lógico;
- Agente Demonstração e
- Agente Informação.

Os agentes aplicativos do sistema APCP foram prototipados com o objetivo de demonstrar a aplicabilidade destes na Internet e a facilidade para adição de novas aplicações ao sistema. Entre os agentes aplicativos sugeridos para integrar o sistema foram desenvolvidos:

- Agente Informação, o qual fornece informações relativas aos requisitos necessários para utilização do sistema;
- Agente Demonstração, cuja função é demonstrar aos usuários os recursos do sistema APCP;
- Agente Cadastro, o qual atualiza o cadastro dos clientes na base de dados da empresa;
- Agente Plano Mestre, cuja função é calcular e fornecer o Plano Mestre de Produção;
- Agente Lista de Materiais, que dimensiona e disponibiliza a Lista de Materiais.

5 CONCLUSÕES

No estudo realizado sobre a literatura relativa a agentes inteligentes verificamos a inexistência de um consenso universal sobre as propriedades que caracterizam a agência. Definimos como exigibilidade de propriedades

para caracterizar o comportamento adequado de um agente, as propriedades fundamentais - autonomia, comunicabilidade e reatividade e as propriedades auxiliares - mobilidade, aprendizagem e cooperatividade. A arquitetura do Modelo de Atendimento e Assessoria na Internet - MAI, demonstrou ser confiável e aplicável em ambientes complexos, apresentando resultados satisfatórios, visto que os processos podem ser executados de forma interativa com o compartilhamento de recursos, independentemente de plataforma de hardware disponível no cliente.

Os sistemas desenvolvidos, abrangendo os domínios específicos de conhecimento (psicologia e planejamento e controle de produção), revelaram-se de bom desempenho, demonstrando a eficiência do gerenciamento dinâmico da arquitetura MAI.

Outro aspecto positivo constatado no modelo proposto é o fato deste adequar-se a múltiplos domínios, permitindo a reutilização de código de seus módulos, sendo necessário codificar apenas os agentes aplicativos pertinentes ao domínio de conhecimento específico de novas aplicações a serem implementadas.

Para trabalhos futuros, recomendamos o desenvolvimento de sistemas em outros domínios bem como o desenvolvimento dos agentes aplicativos do Sistema de Assessoria em Planejamento e Controle da Produção, na plenitude de suas funções, e aqueles que se façam necessários para eficácia do software, como por exemplo: Agente Plano Mestre Nivelado; Agente Estimativa de Vendas; Agente Negociador para Compras e Agente Controlador de Processos da Linha de Produção.

6 BIBLIOGRAFIA

- [AUE95] AUER, Karl. *Agents*. URL:<http://www.pcug.org.au/~kauer/project/main.htm>, 1995.
- [BEL95] BELGRAVE, Marc. *The Unified Agent Architecture. A White Paper*. URL:http://www.ee.mcgill.ca/~belmarc/uaa_paper.html, 1995.
- [CAZ97] CAZELLA, S.C. Uma Arquitetura para Coordenar a Interação de Agentes na Internet. Porto Alegre: Centro de Pós-Graduação em Ciência da Computação – UFRGS, 1997. Dissertação de Mestrado, 1997.
- [COE94] COEN, Michael. *Agency Defined*. URL:http://www.ai.mit.edu/people/mhcoen/agents/section2_1_2.html, 1994.
- [COE95] COEN, Michael. *SodaBot Agent*. URL:<http://www.ai.mit.edu/people/sodabot/>, 1995.
- [FON94] FONER, Lenny. *What's an Agent. Crucial Notions*. URL:http://foner.www.media.mit.edu/people/foner/Julia/subsection3_4_1.html, 1994.
- [FRA96] FRANKLIN, Stan, GRAESSER, *Is it an Agent, or just a Program?: A Taxonomy for Autonomous Agents*. URL:<http://www.msci.memphis.edu/~franklin/AgentProg.html>. 1996.
- [GIL96] GILBERT, Don, APARICIO; MANNY, A; *Intelligent Agent Strategic*. URL:<http://activist.gpl.ibm.com:81/WhitePaper/ptc2.html>, 1996.
- [LIZ90] LIZOTTE, M; MOULIN, B. *A Temporal Planner for Modelling Autonomous Agents*. In: *Decentralized A.I.*, 1990.
- [LUC95] LUCK, Michael, D'INVERNO, Mark. *A Formal Framework for Agency and Autonomy* In: ICMAS-95, 1995.
- [MAE95] MAES, Pattie. *Artificial Life Meets Entertainment: Life like Autonomous Agents*. Communications of ACM, p. 108, 1995.
- [NIS95] NISSEN, Mark. *Intelligent Agents: A Technology and Business Application Analysis*, URL: <http://haas.berkeley.edu/~heilmann/agents/1995>, 1995.

- [RIV96] RIVER, Charles. *Intelligent Agents*.
URL:<http://www.opensesame.com/webpages/corp/background.html>, 1996.
- [SHO93] SHOHAN, Yoav. Agent-Oriented Programing. *Artificial Inteligent* 60: p. 51-92, 1993.
- [SMI94] SMITH, D.C, A; CYPHER, J.; SPOHRER. *KidSim: Programming **Agents** Without a Programming Language*. Communications of ACM, 1994.
- [SUL00] SULLIVAN, Joe. Na introduction to Intelligent Agents.
URL:<http://www.ilstu.edu/~jrsulli/352paper/intelligent.htm>, 2000.
- [WOO94] WOOLDRIDGE, Michael, JENNINGS, Nick. *Intelligent Agents: Theory and Practice*,
URL:http://www.doc.mmu.ac.uk/STAFF/mike/ker95/bibliography3_38.html, 1994.
- [WOO96] WOOLDRIDGE, Michael, JENNINGS *Software agents*. IEE Review, 1996.