

UNIVERSIDADE FEDERAL DE SANTA CATARINA
PROGRAMA DE PÓS-GRADUAÇÃO EM CIÊNCIA DA
COMPUTAÇÃO

MICHEL BARNAD GESZYCHTER

Modelagem Multiagente Escalável para a Integração
de Sistemas Especialistas

Dissertação submetida à Universidade Federal de Santa Catarina como parte dos requisitos para a obtenção do grau de Mestre em Ciência da Computação

Fernando Ostuni Gauthier, Dr.
Orientador

Florianópolis, fevereiro de 2003

Modelagem Multiagente Escalável para a Integração de Sistemas Especialistas

MICHEL BARNAD GESZYCHTER

Esta Dissertação foi julgada adequada para a obtenção do título de **Mestre em Ciências da Computação na Área de Sistemas de Conhecimento** e aprovada em sua forma final pelo Programa de Pós-Graduação em Ciência da Computação.

Banca Examinadora

Fernando Álvaro Ostuni Gauthier, Dr.
Coordenador do Curso

Fernando Álvaro Ostuni Gauthier, Dr.
Orientador

Silvia Modesto Nassar, Dr^a.
Co-orientadora

Maria Marlene de Souza Pires, Dr^a.

AGRADECIMENTOS

*Ao Professor Fernando Ostuni Gauthier,
pela oportunidade de realizar este mestrado e, principalmente,
pela orientação e incentivo durante todo o desenvolvimento deste trabalho.*

*Às Professoras Sílvia Modesto Nassar e Maria Marlene de Souza Pires,
pela co-orientação e pelo apoio dado com muita dedicação, interesse e preocupação,
mostrando-se sempre dispostas a ajudarem-me a desenvolver este trabalho.*

*Agradeço, em especial, a Lisiane, minha futura esposa,
que sempre esteve presente com seu amor incondicional,
mesmo quando este trabalho nos privou do convívio.
Seu apoio só fez aumentar o amor que sinto por ela.*

*A minha mãe e ao meu pai,
pelo estímulo e compreensão nos momentos difíceis,
fundamentais para a conclusão deste trabalho,
e pelas lições de vida que me dão todos os dias.*

E a Deus pelo Dom da Vida ...

RESUMO

Este trabalho consiste no desenvolvimento de uma modelagem multiagente escalável para a integração de sistemas especialistas. Foi desenvolvido como protótipo, um sistema multiagente denominado CLIDENP – Clínica Virtual para Diagnóstico e Educação em Nutrição Pediátrica, que integrou os sistemas especialistas probabilísticos que realizam o diagnóstico da desnutrição infantil pertencentes ao mesmo domínio de conhecimento e à mesma linha de pesquisa.

O objetivo principal da CLIDENP é permitir o cálculo das hipóteses diagnósticas do estado nutricional e sugerir a dieta alimentar de crianças com até dois anos de idade, além de disponibilizar um ambiente de diagnóstico e terapia nutricional, e de ensino-aprendizagem em desnutrição infantil. Todos estes processos são realizados pelos Agentes Especialistas implementados através de suas interações durante a execução do Sistema Multiagente CLIDENP.

Na integração deste conjunto de sistemas especialistas legados em forma de um sistema multiagente, foi concebido um modelo multiagente que tem como objetivo principal à manutenção da escalabilidade do sistema, ou seja, a capacidade do sistema de crescer tanto em número de agentes já existentes no sistema, quanto na inserção de novos agentes a posteriori.

Além disso, este trabalho visa atender a uma demanda muito grande existente nos ambientes de trabalho e acadêmico, que é a de propor uma modelagem capaz de fazer com que os sistemas possam trocar informações e serviços entre si, resolvendo problemas computacionais de forma distribuída, organizada em grupos de trabalho e aplicada a problemas que não podem ser resolvidos isoladamente, devido a sua complexidade ou necessidade computacional.

PALAVRAS-CHAVE: Inteligência Artificial, Sistemas Multiagentes, Sistemas Especialistas, Agentes Inteligentes, Clínica Virtual.

ABSTRACT

This work is about the development of a multiagent modeling for the integration of specialists systems. A multiagent system called CLIDENP – Clínica Virtual para Diagnóstico e Educação em Nutrição Pediátrica (Virtual Clinic for Diagnosis and Education in Pediatric Nutrition) was developed as a prototype, integrating the probabilists specialists systems who carry through the diagnosis of the infantile malnutrition pertaining to the same domain of knowledge and the same line of research.

The CLIDENP's core objective is to allow the calculation of the diagnostic hypotheses of the nutritional state and the alimentary diet of children with up to two years of age, developing and publishing an environment of diagnosis and nutritional therapy, teaching in infantile malnutrition. All these processes are carried through by the implemented Specialists Agents, through its interactions during the execution of the Multiagent System CLIDENP.

In the integration of this set of legacies specialists systems in form of a multiagent system, a multiagent model was conceived that has as main objective the maintenance of the capacity of the system to grow in such a way in number of existing agents already in the system, as in the insertion of new agents a after this work.

Moreover, this work aims to take care of a cery great existing demand in several work and academic environments, who is to consider a modeling capable to make the systems able to change information and services between themselves, solving computational problems in a distributed way, organized in work groups and applied to problems that cannot be solved separately, because its complexity or computational need.

KEY WORDS: *Artificial Inteligence, Multiagents Systems, Specialist Systems, Inteligent Agents, Virtual Clinic.*

ÍNDICE

LISTA DE FIGURAS	IX
LISTA DE ABREVIATURAS	X
1 INTRODUÇÃO	1
1.1. OBJETIVOS	3
1.1.1. GERAL	3
1.1.2. ESPECÍFICOS	4
1.2. MOTIVAÇÃO	4
1.3. ORGANIZAÇÃO	6
2 FUNDAMENTAÇÃO TEÓRICA	8
2.1 INFORMÁTICA MÉDICA	8
2.2 A INTERNET NA SAÚDE	12
2.3 INTELIGÊNCIA ARTIFICIAL	15
2.3.1 INTELIGÊNCIA ARTIFICIAL SIMBÓLICA	15
2.3.2 INTELIGÊNCIA ARTIFICIAL CONEXIONISTA	16
2.3.3 INTELIGÊNCIA ARTIFICIAL DISTRIBUÍDA	16
2.3.3.1 RESOLUÇÃO DISTRIBUÍDA DE PROBLEMAS	20
2.4 SISTEMAS MULTIAGENTE	21
2.4.1 AGENTES	24
2.4.1.1 AGENTES MÓVEIS	27
2.4.1.2 AGENTES DELIBERATIVOS OU SIMBÓLICOS	28
2.4.1.3 AGENTES REATIVOS	28
2.4.1.4 AGENTES COLABORATIVOS	29
2.4.1.5 AGENTES COLABORATIVOS EVOLUTIVOS	29
2.4.1.6 AGENTES DE INTERFACE	29
2.4.1.7 AGENTES INTELIGENTES	30
2.4.2 COMUNICAÇÃO ENTRE AGENTES	31
2.4.3 CLASSIFICAÇÕES DE MAS	34
2.4.3.1 SISTEMAS MULTIAGENTE FEDERADOS	35
2.4.3.2 SISTEMAS MULTIAGENTE DEMOCRÁTICOS	35
2.4.3.3 SISTEMAS MULTIAGENTE ABERTOS	36
2.5 SISTEMAS ESPECIALISTAS	36
2.5.1 COMPONENTES	38

2.5.2 REPRESENTAÇÃO DAS INCERTEZAS	39
2.5.3 SISTEMAS ESPECIALISTAS PROBABILÍSTICOS	41
3 O SISTEMA MULTIAGENTE PROPOSTO: CLIDENP	43
3.1 DOMÍNIO DE CONHECIMENTO	43
3.2 Os SISTEMAS LEGADOS ENVOLVIDOS	43
3.2.1 SISPAN	44
3.2.1.1 DEFINIÇÃO	44
3.2.1.2 CARACTERÍSTICAS TÉCNICAS	44
3.2.1.3 BASE DE CONHECIMENTO	45
3.2.2 SACI	46
3.2.2.1 DEFINIÇÃO	46
3.2.2.2 CARACTERÍSTICAS TÉCNICAS	47
3.2.2.3 BASE DE CONHECIMENTO	47
3.2.3 SANEP	48
3.2.3.1 DEFINIÇÃO	48
3.2.3.2 CARACTERÍSTICAS TÉCNICAS	49
3.2.3.3 BASE DE CONHECIMENTO	49
3.2.4 SANPAP	50
3.2.4.1 DEFINIÇÃO	50
3.3 MODELO DE AGENTES CLIDENP	51
4 IMPLEMENTAÇÃO DA CLIDENP	57
4.1 MODELO FUNCIONAL	59
4.2 MODELO DE IMPLEMENTAÇÃO	65
4.2.1 O AGENTE PADRÃO	66
4.2.2 O AGENTE GERENTE	68
4.2.3 Os AGENTES ESPECIALISTAS	69
4.2.4 O AGENTE INTERFACE	71
4.2.5 A BASE DE DADOS CLIDENP	73
4.3 CICLOS DO SISTEMA MULTIAGENTE	74
4.3.1 LANÇAMENTO DO SISTEMA	75
4.3.2 INTERAÇÃO ENTRE OS AGENTES	76
4.3.3 ENCERRAMENTO DO SISTEMA	77
4.4 ESCALABILIDADE DO SISTEMA MULTIAGENTE	78
4.4.1 INTEGRANDO NOVOS AGENTES ESPECIALISTAS À CLIDENP	78
5 CONCLUSÕES	81

5.1 TRABALHOS FUTUROS	83
6 REFERÊNCIAS BIBLIOGRÁFICAS	85
ANEXO A: LINGUAGEM DE COMUNICAÇÃO DOS AGENTES	91
ANEXO B: MODELO DE ENTIDADE E RELACIONAMENTO DA BASE DE DADOS CLIDENP	93
ANEXO C: MODELOS DE OBJETOS DA CLIDENP	94

LISTA DE FIGURAS

<i>FIGURA 1 – ARQUITETURA DA CLIDENP</i>	2
<i>FIGURA 2 - ETAPAS DA DPS</i>	20
<i>FIGURA 3 – FLUXO DO PROJETO DE UM DPS</i>	21
<i>FIGURA 4 - SISTEMA MULTIAGENTE</i>	22
<i>FIGURA 5 – FLUXO DO PROJETO DE UM MAS</i>	23
<i>FIGURA 6 - TIPOLOGIA DE AGENTES</i>	27
<i>FIGURA 7 - SISTEMA MULTIAGENTE FEDERADO</i>	35
<i>FIGURA 8 - SISTEMA MULTIAGENTE ABERTO</i>	36
<i>FIGURA 9 - COMPONENTES DE UM SISTEMA ESPECIALISTA</i>	38
<i>FIGURA 10 – INTERFACE SISPAN</i>	44
<i>FIGURA 11 – REPRESENTAÇÃO GRÁFICA DA BASE DE CONHECIMENTO DO SISPAN</i>	45
<i>FIGURA 12 – INTERFACE SACI</i>	47
<i>FIGURA 13 – REPRESENTAÇÃO GRÁFICA DA BASE DE CONHECIMENTO DO SACI</i>	48
<i>FIGURA 14 – INTERFACE SANEP</i>	49
<i>FIGURA 15 – REPRESENTAÇÃO GRÁFICA DA BASE DE CONHECIMENTO DO SANEP</i>	50
<i>FIGURA 16 - ENQUADRAMENTO DOS AGENTES CLIDENP</i>	52
<i>FIGURA 17 – MODELO DE AGENTES DA CLIDENP</i>	54
<i>FIGURA 18 – UTILIZAÇÃO DA CLIDENP VIA INTERNET</i>	56
<i>FIGURA 19 - ETAPAS PARA O ATENDIMENTO/DIAGNÓSTICO MÉDICO(IATROS)</i>	58
<i>FIGURA 20 – MODELO FUNCIONAL DAS ENTIDADES FUNDAMENTAIS CLIDENP</i>	61
<i>FIGURA 21 – MODELO FUNCIONAL CLIDENP</i>	63
<i>FIGURA 22 – COMUNIDADES LÓGICAS DE AGENTES DA CLIDENP</i>	64
<i>FIGURA 23 – MODELO FUNCIONAL DA COMUNICAÇÃO DOS AGENTES CLIDENP</i>	65
<i>FIGURA 24 – MODELO DE OBJETOS DO AGENTE PADRÃO</i>	68
<i>FIGURA 25 – MODELO DE OBJETOS DO AGENTE GERENTE</i>	69
<i>FIGURA 26 – MODELO DE OBJETOS DOS AGENTES ESPECIALISTAS</i>	71
<i>FIGURA 27 – INTERFACE DO AGENTE INTERFACE</i>	72
<i>FIGURA 28 – DIAGRAMA DE FLUXO DE DADOS DO AGENTE INTERFACE</i>	73
<i>FIGURA 29 – BASE DE DADOS CLIDENP</i>	74

LISTA DE ABREVIATURAS

[A]

AOP – *Agent Oriented Programing*

API – *Application Program Interface*

[C]

CME – *Continued Medical Education*

CRM – Conselho Regional de Medicina

[D]

DEP – Desnutrição Energético Protéica

DPS – *Distributed Problem Solving*

[F]

FC – Falência no Crescimento

FTP – *File Transfer Protocol*

[G]

GUI – *Graphical User Interface*

[H]

HON – *Health On the Net Foundation*

HTTP – *HyperText Transfer Protocol*

[I]

IA – Inteligência Artificial

IAC – Inteligência Artificial Conexionista

IAD – Inteligência Artificial Distribuída

IAM – Inteligência Artificial em Medicina

IAS – Inteligência Artificial

[K]

KQML – *Knowledge Query and Manipulation Language*

[L]

LCA – Linguagem para Comunicação de Agentes

[M]

MAS – *Multiagent Systems*

[N]

NE – Nutrição Enteral

NP – Nutrição Parenteral

[O]

OOP – *Oriented Objects Programming*

[P]

PHP – *Pre Hypertext Processor*

[R]

RB – Redes Bayesianas

[S]

SACI – Sistema de Apoio na Avaliação de Distúrbios do Crescimento Infantil

SANEP – Sistema Especialista Probabilístico de Apoio a Nutrição Enteral Pediátrica

SE – Sistema Especialista

SEP – Sistema Especialista Probabilístico

SISPAN – Sistema Pediátrico para Avaliação Nutricional

[T]

TCP/IP – *Transfer Control Protocol / Internet Protocol*

[U]

URL – *Uniform Resource Locator*

[W]

Web – WWW

WWW – *World Wide Web*

1 INTRODUÇÃO

As pesquisas no universo de desenvolvimento de software tem se desenvolvido em um ritmo muito acelerado nos últimos anos. Contudo, se por um lado é possível observar que existem *softwares* construídos para as mais diversas áreas de conhecimento; por outro, ainda existe uma demanda muito grande por programas que possam trocar informações e serviços entre si. Essa demanda decorre da necessidade da resolução de problemas, ser praticada de forma distribuída, organizada em grupos de trabalho e aplicada a problemas que não poderiam ser resolvidos isoladamente. Esta parece ser a tendência decorrente das novas necessidades das empresas, das pesquisas das áreas de Sistemas Multiagentes e Sistemas Distribuídos e da globalização porque passa o planeta.

Levando em conta todas as tendências apresentadas, este trabalho, irá demonstrar como é possível fazer uma modelagem para a integração de sistemas, para a troca de informações e serviços em prol da solução de problemas que não podem ser resolvidos isoladamente, utilizando-se para tanto de três sistemas especialistas probabilísticos, objetos de trabalhos anteriores, que realizam diagnósticos distintos e ao mesmo tempo complementares entre si, e da engenharia de software aliada ao conceito de Sistemas Multiagentes.

Assim, esta modelagem gerará a CLIDENP – Clínica Virtual para Diagnóstico e Educação em Nutrição Pediátrica que, quando completa, vai permitir o cálculo das hipóteses diagnósticas do estado nutricional e da dieta alimentar de crianças com até dois anos de idade. Além de disponibilizar um ambiente de diagnóstico e terapia nutricional, a CLIDENP permitirá também o ensino-aprendizagem em desnutrição infantil via Internet.

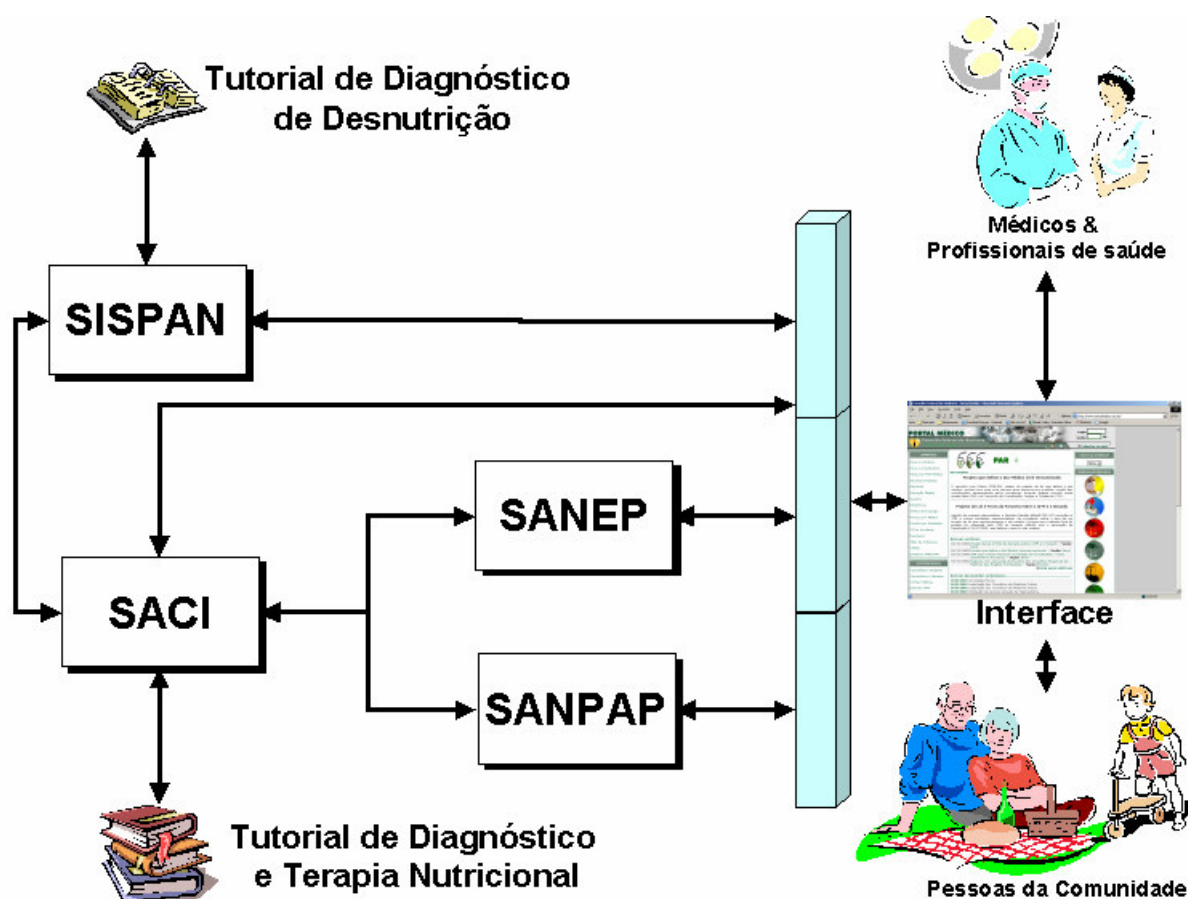


Figura 1 – Arquitetura da CLIDENP

Dentre os conceitos existentes na inteligência artificial distribuída (IAD), escolheu-se a abordagem de sistemas multiagentes (MAS) para realizar a interligação de todos os módulos envolvidos na CLIDENP. Esta abordagem permite a modelagem dos módulos em forma de agentes, a comunicação entre os mesmos através de uma linguagem de comunicação padronizada e o relacionamento previamente concebido entre as entidades (agentes), via arquitetura do sistema multiagente. A IAD ainda permite que estes vários processos (ou sistemas) autônomos realizem atos de inteligência global, somente através de processamento local e comunicação interprocessos. O objetivo na utilização de sistemas especialistas baseados em conhecimentos e distribuídos é poder coordenar atividades de sistemas especialistas resolvendo problemas que os sistemas compartilham totalmente ou em parte.

Esta evolução da inteligência artificial, ou seja, a distribuição do processamento e unificação das respostas, tem proporcionado resultados satisfatórios quanto à aplicabilidade das técnicas inteligentes, no meio científico, tecnológico, como também em aplicações voltadas à área médica, econômica, política e social, entre outros.

No caso da área médica, dentre os principais modelos de inteligência artificial existentes e utilizados, destacam-se os sistemas especialistas probabilísticos, que trabalham com a incerteza num domínio de conhecimento, por meio da distribuição de probabilidades condicionais em redes bayesianas. Sistemas como estes, têm sido aplicados com sucesso na área médica, uma vez que trabalham com a incerteza presente nas informações e no raciocínio médico.

Através da utilização de todos estes conceitos, esta pesquisa se propõe desenvolver uma modelagem multiagente para integrar sistemas especialistas que poderá auxiliar as equipes de suporte nutricional, fazendo com que o profissional utilize melhor o seu tempo, preocupando-se mais com a fisiopatologia das doenças e com o diagnóstico. Espera-se com esse estudo auxiliar no diagnóstico e educação em nutrição pediátrica, de forma a permitir o apoio ao diagnóstico e tratamento de desnutrição infantil, propiciando também apoio informatizado ao ensino em nutrição pediátrica nos cursos de Medicina ou outros afins.

É muito importante salientar que a *Internet* é a maior aliada na divulgação e disponibilização tanto da pesquisa quanto do sistema, aos profissionais da área médica. A *Internet* é base para todo o desenvolvimento, e serão utilizados os conceitos e tecnologias disponíveis para otimizar o *site* e o sistema multiagente da CLIDENP.

1.1. OBJETIVOS

1.1.1. GERAL

O objetivo geral desta pesquisa é o desenvolvimento de uma modelagem multiagente escalável para a integração de sistemas especialistas. Esta modelagem será desenvolvida

através da prototipação, que consistirá em transformar os sistemas especialistas em diagnóstico e educação em nutrição pediátrica SISPAN (Sistema Pediátrico de Avaliação Nutricional), SACI (Sistema para Avaliação de Distúrbios do Crescimento Infantil), SANEP (Sistema de Apoio à Nutrição Enteral Pediátrica) e SANPAP (Sistema de Apoio à Nutrição Parenteral Pediátrica) previamente desenvolvidos em Agentes Especialistas da comunidade de agentes do Sistema Multiagente CLIDENP – Clínica Virtual de Diagnóstico e Educação em Nutrição Pediátrica.

1.1.2. ESPECÍFICOS

Este trabalho possui os seguintes objetivos específicos:

- ◆ Investigar as técnicas e ferramentas disponíveis para implementação de sistemas multiagentes na plataforma Web;
- ◆ Integrar ao ambiente multiagente os agentes especialistas probabilísticos SISPAN, SACI, SANEP e SANPAP;
- ◆ Projetar e desenvolver o protótipo de um *site*, ou “clínica virtual”, como interface principal de acesso;
- ◆ Evidenciar a aplicabilidade da utilização de sistemas multiagentes e sistemas especialistas em problemas do mundo real;
- ◆ Avaliar qualitativamente o protótipo desenvolvido, junto a especialistas da área.

1.2. MOTIVAÇÃO

Doenças relacionadas aos distúrbios do crescimento infantil, em crianças com até dois anos, são denominadas “Doenças Silenciosas”, uma vez que é difícil diagnosticá-las, pois as crianças nesta faixa etária, mesmo mantendo ou aumentando seu peso poderão não estar saudáveis [SIMÕES, 01].

Inicialmente, o que motivou a realização deste trabalho foi o tema, o fato de poder contribuir com a obtenção do diagnóstico de doenças relacionadas com a desnutrição infantil, através da construção de uma ferramenta de apoio ao diagnóstico.

Um diagnóstico mais preciso e veloz é a busca eterna dos médicos, e a ciência da computação surge como aliada ao fornecer cada vez mais soluções e ferramentas que auxiliem os médicos em seu trabalho. No caso de pacientes crianças, o diagnóstico veloz torna-se difícil, pois existem muitos fatores variantes em cada caso. Na questão da precisão, somente o médico pode determiná-la.

A maioria dos médicos pediatras costuma acompanhar a evolução do crescimento infantil utilizando uma tabela onde são colocados os dados referentes a peso e altura, conforme a idade das crianças. O diagnóstico quando realizado dessa forma é deficiente, pois não permite visualizar a desaceleração (casos leves) ou falência (casos graves) do crescimento infantil. Por meio dessa forma de acompanhamento do crescimento infantil, só é possível diagnosticar algum distúrbio quando estiver em seu grau mais grave, ou seja, no momento em que a criança pára totalmente de crescer ou começa a perder peso (PIRES, 1997).

Assim, as ferramentas de apoio ao diagnóstico são importantíssimas, pois utilizam o processamento da máquina para cálculos tornando o trabalho do médico mais rápido, resultando em um diagnóstico mais veloz.

Do ponto de vista computacional, com a utilização dos conceitos de sistemas multiagentes em conjunto com o conceito de sistemas especialistas probabilísticos, tem-se o tratamento da incerteza realizado de forma distribuída. Isto proporcionaria mais rapidez na obtenção do resultado, pois ocorre uma divisão do problema a ser resolvido em diversos problemas menores e uma conseqüentemente uma maior especialização dos agentes envolvidos.

Além disso, este trabalho visa atender a uma demanda muito grande existente nos ambientes de trabalho e acadêmico que é a de propor uma modelagem capaz de fazer com que os sistemas possam trocar informações e serviços entre si, resolvendo problemas computacionais de forma distribuída, organizada em grupos de trabalho e aplicada a problemas que não podem ser resolvidos isoladamente, devido a sua complexidade ou necessidade computacional.

Desta forma, a grande motivação deste trabalho, será o de modelar um sistema computacional capaz de resolver problemas de forma distribuída utilizando todos os conceitos da Engenharia de Software, da Inteligência Artificial e de Sistemas Especialistas e Multiagentes. Todos estes conceitos da Ciência da Computação aliados, com o foco na integração de sistemas legados, informações e serviços, poderá trazer melhorias na forma de desenvolver novos sistemas no futuro.

Ao integrar estas tecnologias emergentes, conforme proposto, o profissional da saúde poderá utilizar melhor o seu tempo, preocupando-se mais com as doenças e com o diagnóstico. Espera-se com esse estudo auxiliar no diagnóstico e educação em nutrição pediátrica, de forma a permitir o apoio ao diagnóstico e tratamento de desnutrição infantil.

Por fim, com o auxílio da *Internet*, ter-se-á um alcance muito maior e facilitará, em termos de interface, tanto a utilização da ferramenta quanto o ensino-aprendizagem de desnutrição infantil.

1.3. ORGANIZAÇÃO

Este relatório encontra-se organizado em cinco capítulos principais: Introdução, Fundamentação Teórica, O Sistema Proposto: CLIDENP, Implementação da CLIDENP e Conclusões.

Esta organização visa seguir a seguinte seqüência lógica de apresentação do contexto e do relatório: inicialmente, no Capítulo de Introdução, são apresentados os objetivos (gerais e específicos) e as motivações da realização desta pesquisa. No Capítulo da Fundamentação Teórica são apresentados os principais tópicos relacionados ao tema desta pesquisa, que constituem o embasamento teórico necessário, tais como: Inteligência Artificial, Sistemas Multiagentes e Sistemas Especialistas.

A seguir, no Capítulo do Sistema Proposto: CLIDENP é apresentada a proposta de desenvolvimento objeto deste trabalho. Além disso, no capítulo da Implementação da CLIDENP é apresentada a metodologia e as etapas de desenvolvimento da ferramenta proposta, bem como os modelos funcionais e de implementação do protótipo. Finalmente, os resultados finais obtidos são avaliados e comentados na conclusão juntamente com as propostas de trabalhos futuros.

Como anexo ao relatório, são apresentados: a Linguagem de Comunicação dos Agentes, o Modelo de Entidade e Relacionamento da Base de Dados e o Modelo de Objetos da CLIDENP.

2 FUNDAMENTAÇÃO TEÓRICA

Neste capítulo, será apresentada uma revisão sobre Informática Médica, Inteligência Artificial, Sistemas Multiagente e Sistemas Especialistas, podendo-se verificar a aplicabilidade de conceitos como a Inteligência Artificial e Sistemas Multiagente em problemas do mundo real. É importante ressaltar também que o ponto-chave para a tomada de decisão é a utilização da Inteligência Artificial, o que também demonstra as precauções e cuidados necessários ao desenvolvimento dos Sistemas Especialistas.

2.1 INFORMÁTICA MÉDICA

A Saúde é uma das áreas onde há maior necessidade de informação para a tomada de decisões. A Informática Médica é o campo científico que lida com recursos, dispositivos e métodos para otimizar o armazenamento, recuperação e gerenciamento de informações biomédicas.

O crescimento da Informática Médica como uma disciplina deve-se, em grande parte: aos avanços nas tecnologias de computação e comunicação, à crescente convicção de que o conhecimento médico e as informações sobre os pacientes não são gerenciáveis por métodos tradicionais baseados em papel, e devido à certeza de que os processos de acesso ao conhecimento e tomada de decisão desempenham papel central na Medicina moderna [HOGARTH, 98].

A Informática Médica ou Informática em Saúde (em Inglês *Medical Informatics*) é definida por Blois & Shortliffe como "um campo de rápido desenvolvimento científico que lida com armazenamento, recuperação e uso da informação, dados e conhecimento biomédicos para a resolução de problemas e tomada de decisão" [BLOIS, 90].

Esta nova disciplina concentra-se principalmente em prover mecanismos para desenvolver computacionalmente processos existentes. Assim as seguintes áreas de atuação estão intimamente ligadas a ela: [HOGARTH, 98]

- ◆ Sistemas de Informação em Saúde;
- ◆ Prontuário Eletrônico do Paciente;
- ◆ Telemedicina;
- ◆ Sistemas de Apoio à Decisão;
- ◆ Processamento de sinais biológicos;
- ◆ Processamento de Imagens Médicas;
- ◆ *Internet* em Saúde;
- ◆ Padronização da Informação em Saúde.

O principal objetivo da Informática Médica atualmente é o estudo e desenvolvimento de metodologias e sistemas computacionais (em *hardware* e / ou *software*) para o apoio as atividades médicas em situações reais, para atender as necessidades sociais através da transferência de tecnologia para o setor industrial da medicina, sendo o campo da Ciência de Computação preocupada em analisar e disseminar dados médicos através de aplicação de computadores a vários aspectos da Saúde e Medicina [SABBATINI, 98].

As origens da disciplina geral de informática podem ser traçadas a uma publicação russa intitulada *Oznovy Informatiki* (Fundamentos da Informática), publicada em 1968. Na época, foi descrita como um conceito de ciência da informação, no contexto de uma era emergente de computação [HOGARTH, 98].

Com a formalização da disciplina em 1974, a informática médica passou a ser crescentemente reconhecida como um componente importante da prática global de medicina. Surgiram muitos centros acadêmicos e de pesquisa de informática médica, com o interesse de se desenvolver sistemas de registro médico computadorizado que incorporassem princípios básicos de projeto determinados pela informática médica [HOGARTH, 98].

Outras áreas que deram um grande ímpeto à Informática Médica foram a das aplicações da Inteligência Artificial à medicina, a qual se iniciou com o desenvolvimento dos primeiros sistemas especialistas de apoio à decisão, como o MYCIN, desenvolvido pelo Dr. Edward Shortliffe [SHORTLIFFE, 90], de Stanford, em 1974; e a das aplicações da

computação no ensino médico, exemplificado pelos primeiros sistemas de avaliação formativa, pelos tutoriais eletrônicos e pelas simulações de casos clínicos, como os desenvolvidos pelo Dr. A. Octo Barnett, na Universidade Harvard, que também foi o inventor do MUMPS (*Massachusetts General Hospital Utility for Multiprogramming Systems*) em 1972, uma linguagem de programação multi-usuária voltada aos bancos de dados médicos, que teve um papel fundamental nos primeiros sistemas de informação hospitalar e laboratorial. No mesmo período de tempo ocorreu um forte programa de desenvolvimento da informática médica nos Estados Unidos, financiado em grande parte pela Biblioteca Nacional de Medicina Americana [HOGARTH, 98].

A informática aplicada à medicina entrou no Brasil com um certo atraso em relação aos EUA e Europa. No início da década de 70, teve início simultaneamente em alguns centros universitários, principalmente no Hospital da Universidade Federal do Rio de Janeiro (UFRJ), no Instituto do Coração e nos Hospitais das Clínicas da USP em São Paulo e Ribeirão Preto. Um professor da UFRJ, Luiz Carlos Lobo, trouxe na época o MUMPS para o Brasil e fundou o Núcleo de Tecnologia de Educação em Saúde, que iniciou pioneiramente a aplicação de minicomputadores Digital PDP-11 em sistemas de apoio ao ensino. [SABBATINI, 98] Na mesma época, começaram as atividades em Engenharia Biomédica na UFSC, com a criação do Grupo de Pesquisas em Engenharia Biomédica (GPEB), atual IEB-UFSC, oficialmente em 1974 [IEB-UFSC, 02].

O desenvolvimento da Informática em Saúde brasileira deu um grande salto a partir de 1983, com a criação de novos grupos especificamente dedicados a esta área de pesquisa e ensino. O divisor de águas da Informática em Saúde nacional ocorreu em 1986. O primeiro reconhecimento do grau de desenvolvimento nacional na área deu-se em um seminário realizado em Informática em Saúde em Brasília, por iniciativa do Ministério da Saúde. Os pesquisadores presentes resolveram então se organizar e fundaram em novembro de 1986 a Sociedade Brasileira de Informática em Saúde, durante o I Congresso Brasileiro de Informática em Saúde. Esta sociedade tornou-se o foco principal da atividade profissional brasileira na área, organizando sucessivamente diversos congressos nacionais, regionais e especializados. Conjuntamente com outras

sociedades irmãs, entre as quais a Sociedade Brasileira de Engenharia Biomédica, participa desde então do Fórum Nacional de Ciência e Tecnologia em Saúde (FNCTS) [SABBATINI, 98].

Em 1988 o governo federal, através do CNPq, SEI e outros, efetuou um estudo, envolvendo dezenas de colaboradores, visando um Plano Nacional de Desenvolvimento da Informática em Saúde (que, infelizmente, acabou engavetado). A SBIS produziu também o primeiro "Quem é Quem na Informática em Saúde no Brasil". Atualmente a SBIS e vários centros de pesquisa desenvolvem programas de colaboração com o MS (DATASUS) visando padronização de componentes e linguagens, estabelecimento do Cartão de Saúde e outros [SABBATINI, 98].

Com relação às publicações, a primeira revista do gênero foi fundada em 1986, e se intitulava "Revista Brasileira de Informática em Saúde". Foi editada até 1988 e reiniciada brevemente alguns anos depois. Em 1993, o Núcleo de Informática Biomédica deu início à revista *Informédica*, publicada durante dois anos e meio com o apoio dos Laboratórios Biosintética, e que reiniciou publicação em janeiro de 1998, com o novo nome de Revista "Informática Médica" [SABBATINI, 98].

Na UFSC, o IEB-UFSC edita a Revista On-Line IEB-UFSC, esta revista eletrônica possibilita informar, com maior rapidez e menor limitação de espaço, o público crescentemente interessado em Tecnologia para a Saúde. A Revista On-Line edita os resultados obtidos nas pesquisas mais recentes da área [IEB-UFSC, 02].

Hoje, existem disponíveis uma série de *Sites* na *Internet* com diversos recursos para os internautas como notícias, fóruns, *chats*¹ com médicos, tira-dúvidas por *e-mail*, bibliotecas, testes. A interatividade é o grande trunfo de boa parte dos *sites* de saúde da rede. Os grandes portais de saúde funcionam quase como um *check-up*. Dicionários de doenças e de sintomas, especialistas que respondem perguntas, dicas para uma vida saudável, fóruns com gente que sofre dos mesmos males, não tem fim à lista de serviços oferecidos pelos *sites* e portais especializados em saúde. Vídeos, sons e animações

¹ Comunicação em tempo real entre dois usuários através da *Internet*.

descomplicam a medicina e ajudam qualquer pessoa a entender como seu corpo funciona [REIS, 01].

2.2 A INTERNET NA SAÚDE

Assistiu-se nos últimos anos a uma grande revolução no mundo da informação: a *Internet*. A grande rede mundial de computadores, termo bastante comum atualmente, tem transformado o dia-a-dia das pessoas e, principalmente, das empresas. A presença da *Internet* mudou a forma de fazer negócios: antes se dizia que a informação era essencial para a empresa, hoje ela é o "negócio".

Nos Estados Unidos, segundo o *Healthcare Financing Administration* gasta-se, anualmente,

cerca de um trilhão de dólares no mercado de saúde, com a previsão de passar para mais de dois trilhões a partir de 2007. A saúde tem um mercado altamente distribuído, que precisa compartilhar informações, e isso é feito atualmente, na maioria das vezes, de forma manual, principalmente no Brasil. Em países desenvolvidos como os Estados Unidos e o Japão, o percentual de hospitais informatizados é de 98%, enquanto no Brasil este percentual é de apenas 4% [REIS, 01].

A prática médica diária necessita, invariavelmente, da troca de informação, ou apenas da informação propriamente dita, para funcionar. Sem isso, o atendimento ao paciente torna-se muito difícil, por vezes, impossível. Por causa disto, a saúde tornou-se uma indústria altamente dependente dos sistemas de informação, devido à necessidade da melhoria de seus processos, objetivando melhor atender os pacientes.

Hoje os *sites* de medicina na *Internet*, preocupam-se basicamente em atender as necessidades das seguintes classes de internautas: [COSTA, 01]

a) Pacientes:

- ◆ Querem melhorar sua educação e serem capazes de atuar mais ativamente no controle de sua saúde (*Patient Empowerment*);
- ◆ Precisam se comunicar melhor com seus médicos.

b) Médicos:

- ◆ Necessitam de informações para se manterem atualizados e também melhorarem o gerenciamento de seus “negócios”;
- ◆ Precisam se comunicar com outros médicos, com laboratórios, farmácias e planos de saúde, para troca de informações sobre os pacientes, reembolso, etc.

c) Planos de Saúde:

- ◆ Precisam de informações para gerenciar os custos e a qualidade do atendimento;
- ◆ Querem se comunicar com os seus prestadores para reembolso e gerenciamento;
- ◆ Prestadores, distribuidores e indústrias precisam se comunicar para controlar o fluxo de compra/venda.

Disso tudo, o principal componente explorado foi, sem dúvida, o conteúdo. A explosão de *sites* de saúde na *Web* foi espantosa. Há *sites* voltados tanto para o paciente como para o profissional, oferecendo recursos como artigos, notícias, dicas de saúde, acesso a sistemas de apoio à decisão, *Webmail* e outros serviços mais especializados voltados ao profissional. Com isso, surgiu também outra grande revolução: a educação à distância. Com a *Internet*, novas formas de aprendizado foram aparecendo e diversos *sites* se especializaram no oferecimento de cursos, no processo chamado de Educação Médica Continuada (*Continued Medical Education, CME*) [COSTA, 01].

Toda essa revolução fez com que os conselhos e associações médicas, bem como entidades independentes, como a *Health On the Net Foundation (HON)*, procurassem normatizar e regulamentar de certa forma, o comportamento e o conteúdo dos *sites* de saúde, criando códigos de conduta e princípios. Esses códigos ditam algumas regras que os *sites* devem seguir para não ferir a ética médica, protegendo, principalmente, os pacientes que usufruem desses serviços e que muitas vezes fazem uso dessas informações para decidir as suas atitudes relacionadas à saúde [HON, 02].

Não só na *Internet* é possível presenciar o crescimento da importância da informática aplicada à medicina. Inclusive os profissionais da medicina estão começando a considerar a informática como um aliado nas seguintes questões:

- ◆ na procura de um atendimento mais preciso, através dos sistemas e informações disponíveis em *sites* médicos;
- ◆ no aperfeiçoamento profissional mais rápido e fácil, através do *e-learning*² e dos artigos publicados nos *sites* médicos;
- ◆ no compartilhamento de informações relevantes ao atendimento, tratamento de outros pacientes por outros profissionais da saúde, disponibilizando na *Internet* novas técnicas e tratamentos para os mais diversos tipos de doenças;
- ◆ Criação de grandes bases de conhecimento nas universidades, hospitais, ambulatórios, consultórios através da tecnologia da *Internet* utilizada como Intranet dessas instituições, entre outras.

Ao disponibilizar estes serviços na grande rede, popularizariza-se a medicina, tornando-a mais acessível principalmente para as famílias que desejam obter uma maior qualidade de vida em casa, sem a necessidade da consulta constante a profissionais da saúde.

Por causa de tudo isso, a Ciência da Computação tem um papel importantíssimo, relacionado diretamente com o trabalho médico. Nem os *sites* na *Internet* nem os sistemas especialistas, já existentes ou que venham a ser desenvolvidos, poderão substituir o trabalho do médico. Toda e qualquer ferramenta computacional poderá ser muito útil no treinamento de estudantes e na educação continuada dos profissionais em exercícios diagnósticos e desenvolvimento do raciocínio lógico, importante na prática clínica. A Ciência da Computação jamais poderá tomar o lugar do médico ou de qualquer profissional da saúde. Isto simplesmente porque o paciente é um ser humano, e deve ser examinado do ponto de vista psicológico, histórico, clínico, de forma unilateral; e só o médico tem a capacidade de fazê-lo.

Por fim, é importante salientar que, para muitos profissionais da saúde, os sistemas de informação são essenciais apenas para as tarefas administrativas do hospital, da clínica ou do consultório e, por isso, a intenção deste trabalho é também mostrar a estes profissionais que os sistemas de informação podem ser muito mais utilizados por eles, trazendo mais benefícios em seu trabalho e no atendimento aos pacientes.

² Treinamento on-line através da tecnologia disponível na *Internet*

2.3 INTELIGÊNCIA ARTIFICIAL

O objetivo da Inteligência Artificial (IA) é representar o comportamento humano utilizando modelos computacionais. Este comportamento está fundamentado em dois temas: o conhecimento e a aprendizagem. O conhecimento é sua principal matéria prima, podendo ser representada por muitos formalismos. A aprendizagem também tem despertado o interesse da comunidade de Inteligência Artificial e apresenta-se constantemente em todos os temas relacionados a IA, pois é uma forma dos sistemas evoluírem e, portanto, de se construírem sistemas inteligentes.

Uma definição conveniente e totalmente aceita para Inteligência Artificial, vem sendo procurada e, apesar de existirem muitas definições, ainda não foi encontrada. Uma evidência deste fato são os vários conceitos encontrados: [BARRETO, 01]

- ◆ para Patrick Winston *“Inteligência Artificial é o estudo de conceitos que permitem os computadores serem inteligentes”*;
- ◆ para Elaine Rich e Kevin Knight *“Inteligência Artificial é o estudo de como fazer os computadores realizarem coisas que, no momento, as pessoas fazem melhor”*;
- ◆ para Eugene Charniak e Drew McDermott *“Inteligência Artificial é o estudo das faculdades mentais com o uso de modelos computacionais”*.

Analisando as definições citadas, observa-se que a primeira define IA utilizando o termo inteligente, o que causa recursividade por não esclarecer inteligência. Já a segunda definição condena a IA a nunca conseguir um sucesso, pois cada vez que for resolvido um problema ele deixa de ser do domínio de IA [BARRETO, 01]. A definição mais convincente dentre as citadas é a terceira, pois não apresenta recursividade, relaciona-se com outras ciências (por exemplo, psicologia, lógica e fisiologia) no que diz respeito às faculdades mentais e, além disso, se mostra independente de arquiteturas computacionais, podendo ser adotada tanto por computadores digitais como por neurocomputadores [REIS, 01].

2.3.1 INTELIGÊNCIA ARTIFICIAL SIMBÓLICA

Na Inteligência Artificial Simbólica (IAS) o comportamento inteligente é simulado, baseado nos princípios da psicologia cognitiva. Na IAS se espera que um conhecimento sobre o problema particular a resolver e das técnicas úteis para o caso possam levar a manipulando conhecimentos básicos e imitando, ao menos até certo ponto, o modo de raciocínio usado por humanos na solução do problema se obtenha a solução desejada [REIS, 97].

Para isto, a IAS deve ser utilizada quando: (a) o domínio do problema é bem definido; (b) quando se tem idéia de como o problema seria resolvido; e (c) quando for explícito o modo de achar uma solução.

2.3.2 INTELIGÊNCIA ARTIFICIAL CONEXIONISTA

A Inteligência Artificial Conexionista (IAC) acredita na construção de neurocomputadores e se estes forem bastante parecidos ao cérebro humano, eles apresentarão um comportamento inteligente. Se apenas uma parte pequena do cérebro for reproduzida, emergirá a função que a IAC realiza [REIS, 97].

A IAC vem atraindo a atenção de pesquisadores, pois apresenta um melhor desempenho quando o problema não é bem definido (falta de conhecimento explícito de como realizar a tarefa).

2.3.3 INTELIGÊNCIA ARTIFICIAL DISTRIBUÍDA

A Inteligência Artificial Distribuída (IAD) é o estudo do comportamento computacionalmente inteligente, resultantes da interação de múltiplas / distribuídas entidades dotadas de certo grau, normalmente variável e heterogêneo, de autonomia. Estas entidades são usualmente chamadas de agentes, e o sistema como um todo de sociedade ou comunidade. Segundo [WITTIG, 92], a IAD foi, em 1981, inicial e genericamente definida como o estudo relacionado com “uma rede de sistemas inteligentes fracamente ligados”. Em [BOND, 88] se define IAD como o estudo

relacionado aos problemas de distribuição e coordenação de conhecimento e ações em ambientes que envolvem múltiplas entidades.

A Inteligência Artificial Distribuída é um subcampo da inteligência artificial que tem investigado modelos de conhecimento, assim como, técnicas de comunicação e raciocínio, que agentes computacionais devem necessitar para participar em “sociedades” compostas de computadores e pessoas. Mais genericamente, a IAD está envolvida com situações nas quais vários sistemas interagem em ordem para resolver um problema em comum: computadores e pessoas, sensores, sistemas aéreos, robôs, veículos inteligentes, etc. [MOULIN, 96].

As pesquisas passaram a tratar de questões em um nível abstrato mais alto, voltando-se para o estudo das estratégias pelas quais a decomposição e coordenação do processamento em um sistema distribuído seriam conjugadas com as demandas estruturais do domínio das tarefas, dando origem à área designada Resolução Distribuída de Problemas (*Distributed Problem Solving* – DPS). Concomitantemente, surgiram pesquisas em Inteligência Artificial inspiradas em conceitos clássicos de organizações e sociedades para lidar com o fato de que sistemas inteligentes tradicionais falham em questões fora do seu domínio original, mesmo se o novo domínio é muito próximo do anterior.

A solução convencional de dotar o sistema de mais conhecimento, apenas consegue resolver alguns problemas particulares, tendo sido trocada pela alternativa de se colocar o sistema em uma sociedade de sistemas, de forma que ele passasse a dispor de uma coleção de diversos outros sistemas, com capacidades diversas, que o auxiliasse a superar as suas limitações, de forma similar ao modo como as pessoas superam as suas limitações individuais trabalhando em grupo. O paradigma que se emerge por trás dessa solução é conhecido como Sistemas Multiagente (*Multiagent Systems* – MAS) [DEMAZEAU, 01][SICHMAN, 92].

A IAD difere da área de processamento distribuído por apresentar, além da distribuição dos dados, controle distribuído e envolver uma forte cooperação entre as entidades processadoras.

A Inteligência Artificial Distribuída está voltada para a solução colaborativa de problemas globais por um grupo distribuídos de entidades (que variam de elementos de processamento simples a entidades racionais). A atividade é colaborativa no sentido de que é necessário o compartilhamento de informações e o grupo é distribuído no senso de que o controle e os dados são lógica e algumas vezes geograficamente distribuídos. [FLEISCHHAUER, 96]

Os benefícios da IAD são similares aos benefícios derivados de ter um grupo de pessoas trabalhando juntas para resolver problemas. Problemas que são muito extensos para um só especialista resolver pode ser resolvido por um grupo. Os problemas podem ser resolvidos em partes por pessoas trabalhando em paralelo.

Existem muitas razões para que seja empregado o conceito de Inteligência Artificial Distribuída: [DEMAZEAU, 01]

- ◆ **capacidade de processamento:** esta é uma das principais razões para se aplicar às funcionalidades distribuídas em um sistema: o alto custo do *hardware* necessário para resolver o problema de forma centralizada, que nem sempre é rápido o suficiente para tal tarefa;
- ◆ **segurança e tolerância à falhas:** Se não se pode confiar a resolução de problemas a um único processo, nada mais justo que encaminhar o problema a um grupo de *experts*. Tendo vários processos especialistas, quando um problema surge, este pode ser colocado em evidência, sendo analisado por vários sistemas, paralelamente. Como consequência disso, existem várias opiniões - e não apenas uma, sobre qual atitude ser tomada;
- ◆ **aproveitamento da tecnologia existente:** Utilizando-se da IAD, pode-se resolver problemas que não seriam possíveis resolver com a tecnologia já desenvolvida. Com a IAD a base de conhecimentos pode ser subdividida em áreas, assim como o

problema. A IAD estimula a decomposição do problema total de gerenciamento, tornando a solução o somatório de módulos (sistemas);

- ◆ **adaptabilidade:** sistemas de IAD são mais apropriados para lidar com problemas distribuídos em termos espaciais, lógicos, temporais ou semânticos;
- ◆ **desenvolvimento e gerenciamento:** a inerente modularidade do sistema permite o desenvolvimento de partes do mesmo separadamente, garantindo uma continuidade do ambiente;
- ◆ **eficiência e velocidade:** concorrência e distribuição de processos em diferentes computadores pode aumentar a velocidade de computação e raciocínio, sempre ligado a níveis de coordenação aceitáveis;
- ◆ **integração:** agregação de recursos distribuídos, tais como, redes de computadores ou diferentes domínios de especialistas;
- ◆ **autonomia:** o controle de processos locais de partes de sistemas isolados ou separados pode ser encarado como uma maneira de proteção ou segurança necessária;
- ◆ **naturalidade:** alguns problemas são mais bem descritos de forma distribuída;
- ◆ **limitação de recursos:** agentes computacionais individuais ligados a recursos escassos podem cooperar em busca da resolução de problemas complexos;
- ◆ **especialização:** conhecimento e ações podem ser escolhidos de acordo com o domínio do agente.

Com a distribuição dos processos, surge o problema da padronização. A linguagem de comunicação entre os processos deve ser padronizada, como também a forma de representar o conhecimento - para que o mesmo seja trocado e tratado por processos especialistas distintos sem a perda de informação.

Atualmente, as aplicações de IAD são as mais diversas, incluindo visão computacional, sistemas de informação geográfica, controle de tráfego aéreo, ambientes inteligentes de aprendizagem, automação de escritórios, gerenciamento de redes, etc.

Segundo [SICHMAN, 92], a IAD pode ser dividida em duas áreas maiores, de acordo com o modelo usado para projetar a sociedade distribuída de agentes: Resolução

Distribuída de Problemas e Sistemas Multiagentes. Em qualquer um dos casos, usamos o termo **agente** para as entidades que participam nas atividades de solução de problemas. A grande diferença pode ser observada pela autonomia desses agentes.

2.3.3.1 RESOLUÇÃO DISTRIBUÍDA DE PROBLEMAS

Em DPS, os agentes cooperam uns com os outros, dividindo e compartilhando conhecimento sobre o problema e sobre o processo para obter uma solução. Os agentes são projetados para resolver um problema ou classe de problemas específicos, possuindo, porém, uma visão incompleta do objetivo global. Isto significa que, a priori, os agentes não podem ser usados para resolver qualquer outro problema similar. Em outras palavras, existe um domínio específico para ação dos agentes [DEMAZEAU, 01].

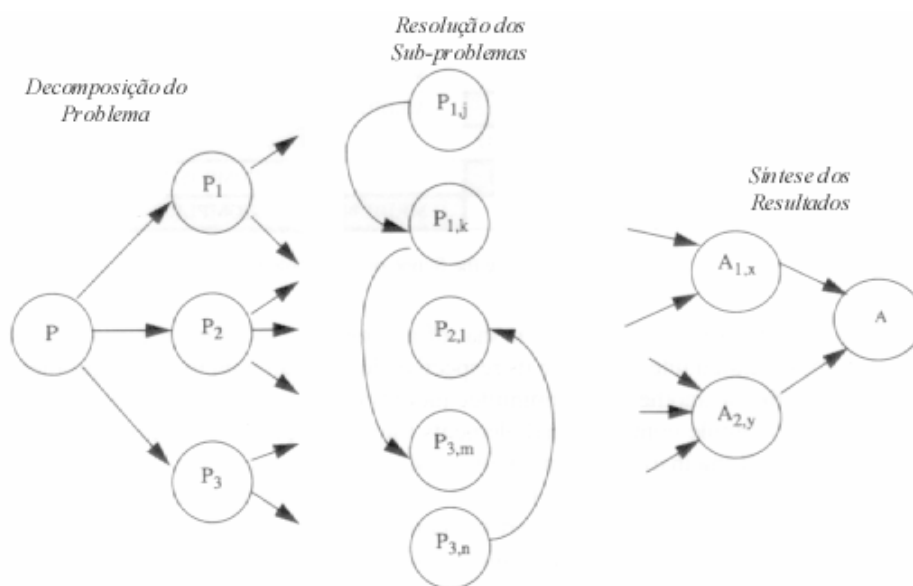


Figura 2 - Etapas da DPS

Os agentes geralmente compartilham um objetivo comum, tem uma linguagem e semântica comum. Além disso, um único agente nunca é capaz de resolver um problema isoladamente, mas apenas a comunidade de agentes como um todo é capaz de realizar a tarefa.

O projeto de um sistema DPS é realizado por um projetista que, primeiramente, realizará uma análise do problema a ser resolvido para, então, identificar os agentes necessários para a solução desse problema. Desta maneira, a tarefa de resolução será decomposta em vários agentes, buscando melhorar o processamento do sistema através da execução paralela. Esta abordagem pode ser representada segundo a Figura 3 [SICHMAN, 92].

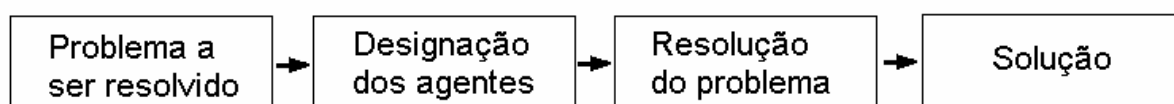


Figura 3 – Fluxo do Projeto de um DPS

2.4 SISTEMAS MULTIAGENTE

Um ambiente multiagente pode ser definido como um sistema no qual diversos agentes interagem entre si e, eventualmente, com outros sistemas (inclusive outros multiagente). Na abordagem MAS o foco principal do projetista não é necessariamente voltado para um problema específico, mas sim em como coordenar inteligentemente o comportamento de um conjunto de agentes autônomos, para obter a solução de um problema apresentado. Esta busca de autonomia se traduz em arquiteturas mais flexíveis, onde a organização do sistema (usualmente chamado de comunidade ou sociedade) está sujeita a mudanças visando adaptar-se a variações no ambiente e / ou no problema a ser resolvido. Muitas vezes, os agentes podem entrar e sair dinamicamente da comunidade [DEMAZEAU, 01].

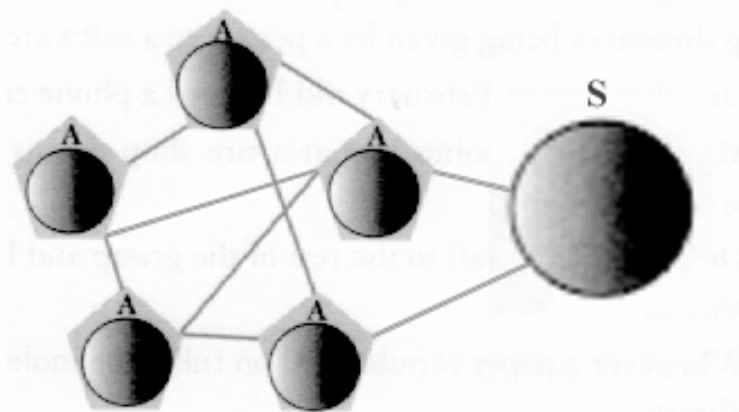


Figura 4 - Sistema Multiagente

A característica essencial da abordagem MAS reside na filosofia de Resolução Distribuída de Problemas (DPS), na qual é adotada uma estratégia de “dividir para conquistar”. Assim, um problema é dividido em subproblemas e cada um é executado separadamente por um “processador” (um agente), cada um destes comunicando ou cooperando entre si quando necessário, com a idéia básica de que a soma dos resultados locais corresponde à solução do problema geral [JENNINGS, 94].

Em DPS, toda a comunidade (sociedade) de agentes é responsável como um todo pelo acompanhamento de um dado problema, ou seja, um agente nunca é capaz de solucioná-lo sozinho. Os agentes geralmente partilham um objetivo comum global, utilizando uma mesma linguagem e semântica de interação. Em MAS, os agentes são muito mais independentes, partilham o mesmo ambiente, competem entre si por recursos, têm que coordenar as suas atividades visando aumentar a eficiência da resolução do problema e têm que saber evitar conflitos. Além disso, um agente pode eventualmente resolver todo um problema. Não necessitam utilizar uma mesma linguagem o que, por outro lado, implica na necessidade de haver traduções e mapeamentos para as suas representações individuais [WITTIG, 92]. Em [DEMAZEAU, 98] é advogado que a diferença entre estes dois grupos se resume na palavra “autonomia”. Em DPS a comunidade de agentes é desenhada para um problema em específico (*close world*), sem autonomia (e, normalmente, sem existência) para participarem na resolução de um outro tipo de problema. Em MAS a comunidade é desenhada para cooperar num ambiente aberto

multiagente, onde cada agente tem autonomia (existe) e pode, eventualmente, participar na resolução de um dado problema.

Dessa maneira, na abordagem MAS, os agentes são primeiramente criados pelo projetista para após se estudar em que ambientes essa sociedade de agentes pode ser utilizada. O esquema de resolução de problemas na abordagem MAS pode ser observado na Figura 5 [SICHMAN, 92].

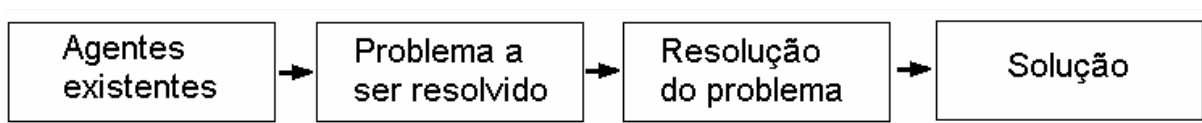


Figura 5 – Fluxo do Projeto de um MAS

Na verdade, vive-se atualmente um momento de “boom” na utilização de MAS na construção de modernos e complexos sistemas distribuídos, nas mais variadas áreas. Uma das razões principais para tal encontra-se na questão tecnológica. Hoje, com base nos desempenhos, capacidades, estabilidades e custos das redes de comunicação, *software* e *hardware*, é possível desenvolver soluções efetivamente viáveis, quer tecnicamente, quer economicamente.

A aplicação de uma abordagem MAS é especialmente útil para aplicações cujos problemas: [GESZYCHTER, 99]

- ◆ sejam intrinsecamente distribuídos;
- ◆ requeiram uma junção de diferentes domínios de conhecimento para a sua execução;
- ◆ requeiram a aplicação de diferentes resolvidores de problemas, integrados em um mesmo ambiente;
- ◆ requeiram diferentes níveis de autonomia e descentralização de resultados e decisões;
- ◆ sejam dinâmicos;
- ◆ sejam extremamente conflitantes, em função das muitas, normais e dinâmicas restrições usualmente existentes, o que requer variados níveis de cooperação e negociação a fim de que a sua execução não seja interrompida.

A abordagem MAS, ao suportar aplicações em domínios de sistemas distribuídos, proporciona uma maior eficiência em termos de: [GESZYCHTER, 99]

- ◆ tolerância à falhas: uma falha local não implica em uma falha global;
- ◆ resolução de problemas complexos e que requeiram cooperação;
- ◆ execução assíncrona;
- ◆ rapidez na obtenção de uma solução devido à execução paralela de tarefas;
- ◆ robustez de um sistema, bem como potencial melhora na qualidade dos resultados na medida que suporta se ter vários resolvidores de problemas em um mesmo ambiente;
- ◆ reutilização e escalabilidade do sistema;
- ◆ integração de fontes de conhecimento e subsistemas heterogêneos;
- ◆ permitir a interconexão e interoperação de múltiplos sistemas existentes;
- ◆ melhora na modularidade, flexibilidade e reusabilidade ao nível de conhecimento.

2.4.1 AGENTES

Até este ponto foram introduzidos os conceitos de IAD, MAS e, vagamente, associado a um agente o conceito de um nó em uma rede que tem capacidade de resolução de problemas. Mas o que será um agente, esta entidade primordial de um sistema multiagente? Apesar do seu caráter essencial, esta é uma das perguntas cuja resposta menos consenso obtém dentro da comunidade de IAD. Isto se deve, basicamente, a que a definição de um agente é fortemente influenciada pelo domínio da aplicação e do problema, das formas de cooperação e dos seus níveis de autonomia. Em [FRANKLIN, 96] é feito um sumário de uma série de definições existentes na literatura, algumas das quais são abaixo transcritas:

- ◆ *“Agentes são sistemas computacionais que habitam um ambiente com alguma dinâmica e complexidade, sentem e agem autonomamente neste ambiente e, desta forma, são capazes de atingir / executar um conjunto de objetivos / tarefas para os quais foram projetados.”;*

- ◆ “Agentes são entidades capazes de executar três funções: percepção das condições dinâmicas do ambiente; ação para afetar condições deste ambiente; e raciocínio para interpretar as percepções, resolver problemas, gerar inferências, e determinar ações.”;
- ◆ “Um agente corresponde a um hardware ou (mais comumente um) software que possui as seguintes propriedades: autonomia (pode agir por si só e controlar as suas ações), sociabilidade (interage com outros agentes), reatividade (recebe informações do ambiente e reage de acordo com elas), e pró-atividade (não reage pura e simplesmente perante o ambiente, mas é capaz de exibir algum comportamento baseado nos seus objetivos).”;
- ◆ “Agentes são programas que estão envolvidos em atividades de diálogo e que podem negociar e coordenar transferências de informação.”;
- ◆ “Um agente corresponde a um módulo de software persistente com suas próprias idéias sobre como uma tarefa deve ser executada, dedicado a um propósito específico dentro de um certo domínio.”;
- ◆ “Um agente é uma entidade real ou virtual: que está imersa em um ambiente dentro do qual ela pode executar certas ações; que é capaz de aprender e representar parcialmente este ambiente; que é capaz de comunicar com os outros agentes; e que possui um comportamento autônomo que é consequência das suas observações, do seu conhecimento e das suas interações com os outros agentes.”.

Um agente é essencialmente caracterizado por aquilo que sabe fazer, pelo comportamento que exhibe. À parte a enorme diversidade das definições de agentes, a questão pode ser colocada em termos de propriedades fundamentais que diferenciam e caracterizam os agentes perante outros tipos de “entidades”. Assim sendo, todo agente **tem** que apresentar pelo menos as três propriedades fundamentais: [AUER, 95]

- ◆ (um certo grau de) **autonomia** para raciocinar e tomar decisões por sua própria vontade;
- ◆ capacidade de **interagir** com outros agentes;
- ◆ (um certo grau de) **independência** para resolver um problema, isto é, o agente tem conhecimento sobre como resolver (pelo menos parte de) um problema.

Na CLIDENP, sistema objeto desta pesquisa, o usuário terá um sistema multiagente com agentes independentes e autônomos, que poderá integrar seus agentes através de uma infraestrutura de comunicações (com todos os métodos necessários, não apenas de comunicação em si, mas também de manutenção da consistência do sistema ao longo do seu ciclo de vida) e a topologia do sistema, utilizando o protocolo HTTP. O projeto e a implementação do comportamento e propriedades dos agentes serão previamente definidos para a correta representação dos sistemas especialistas legados.

Além das propriedades fundamentais, os agentes ainda podem apresentar uma série de outras propriedades, dentre as quais:

- ◆ **agência** que pode ser conceituada como o grau de autonomia e autoridade investida no agente e pode ser medida, ao menos qualitativamente, através da natureza da interação entre o agente e outras entidades do sistema [SOUZA, 97];
- ◆ **comunicabilidade** que é a capacidade de trocar informações com outras entidades, que podem ser agentes, humanos, objetos, seu ambiente, etc. Os agentes, no curso da realização de seus objetivos, devem acessar informações sobre o estado atual do ambiente externo [FLEISCHHAUER, 96];
- ◆ **inteligência** que é um conjunto de recursos, atributos e características que habilitam o agente a decidir que ações executar [WOOLDRIDGE, 95];
- ◆ **mobilidade** que é a capacidade do agente de se transportar de uma máquina à outra [FRANKLIN, 96];
- ◆ **reatividade** que é a propriedade que permite aos agentes perceberem seus ambientes e responderem adequadamente às mudanças neles ocorridas [WOOLDRIDGE, 95];
- ◆ **flexibilidade** que é a habilidade dos agentes de escolher dinamicamente as ações e a seqüência de execução das mesmas, em resposta a um estado do ambiente [AUER, 95];
- ◆ **planejamento** é a habilidade do agente de sintetizar e escolher entre diferentes cursos de ações, com o propósito de alcançar seus objetivos. Especificamente é o processo através do qual o agente determina qual ação é apropriada à situação [AUER, 95].

Existem, basicamente, três formas de classificar os agentes segundo suas propriedades [FLEISCHHAUER, 96]. São elas:

- ◆ segundo sua mobilidade: agentes estáticos ou móveis;
- ◆ segundo sua reatividade: agentes deliberativos ou reativos;
- ◆ segundo as propriedades que o agente pode exibir: autonomia, aprendizado e cooperação.

Usa-se estas três propriedades mínimas para derivar os quatro tipos de agentes que estão incluídos nesta tipologia: *agentes colaborativos*, *agentes colaborativos evolutivos*, *agentes de interface* e *agentes inteligentes*, conforme mostra a Figura 6 [FLEISCHHAUER, 96].

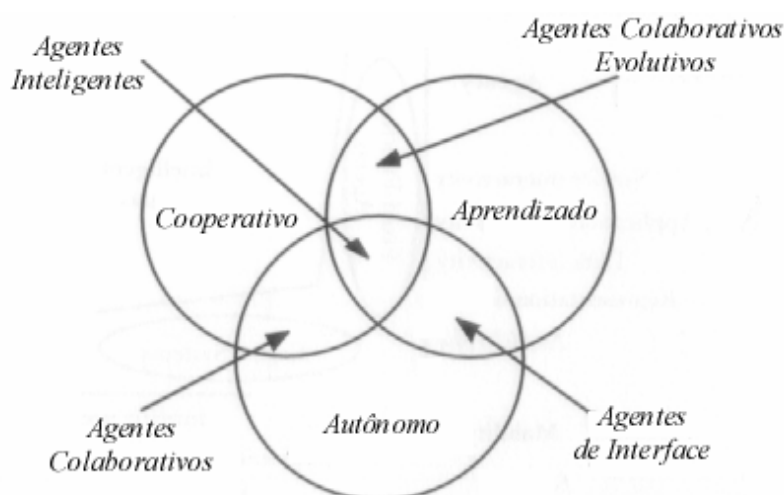


Figura 6 - Tipologia de Agentes

2.4.1.1 AGENTES MÓVEIS

Tem-se observado que a tecnologia de agentes estáticos tem sido utilizada nas últimas décadas em muitas áreas de aplicações, devido ao fato de que a tecnologia de agentes móveis ainda é um conceito novo. Não há muitas aplicações desenvolvidas utilizando esta tecnologia.

Mas o que são agentes móveis? Agentes móveis são processos computacionais capazes de se movimentar em uma rede, tais como WWW, interagindo com outros *hosts*, reunindo as informações e realizando tarefas requeridas pelo seu usuário. Agentes móveis são agentes porque são autônomos e cooperam. Por exemplo, eles podem cooperar ou comunicar com um agente a fim de que outro agente retorne a localização de um dos seus métodos internos [FLEISCHHAUER, 96].

2.4.1.2 AGENTES DELIBERATIVOS OU SIMBÓLICOS

Agentes deliberativos são baseados em modelos de organizações sociais, no senso de sociedades humanas (grupos, hierarquias, comércio). Com os agentes, há uma representação explícita de ambiente e de membros da sociedade. Eles podem raciocinar sobre as ações tomadas no passado e planejar as ações a serem tomadas no futuro. Do ponto de vista de uma sociedade, ela é geralmente composta de um número pequeno de membros [DEMAZEAU, 01].

Os agentes deliberativos baseiam-se em mecanismos de processamento simbólico existentes nos sistemas mais tradicionais da Inteligência Artificial, como redes semânticas, sistemas de regras, etc.

2.4.1.3 AGENTES REATIVOS

Os agentes reativos são uma categoria especial de agentes que não possuem modelos internos de seu ambiente. Ao invés disso, eles atuam de forma a responder a um estímulo ou ao estado atual do ambiente. O ponto mais importante dos agentes reativos é sua simplicidade e a interação com outros agentes.

Agentes reativos são considerados entidades mais simples que os deliberativos devido às suas características, principalmente de não apresentarem estados mentais (desejos, intenções, crenças e outros) e, portanto, não possuem capacidade de raciocínio e de planejamento [FROZZA, 97].

Em um sistema multiagente, não é necessário que cada agente seja individualmente inteligente, pois a idéia principal é que um comportamento global inteligente seja alcançado a partir do comportamento individual do agente. Tudo que um agente reativo sabe sobre as ações e os comportamentos dos outros agentes são percebidos pelas mudanças no ambiente. Cada agente, individualmente, exibe um comportamento de acordo com a situação na qual se encontra no ambiente de solução de algum problema. Quando o ambiente altera-se, os agentes reativos mudam seu comportamento [FROZZA, 97].

2.4.1.4 AGENTES COLABORATIVOS

Como visto na figura 3, os agentes colaborativos enfatizam a cooperação e autonomia (com outros agentes) de forma a realizar suas tarefas. Eles podem aprender, mas este aspecto não é, tipicamente, o de maior ênfase de suas operações. De forma a ser um conjunto de agentes coordenados, eles devem negociar de maneira a atingir situações mutuamente harmoniosas [FLEISCHHAUER, 96].

Em resumo, as características chaves destes agentes incluem: autonomia, capacidades sociais, iniciativa e pró-atividade, uma vez que eles são capazes de atuar racionalmente e autonomamente em um ambiente aberto multiagente.

2.4.1.5 AGENTES COLABORATIVOS EVOLUTIVOS

Um agente colaborativo evolutivo aprendiz observa as ações do usuário em *background*, encontra padrões repetitivos e automatiza-os sob aprovação do usuário observado. O paradigma de agente aprendiz usa a metáfora de assistente pessoal. Agentes são, em particular, aplicáveis quando o domínio da aplicação contém comportamento repetitivo significativo e, ainda quando o comportamento repetitivo difere através do espectro do usuário [FLEISCHHAUER, 96].

2.4.1.6 AGENTES DE INTERFACE

Agentes de interface enfatizam autonomia e aprendizado de forma a realizar suas tarefas. A metáfora que define este agente é de um assistente pessoal que está colaborando com o usuário no mesmo ambiente de trabalho. Essencialmente, o agente de interface suporta e fornece assistência, tipicamente a um usuário aprendendo uma aplicação particular. O agente observa e monitora as ações tomadas pelo usuário, e sugere melhores formas de efetuar a tarefa [FRANKLIN, 96].

Sua cooperação com outros agentes, se existe, é limitada a pedir alguma informação. Os modos de aprendizado são tipicamente por repetições habituais (aprendizado baseado em memória) ou paramétricas, ou mediante técnicas tais como aprendizado evolucionário. Em resumo, um agente de interface é oposto a todo tipo de agente, é aquele que usa técnicas de aprendizado de máquina para apresentar uma interface pseudo-inteligente para o usuário.

2.4.1.7 AGENTES INTELIGENTES

Agentes inteligentes são entidades que realizam algum conjunto de operações em favor de um usuário ou outro programa com algum grau de independência ou autonomia, e assim, empregam algum conhecimento ou representação dos objetivos ou aspirações do usuário. Agentes inteligentes podem ser descritos em termos de espaço definido por três dimensões de agência, inteligência e mobilidade [COSTA, 97][FLEISCHHAUER, 96].

O grau de agência é realçado, aumentado se um agente representa um usuário em algum meio. Um agente mais avançado pode interagir com outras entidades tal como, dados, aplicações ou serviços, além disso, colaboram e negociam com outros agentes [SOUZA, 97].

Inteligência é o grau de raciocínio e comportamento sábio: a habilidade do agente para aceitar a declaração de objetivo do usuário e realizar a tarefa delegada a ele. No mínimo, podem ser algumas declarações de preferências, talvez na forma de regras, com uma engenharia de inferência ou algum outro mecanismo de raciocínio para atuar sobre

essas preferências. A mobilidade aparece numa terceira dimensão, quando a aplicação do agente é projetada para atuar em uma rede. Alguns agentes inteligentes podem ser estáticos, outros residem numa máquina cliente ou instanciados no servidor.

2.4.2 COMUNICAÇÃO ENTRE AGENTES

Na comunidade de Inteligência Artificial Distribuída existe um consenso sobre a necessidade explícita de comunicação para que haja a cooperação entre agentes autônomos.

O controle da comunicação sempre foi um problema crucial nos sistemas distribuídos. A pequena capacidade dos canais de comunicação representa uma limitação para a performance nos sistemas. Na IAD o problema é amplificado pela necessidade de interação entre os agentes. As soluções propostas sempre tentaram a comunicação implícita por uma memória compartilhada ou, se os agentes são mais autônomos, pela aplicação de estratégias de controle especializadas [RABELO, 97].

É de comum acordo, também, que comunicação em IAD é mais complexa do que aquela existente entre sistemas distribuídos tradicionais. Enquanto os processos deste último modelo já são conhecidos e bem definidos (em vários casos, padronizados), não existe um consenso sobre como a comunicação deve ser tratada na IAD.

Na maioria das propostas apresentadas para a comunicação entre agentes, existe uma relação estrita com a teoria da ação na fala, em especial, na forma de classificação dos atos de comunicação. Porém, a alta subjetividade existente na comunicação por linguagem natural, devido à falta de um padrão formal, leva a ambigüidades e dificuldades em sua interpretação, tornando proibitivas as abordagens por estes meios.

A interação e a interoperação entre agentes, para atingir objetivos comuns, requer mais que uma linguagem de comum entendimento dos agentes envolvidos nesse processo. O comportamento cooperativo em uma comunidade de agentes requer o cumprimento de três requisitos [COSTA, 97]:

- (1) uma linguagem em comum;
- (2) um entendimento comum sobre a informação e o conhecimento compartilhados;
- (3) a habilidade para entender / executar o que está incluído em (1) e (2).

Os agentes são em sua maioria entidades computacionais residentes em um nível de conhecimento e eles não estariam bem servidos pelas linguagens e protocolos desenvolvidos para a computação distribuída. Estas linguagens e protocolos focalizam um processo ao invés de um programa ou uma coleção de programas que constituem um agente. Como resultado, uma linguagem de comunicação deve ser poderosa o suficiente para suportar a comunicação entre programas em um alto nível.

Uma linguagem de comunicação não é um protocolo. A distinção entre linguagem de comunicação e protocolo é difusa. Um protocolo, como os que são utilizados no contexto das linguagens de comunicação, pode apresentar um dos seguintes significados:

- ◆ um protocolo de transporte como FTP, HTTP, etc;
- ◆ uma estrutura de interação de alto nível, como negociação, etc;
- ◆ uma restrição às possíveis trocas válidas de primitivas de comunicação.

Uma linguagem de comunicação pode utilizar os protocolos do primeiro tipo como mecanismo de transporte, pode ser usada por protocolos do segundo tipo como uma forma de implementação, e normalmente inclui protocolos do terceiro tipo. Mas definitivamente uma linguagem de comunicação não é meramente um protocolo [FININ, 95].

Uma Linguagem para Comunicação de Agentes (LCA) normalmente consiste em um conjunto de primitivas conhecido por todos os agentes da comunidade e um conjunto de regras de conversação. As primitivas informam o que está sendo compartilhado e o que deve ser feito com as informações. As regras de conversação, por sua vez regulamentam as atitudes adotadas pelos agentes durante a comunicação. Uma LCA, normalmente, é implementada em camadas e, como já foi dito, utilizam mecanismos para a comunicação de processos provenientes de sistemas distribuídos. Finin, Labrou e

Mayfield sugeriram um conjunto de necessidades para uma LCA em "*KQML as an agent communication language*" [FININ, 95]. Essas necessidades são divididas em sete categorias [RABELO, 97]:

- ◆ **forma:** uma boa LCA deve ser declarativa, sintaticamente simples e legível por seres humanos. Deve apresentar consistência, ser fácil de analisar e gerar. Uma linguagem deve ser linear ou facilmente transformada em uma forma linear. Finalmente, deve apresentar uma sintaxe extensível;
- ◆ **conteúdo:** uma LCA deve ser organizada em camadas visando uma boa adaptação com outros sistemas. Em particular uma distinção deve ser feita entre linguagens de comunicação, que expressam atos comunicativos e o conteúdo da linguagem, que expressa fatos sobre o domínio. A organização em camadas facilita a integração da linguagem com a aplicação, além de propiciar uma estrutura conceitual para o entendimento da linguagem. A linguagem deve ainda, basear-se em um conjunto de atos comunicativos (primitivas);
- ◆ **semântica:** a semântica de uma linguagem de comunicação não deve ser ambígua, deve ser possível, apresentar, uma forma canônica, isto é, a similaridade de um significado deve levar à similaridade da representação;
- ◆ **implementação:** a implementação deve ser eficiente, tanto na velocidade quanto no espectro de utilização. Deve proporcionar uma boa integração com as tecnologias de *software* existentes e sua interface deve ser de fácil utilização. Finalmente, a linguagem deve ser amigável com implementações parciais;
- ◆ **redes de comunicação:** uma LCA deve adaptar-se bem às tecnologias modernas de redes de computadores. As linguagens devem suportar todos os tipos básicos de conexão: *ponto a ponto*, *multicast* e *broadcast*. As conexões síncronas e assíncronas devem ser suportadas. A linguagem deve ser formada por um conjunto de primitivas suficientemente rico para poder servir como substrato onde linguagens de mais alto nível e protocolos de interação possam ser construídos. Mais ainda, estes protocolos de mais alto nível devem poder ser implementados independentemente do mecanismo de transporte utilizado;
- ◆ **ambiente:** o ambiente onde os agentes serão requisitados para trabalhar será altamente distribuído, heterogêneo e extremamente dinâmico. Para propiciar um canal de comunicação neste ambiente, uma linguagem de comunicação precisa

fornecer ferramentas para prover heterogeneidade e dinamismo. É necessário proporcionar a interoperação com outras linguagens e protocolos. Finalmente, deve ser facilmente incorporada aos sistemas legados;

- ◆ **confiabilidade:** uma linguagem deve permitir uma comunicação entre agentes confiável e segura. Precauções para segurança e conversações privadas entre dois agentes devem ser suportadas. Na linguagem deve haver uma forma de garantir a autenticidade dos agentes. Deve ainda ser robusta o suficiente para suportar mensagens não apropriadas e mal formuladas e ainda possuir mecanismos para identificação e sinalização de erros e advertências.

Dentre as LCAs, a mais conhecida e relevante é o KQML (*Knowledge Query Manipulation Language*), que tem vindo a ser usada como proposta de fato para a comunicação, não só entre sistemas multiagente, mas também entre sistemas distribuídos e heterogêneos. Mais recentemente, ressalta-se a iniciativa da FIPA (*Foundation for Intelligent Physical Agents*) [FIPA] na direção do alargamento da discussão sobre os vários aspectos relacionados com protocolos de comunicação inter-agentes [RABELO, 97].

2.4.3 CLASSIFICAÇÕES DE MAS

Uma vez que nós tenhamos uma linguagem e a habilidade para construir agentes, ainda resta a questão de como esses agentes devem ser organizados para aumentar a colaboração, poder-se-ia dizer que se tem os “indivíduos” mas temos que pensar na “sociedade” desejada. Duas abordagens diferentes podem ser consideradas: comunicação direta e comunicação assistida. A vantagem da comunicação direta, na qual os agentes manuseiam sua própria coordenação, é que ela não recai na existência, capacidades, ou polarizações de quaisquer outros programas.

Existem diferentes tipos de sistemas multiagente, e em sua maioria, diferem quanto à sua arquitetura, possibilidades de comunicação e complexidade do agente. Os três tipos de arquiteturas de sistemas multiagente geralmente utilizados são Sistemas Multiagente Federados, Democráticos e Abertos [DEMAZEAU, 01].

2.4.3.1 SISTEMAS MULTIAGENTE FEDERADOS

Um Sistema Multiagente é chamado Federado quando existem agentes especiais, complexos, chamados de facilitadores, que possuem um conhecimento a respeito das habilidades de outros agentes mais simples. O papel dos facilitadores é organizar o trabalho entre os agentes mais simples, como mostra a Figura 7. Uma vez que uma determinada requisição é recebida por um facilitador, ele responsabiliza-se por encontrar um agente mais simples capaz de realizar esta requisição. A principal vantagem dessa abordagem é permitir o gerenciamento eficiente da comunicação entre os agentes. Por outro lado, a centralização de importantes tarefas torna o sistema como um todo sensível ao mau funcionamento do agente facilitador [GENESARETH, 94].

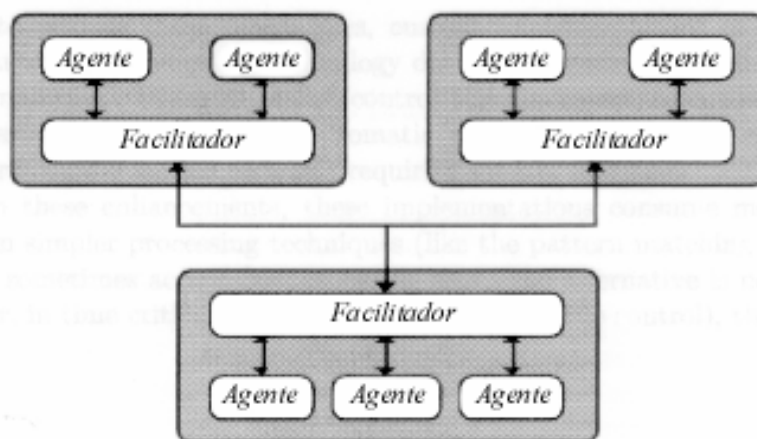


Figura 7 - Sistema Multiagente Federado

2.4.3.2 SISTEMAS MULTIAGENTE DEMOCRÁTICOS

A principal característica de um Sistema Multiagente Democrático é o fato de todos os agentes da comunidade possuírem o mesmo nível hierárquico. Nesta abordagem, os agentes realizam ações coletivas assincronamente e a comunicação é uma destas ações. Esta comunicação tipicamente obedece às regras de uma Linguagem de Cooperação adequada, como por exemplo, KQML. As principais vantagens dessa abordagem são a modularidade e a flexibilidade, e a desvantagem é que cada um dos agentes necessita

conhecer, ao menos parcialmente, a identidade e as habilidades dos demais agentes da comunidade.

2.4.3.3 SISTEMAS MULTIAGENTE ABERTOS

Um MAS é dito aberto quando a composição da comunidade não é fixa, permitindo que os agentes possam subscrever-se e desligar-se da comunidade dinamicamente. Neste tipo de sistema, alguns serviços podem estar disponíveis ou não, de acordo com a composição da comunidade em um dado momento. Os agentes podem adaptar-se e escolher diferentes objetivos a serem alcançados, planos de execução ou padrões de cooperação, dependendo dos serviços disponíveis na comunidade. A principal vantagem dessa abordagem é a sua robustez. A desvantagem é a excessiva troca de mensagens realizada por um complexo protocolo de comunicação capaz de assegurar o caráter de Comunidade Aberta.

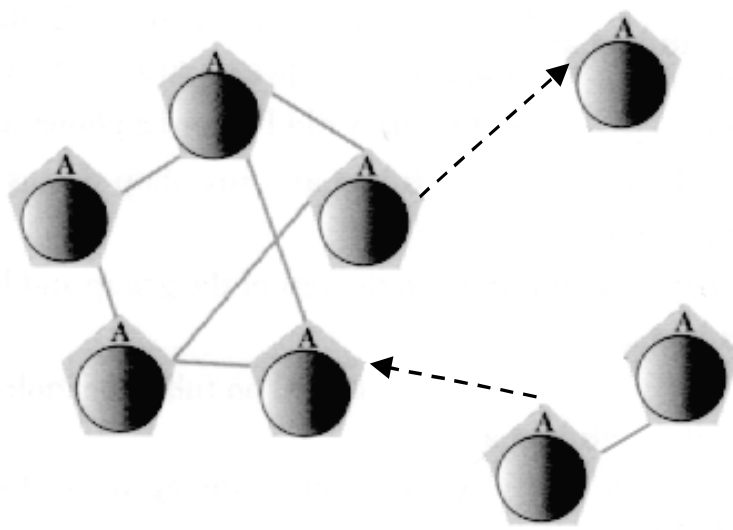


Figura 8 - Sistema Multiagente Aberto

2.5 SISTEMAS ESPECIALISTAS

Um Sistema Especialista é uma aplicação da área da Inteligência Artificial que, como várias outras áreas da Ciência da Computação, não possui uma definição totalmente aceita. Uma evidência deste fato são os vários conceitos encontrados:

- ◆ Sistemas Especialistas (SE) são programas de computadores que imitam o comportamento de especialistas humanos em um domínio de conhecimento específico. Estes sistemas são úteis para a realização de inferências e deduções de problemas envolvendo aspectos não estruturados [REIS, 97].
- ◆ Um SE é uma aplicação da área de IA (Inteligência Artificial) que toma as decisões ou soluciona problemas em um domínio de aplicação, pelo uso do conhecimento e regras definidas por especialistas neste domínio. Os SEs solucionam problemas que normalmente são solucionados apenas por “especialistas” reais [KOEHLER, 98].
- ◆ Sistemas especialistas são sistemas de informação que contêm dados ou sentenças descritivas sobre determinado ramo da atividade humana como, por exemplo, diagnóstico médico, previsão financeira, previsão do tempo, consultas sobre arqueologia e línguas. [SIMÕES, 00].

O objetivo principal dos Sistemas Especialistas é transferir para outros usuários do domínio da aplicação em questão, o conhecimento adquirido junto a um especialista de uma área particular do conhecimento humano, pois o conhecimento do especialista é sua principal matéria prima. Um Sistema Especialista pode representar este conhecimento através de diversos formalismos. Os Sistemas Especialistas podem também ser utilizados com as seguintes finalidades: [RABUSKE, 95]

- ◆ **interpretação:** analisar e interpretar certos tipos de informações (imagens de satélites);
- ◆ **diagnóstico:** inferir as conseqüências de uma dada situação inicial (previsão do tempo);
- ◆ **síntese:** configurar sistemas ou objetos a partir de um conjunto de especificações (configuração de aparelhos utilizados em cromatografia iônica);
- ◆ **planejamento:** estabelecer uma seqüência de ações que atinja determinada meta (trajetórias para robôs);
- ◆ **monitoramento:** acompanhar a evolução de determinado sistema (supervisão de processos industriais);

- ◆ **correção de falhas:** propor medidas corretivas para falhas em sistemas (manutenção de aeronaves);
- ◆ **instrução:** propor problemas e acompanhar sua solução pelo treinamento (treinamento de operadores de processos);
- ◆ **controle:** impor ao sistema certo comportamento desejado (controle de processos industriais).

Os SEs se caracterizam por possuir habilidade para trabalhar ao nível do especialista, representar o conhecimento específico da maneira que o especialista pensa, incorporar o processo de explanação e formas de manipulação de incerteza e, ser normalmente pertinente a problemas que podem ser, simbolicamente ou não, representados [REIS, 01].

2.5.1 COMPONENTES

A estrutura de um SE compreende quatro componentes essenciais: a base de conhecimentos, o módulo de aquisição de conhecimento, o mecanismo de inferência e o módulo de explicações.

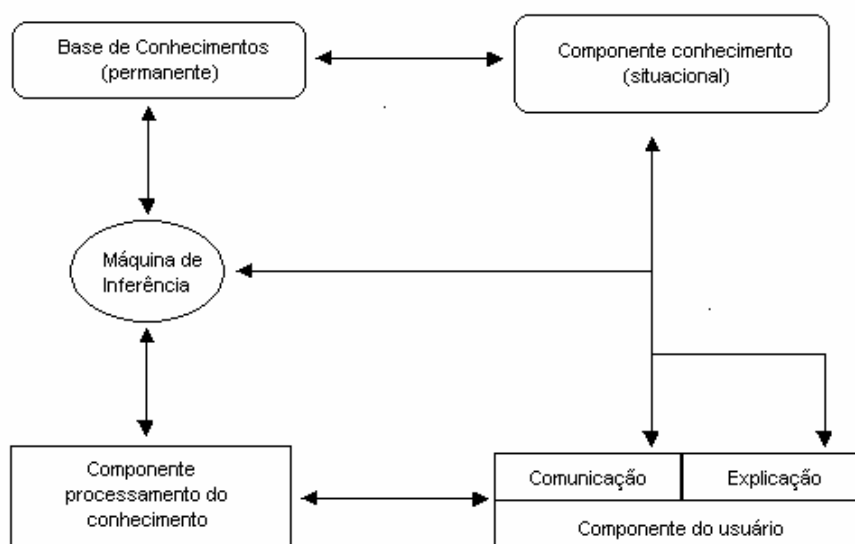


Figura 9 - Componentes de um Sistema Especialista

- ◆ **base de conhecimento:** é o local onde ficam os fatos e regras que representam as regras de inferência do especialista humano [RABUSKE, 1995]. Ela é formada pelas regras e procedimentos que o especialista humano usa na solução de problemas;
- ◆ **mecanismo de inferência:** ou máquina de inferência, é o mecanismo que procura as respostas na base de conhecimento. As respostas são obtidas através das regras necessárias a serem avaliadas, e para isto, a máquina de inferência ordena-as de uma maneira lógica, direcionando o processo de inferência. Uma máquina de inferência toma decisões e julgamentos baseados em dados simbólicos contidos na base de conhecimento [RABUSKE, 1995];
- ◆ **módulo de aquisição do conhecimento:** são subsistemas de aquisição de conhecimento onde podem ser introduzidos novos conhecimentos e antigos conhecimentos podem ser alterados ou eliminados; sendo utilizado para alimentar a base de conhecimento [RABUSKE, 1995];
- ◆ **subsistema de explicações:** é o sistema responsável por explicar os métodos, ou seja, a linha de raciocínio utilizada pelo sistema especialista para a obtenção da conclusão. Assim, os usuários do sistema especialista podem entender porque e como foi obtida uma determinada resposta do sistema. Os subsistemas de explicação são excelentes mecanismos para se utilizar em situações instrutoriais e para *debugging*³ do sistema durante seu desenvolvimento.

A comunicação entre os componentes do Sistema Especialista com o usuário final é feita através de uma interface, que também explica ao usuário como todo o resultado foi obtido. A construção de uma interface depende da ferramenta de desenvolvimento utilizada, mas geralmente é constituída de menus, formulários eletrônicos, relatórios e gráficos. Hoje, estas interfaces podem ser na plataforma *Web* (exibidas pelo *browser*) ou local (através da interface padrão *Windows*). Todas as perguntas, representações gráficas, respostas e resultados de consultas devem ser exibidos por esta interface.

2.5.2 REPRESENTAÇÃO DAS INCERTEZAS

Existem dois métodos de representação de incertezas: o Simbólico e o Numérico. O primeiro trata incertezas através de regras de inferência que representam as exceções no raciocínio do especialista e, portanto é viável para trabalhar com uma pequena quantidade de exceções. Muitas vezes as exceções às regras de inferência são explicitadas com o objetivo de aumentar o conhecimento do sistema para resolver problemas. Se as exceções forem muitas, torná-las explícitas pode ser uma tarefa extremamente difícil o que inviabilizaria esta forma de representação. Outro fator a considerar é que as exceções devem ser identificadas e representadas no SE antes de serem realizadas inferências e combinações. Por outro lado, o método numérico propaga a incerteza numericamente através das inferências e combinações de evidências [NASSAR, 02].

Os principais paradigmas numéricos de representação de incerteza são:

- ◆ **lógica difusa:** os valores assumidos pelos extremos lógicos na álgebra booleana, falso e verdadeiro, seriam representados numericamente por 0 e 1 respectivamente. Na lógica difusa o elemento possui um grau de pertinência que varia de 0 a 1;
- ◆ **fatores de certeza:** São fatores aproximados, baseados mais na pragmática do que na estatística. O sistema especialista associa a cada uma de suas informações um determinado grau de crença.
- ◆ **teoria da probabilidade:** Esta abordagem exige probabilidades, ou seja, números associando eventos e populações. Trata-se de uma estrutura de representação de eventos aleatórios, onde a probabilidade de um evento assume valor de 0 a 1. Nesta teoria, é considerada também a probabilidade de que ocorra um evento (“conseqüência”) condicionado a ocorrência de um outro evento A (“causa”), esta probabilidade é suportada pelo enfoque do teorema de Bayes. Os programas de diagnóstico diferencial baseados no teorema de Bayes usam a abordagem de construir uma matriz contendo as probabilidades de um dado sintoma, sinal, etc., ser associado a uma doença, como resultado de algum estudo com base frequentista numa amostra de pacientes.

³ acompanhamento passo-à-passo de um algoritmo executado por um sistema de informação

- ◆ **teoria da evidência:** Esta teoria trata a representação de incertezas de forma semelhante à da Teoria de Bayes. O que ocorre é que o raciocínio é feito com medidas de crença, que são obtidas através de funções de crença. Estas funções tornam-se realmente úteis apenas quando agregadas pela regra de combinação de Dempster de modo a gerar uma única função de crença.

Os sistemas especialistas proliferaram porque é mais fácil implementar bases de conhecimento médico mais restritas. Elas são mais bem definidas em termos de representação do conhecimento, regras de decisão, dados para apoiar a decisão, padronização de nomenclatura e concordância entre os especialistas; e produzem resultados mais úteis do ponto de vista prático, pois enfocam seus poderes de resolução em problemas diagnósticos difíceis. São geralmente considerados como especialistas confiáveis, pelos médicos que os utilizaram (KULIKOWSKI, 82).

Um SE além de sugerir diagnósticos e auxiliar no apoio a decisão, pode citar dados da literatura que respaldem as suas afirmações, avaliações, propostas, entre outros, estes SE possuem a característica de capacidade explicativa.

2.5.3 SISTEMAS ESPECIALISTAS PROBABILÍSTICOS

Os Sistemas Especialistas Probabilísticos (SEP), são os que visam tratar de uma forma adequada o conhecimento vago e incompleto. A tendência da utilização de conhecimento vago através de métodos puramente probabilísticos, parecia limitada, mas prosperou na medida que foi possível estabelecer uma relação entre a probabilidade e a lógica da matemática, isto é, os estudos sobre estruturas de relações de dependência e independência em conjuntos de variáveis, operacionalizaram os modelos probabilísticos para manipular conhecimentos [RODDER, 95].

O conhecimento necessário para resolver problemas que exigem alto grau de certeza é geralmente de natureza incerta. A necessidade de tratar a incerteza nesses sistemas levou à construção de sistemas especialistas probabilísticos, que possuem todas as

características de um sistema especialista clássico, porém a sua base de conhecimento é representada por meio da distribuição de probabilidades [KOEHLER, 98].

Os sistemas especialistas probabilísticos têm em sua base de conhecimento fatos e regras que representam o conhecimento do especialista num domínio de aplicação. Aos fatos e às regras são associadas às incertezas presentes no domínio, sendo explicitada a crença em sua ocorrência por meio de valores de probabilidade. O raciocínio realizado pelo sistema deve considerar estas probabilidades para a partir dos dados de entrada associar um vetor de probabilidades ao conjunto de hipóteses diagnósticas. A hipótese com maior probabilidade de ocorrência pode ser considerada a conclusão do sistema, estando associada a ela o grau de certeza da resposta do sistema [KOEHLER, 98].

A principal vantagem de um SEP é o tratamento da incerteza sobre os conhecimentos. A lógica clássica na sua mais restrita forma considera uma proposição (regra) apenas como verdadeira ou falsa, o que limita severamente o tratamento de valores intermediários, muito comuns e necessários no trato do conhecimento subjetivo [REIS, 01].

Por tratarem melhor as incertezas sobre os conhecimentos através da aplicação de probabilidades, os SEPs são utilizados em domínios de aplicação mais críticos, com o objetivo de reforçar a qualidade dos resultados fornecidos ao próprio especialista.

3 O SISTEMA MULTIAGENTE PROPOSTO: CLIDENP

O objetivo do projeto CLIDENP é o desenvolvimento um aplicativo utilizando principalmente as tecnologias de sistemas multiagentes e sistemas especialistas, onde além de integrar todos os sistemas especialistas previamente desenvolvidos de forma adequada, espera-se também permitir o treinamento dos profissionais da saúde e a popularização dos conceitos da desnutrição infantil.

Cada agente CLIDENP deverá deter a inteligência gerada, ou programada, nos sistemas de avaliação nutricionais ligados a esta pesquisa – SISPAN, SACI, SANEP e SANPAP.

3.1 DOMÍNIO DE CONHECIMENTO

Para o desenvolvimento de um Sistema Multiagente, é necessário considerar uma variedade de aspectos, incluindo a definição do domínio de conhecimento, a concepção e implementação das suas associadas funcionalidades (comportamentos), o desenvolvimento da arquitetura dos agentes (modelos mental, de ação e de cooperação), e a definição dos protocolos e infra-estruturas de comunicação que os agentes irão precisar para se comunicar com outros agentes (ou sistemas) [RABELO, 97].

No caso da CLIDENP, o domínio de conhecimento será uma clinica virtual para o diagnóstico e ensino da desnutrição infantil, sendo que todas as técnicas de inferência para a obtenção do diagnóstico estarão implementadas em seus agentes, que terão todas as funcionalidades dos sistemas especialistas legados a representar. Por isso, cada agente será diferente em termos de comportamentos, arquitetura e modelo.

3.2 OS SISTEMAS LEGADOS ENVOLVIDOS

Os Agentes Especialistas da CLIDENP são baseados em alguns sistemas legados que tem características diferentes, dependendo do processo de raciocínio do especialista que automatiza. A seguir, serão mostradas as principais características destes sistemas.

3.2.1 SISPAN

3.2.1.1 DEFINIÇÃO

O SISPAN – Sistema Pediátrico para Avaliação Nutricional – é um Sistema Especialista Probabilístico que utiliza a Teoria da Probabilidade para tratar a incerteza, ou seja, é uma aplicação desenvolvida para auxiliar na avaliação do estado nutricional em crianças de 0 a 2 anos de idade, com base nos sinais e sintomas e dados antropométricos (Figura 10) [KOEHLER, 98].

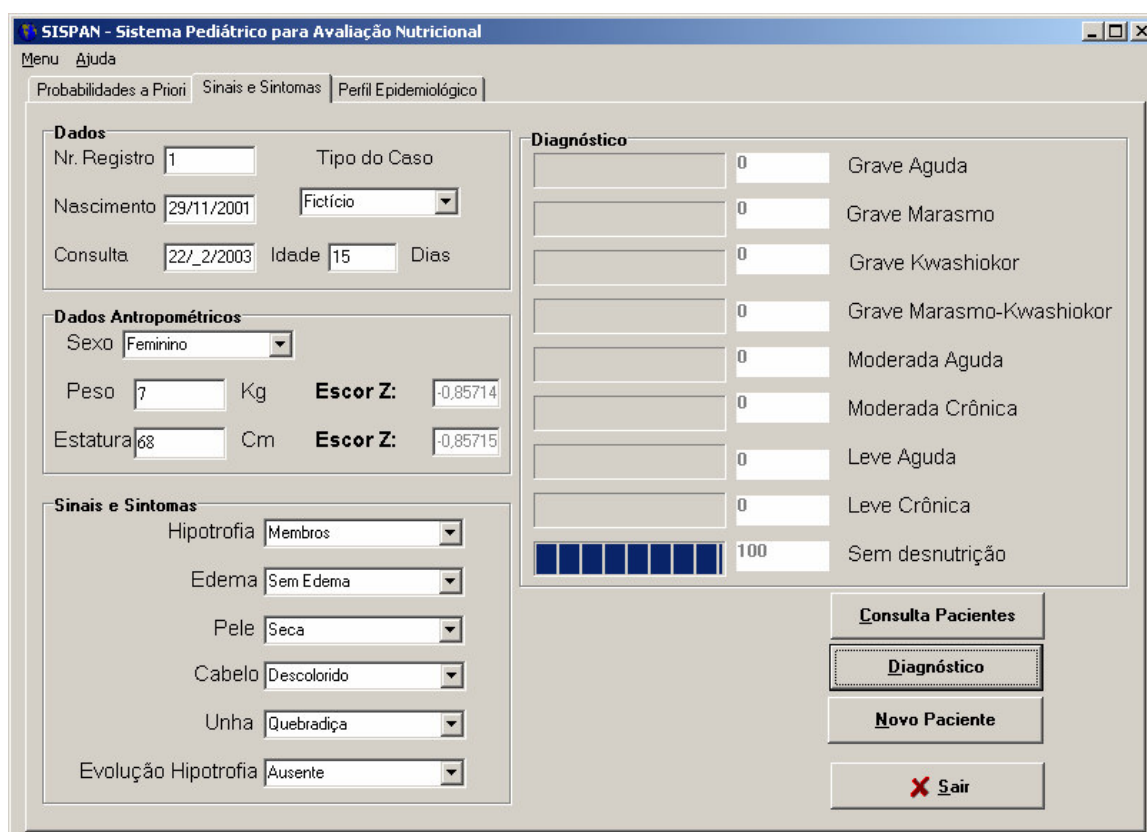


Figura 10 – Interface SISPAN

3.2.1.2 CARACTERÍSTICAS TÉCNICAS

O SISPAN foi implementado utilizando a linguagem de programação Borland C++ Builder 1.0, gerando um executável que integra a DLL principal do software Netica

(Shell SEP) e salvando tanto os dados de sinais e sintomas e antropométricos quanto às hipóteses diagnósticas encontradas em uma base de dados Paradox 7.0. A Interface principal está acoplada ao Sistema Especialista Probabilístico e a utilização deste sistema só pode ser feita em sistemas operacionais da Microsoft (MS Windows).

3.2.1.3 BASE DE CONHECIMENTO

O SISPAN possui uma base de conhecimento para Avaliação do Estado Nutricional, ilustrada na Figura 11.

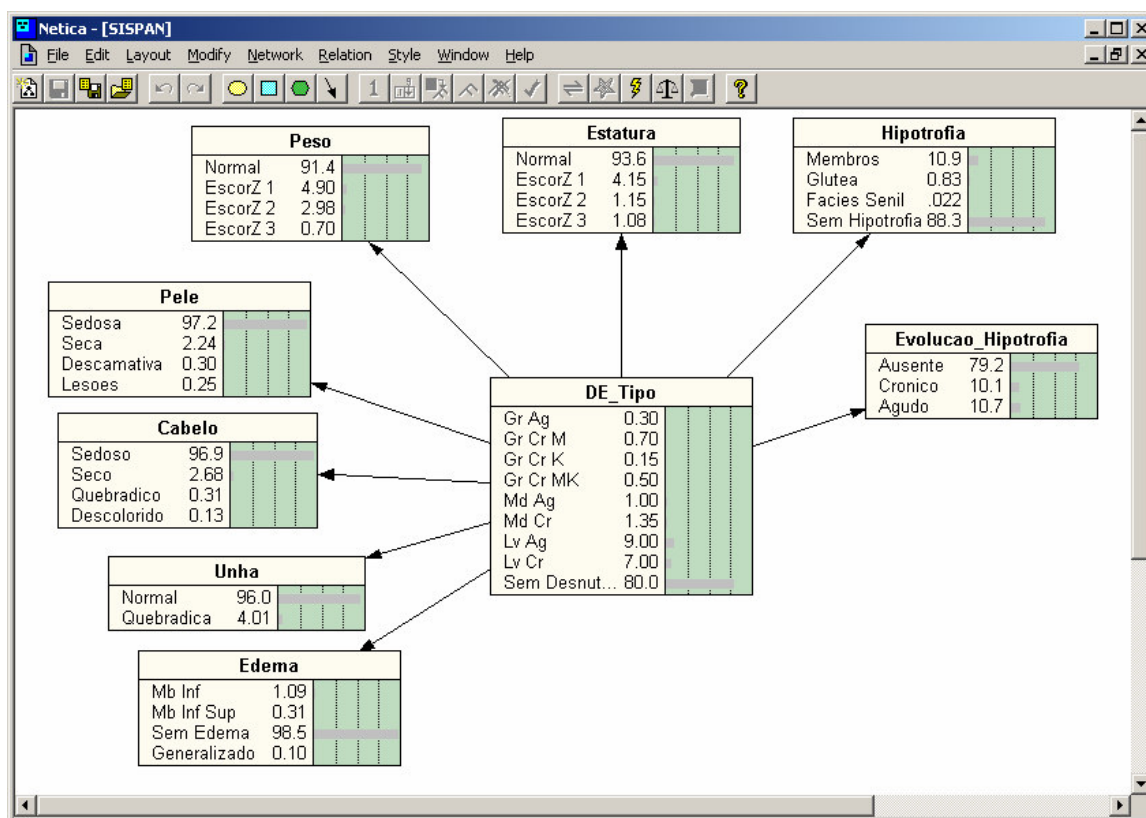


Figura 11 – Representação gráfica da base de conhecimento do SISPAN

A base de conhecimento do SISPAN, possui oito nós, que representam as variáveis para a aquisição do conhecimento especialista, com as probabilidades condicionais *a priori* atribuídas. São variáveis da RB SISPAN:

- ◆ *Edema;*
- ◆ *Pele;*

- ◆ *Cabelo;*
- ◆ *Unha;*
- ◆ *Estatutura;*
- ◆ *Peso;*
- ◆ *Hipotrofia;*
- ◆ *Evolução da Hipotrofia.*

Estas variáveis estão diretamente ligadas por arcos ao nó central, indicando as dependências causais entre as variáveis e as hipóteses diagnósticas.

3.2.2 SACI

3.2.2.1 DEFINIÇÃO

O SACI – Sistema de Apoio na Avaliação de Distúrbios do Crescimento Infantil – é um Sistema Especialista Probabilístico que também utiliza a Teoria da Probabilidade para tratar a incerteza, ou seja, é uma aplicação desenvolvida para auxiliar na avaliação de distúrbios do crescimento infantil em crianças de 0 a 2 anos de idade, com base nos sinais, dados antropométricos e diagnóstico prévio, realizado pelo SISPAN. Outra função importante deste sistema especialista, é a construção do gráfico da curva de crescimento infantil para crianças da mesma faixa etária (Figura 12) [SIMÕES, 01].

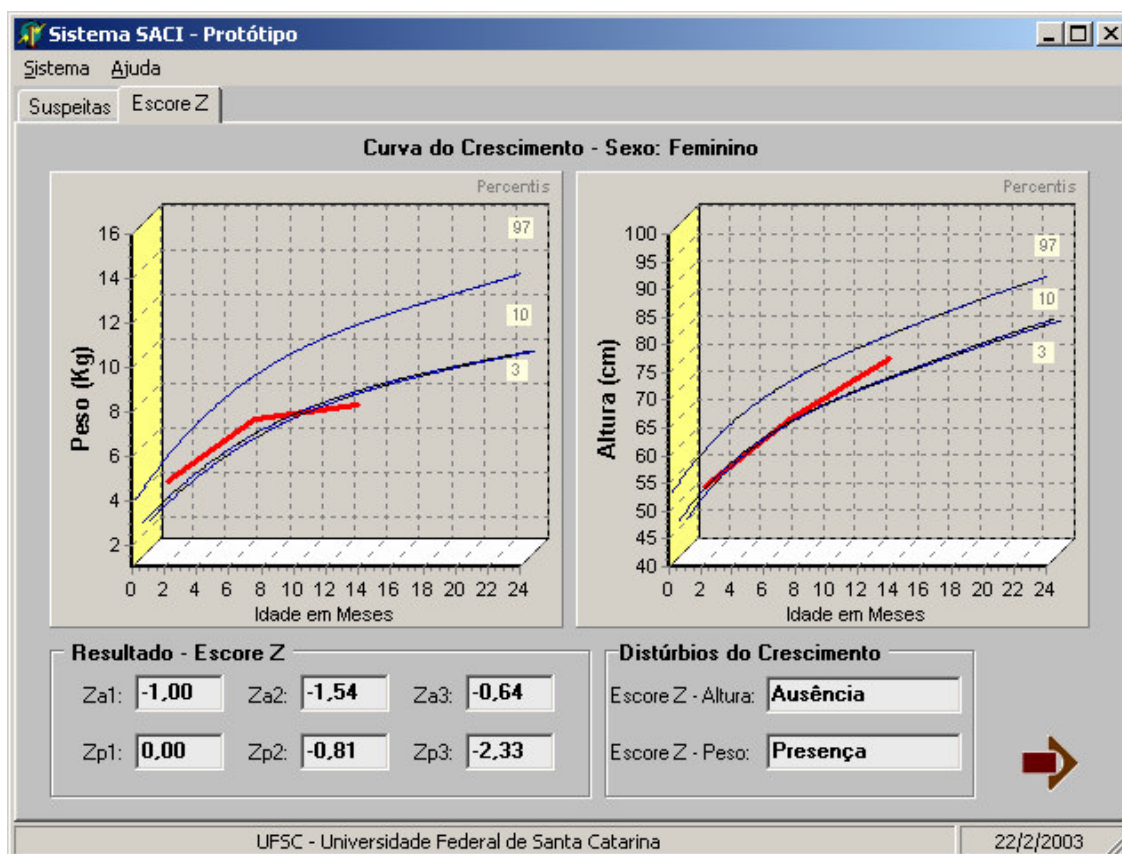


Figura 12 – Interface SACI

3.2.2.2 CARACTERÍSTICAS TÉCNICAS

O SACI foi implementado utilizando a linguagem de programação Borland Delphi 5.0, gerando um executável que integra a DLL principal do software Netica (*Shell SEP*). A Interface principal está acoplada ao Sistema Especialista Probabilístico e a utilização deste sistema só pode ser feita em sistemas operacionais da Microsoft (MS Windows). Este sistema não possui base de dados própria, o que implica na reentrada dos dados do paciente necessários para as solicitações de diagnóstico.

3.2.2.3 BASE DE CONHECIMENTO

O SACI possui uma base de conhecimento para Avaliação do Estado Nutricional, ilustrada na Figura 13.

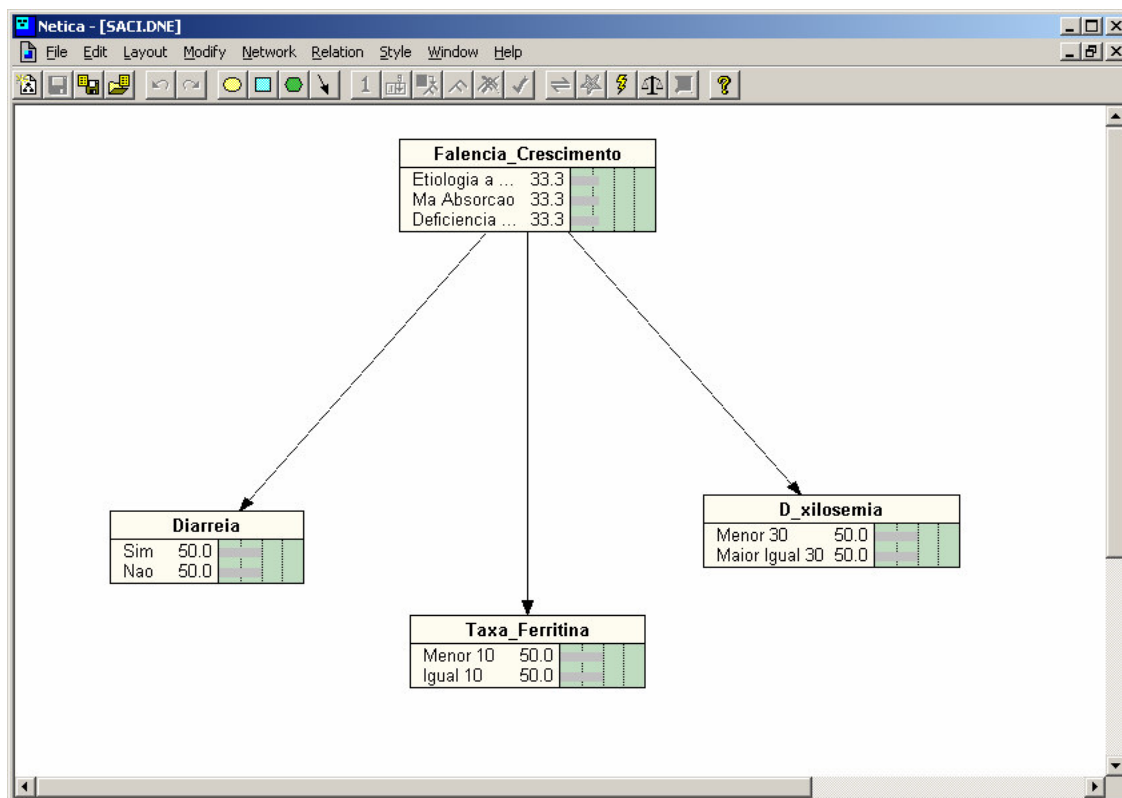


Figura 13 – Representação gráfica da base de conhecimento do SACI

No caso do SACI, sua base de conhecimento possui apenas 3 nós, que representam as variáveis para a aquisição do conhecimento especialista, com as probabilidades condicionais *a priori* atribuídas. São variáveis da RB SACI:

- ◆ ***D-Xilosemia;***
- ◆ ***Taxa de Ferritina;***
- ◆ ***Diarréia.***

Estas variáveis estão diretamente ligadas por arcos ao nó central, indicando as dependências causais entre as variáveis e as hipóteses diagnósticas.

3.2.3 SANEP

3.2.3.1 DEFINIÇÃO

O SANEP – Sistema Especialista Probabilístico de Apoio a Nutrição Enteral Pediátrica – é um Sistema Especialista Probabilístico que utiliza a Teoria da Probabilidade para

tratar a incerteza, ou seja, é uma aplicação desenvolvida para auxiliar no cálculo da dieta enteral, uma das formas de tratamento nutricional para crianças de 0 a 2 anos de idade, com base nos sinais e sintomas e dados antropométricos e nos diagnósticos do SISPAN e SACI (Figura. 14) [REIS, 01].

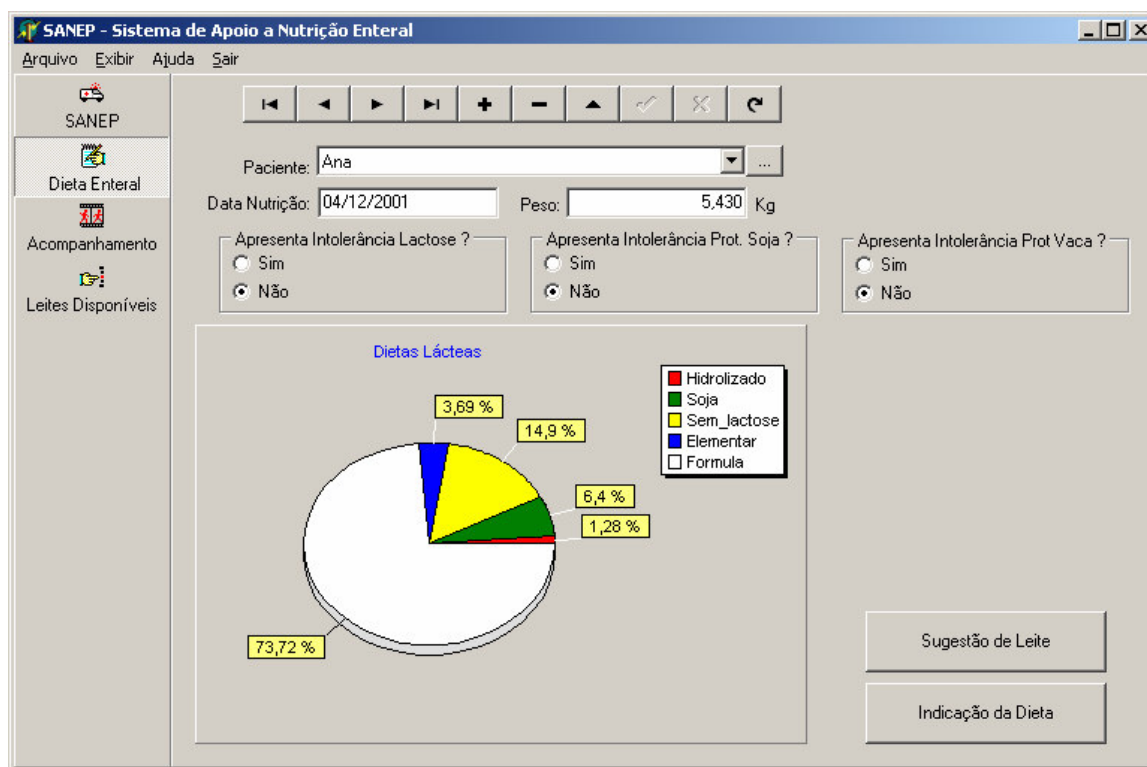


Figura 14 – Interface SANEP

3.2.3.2 CARACTERÍSTICAS TÉCNICAS

O SANEP também foi implementado utilizando a linguagem de programação Borland Delphi 5.0, gerando um executável que integra a DLL principal do software Netica (*Shell SEP*) e salvando tanto os dados de sinais e sintomas e antropométricos quanto às hipóteses diagnósticas encontradas em uma base de dados Paradox 7.0. A Interface principal está acoplada ao Sistema Especialista Probabilístico e a utilização deste sistema só pode ser feita em sistemas operacionais da Microsoft (MS Windows).

3.2.3.3 BASE DE CONHECIMENTO

O SANEP possui uma base de conhecimento para o Cálculo da Dieta Enteral, ilustrada na Figura 15.

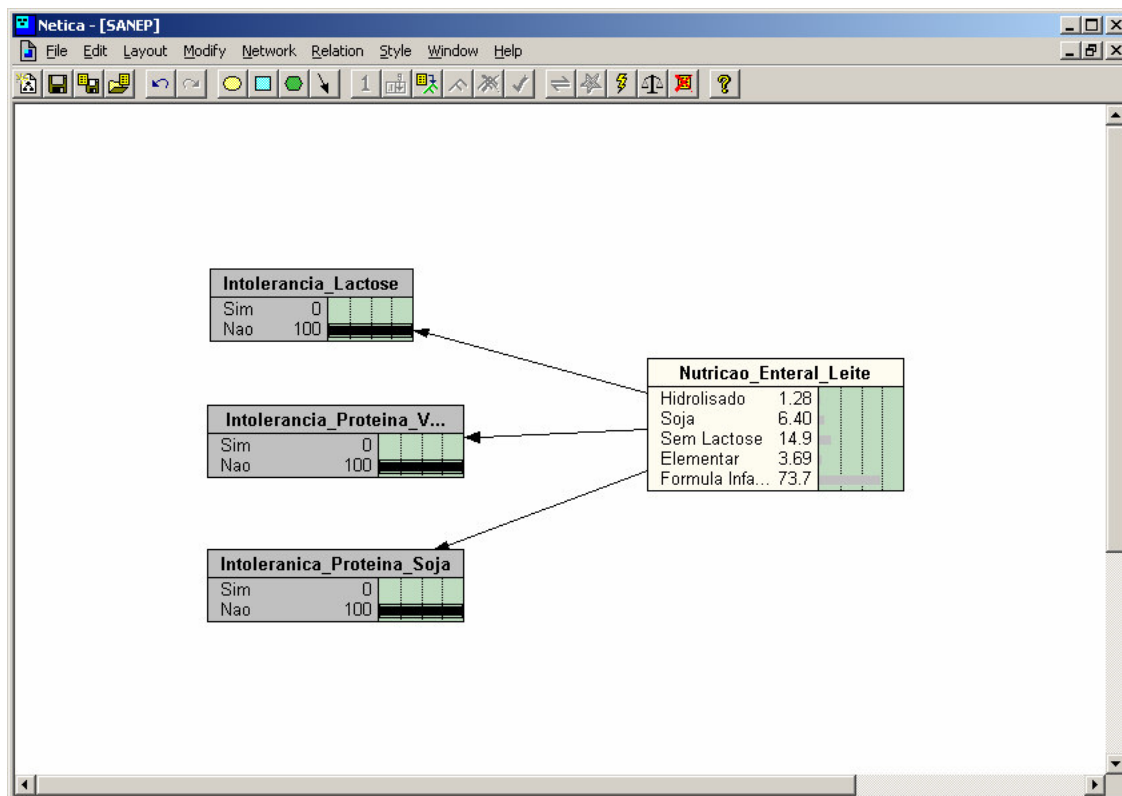


Figura 15 – Representação gráfica da base de conhecimento do SANEP

A base de conhecimento do SANEP, assim como a do SACI, possui três nós, que representam as variáveis para a aquisição do conhecimento especialista, com as probabilidades condicionais *a priori* atribuídas. São variáveis da RB SANEP:

- ◆ **Intolerância Lactose;**
- ◆ **Intolerância à Proteína da Vaca;**
- ◆ **Intolerância à Proteína da Soja.**

Estas variáveis estão diretamente ligadas por arcos ao nó central, indicando as dependências causais entre as variáveis e as hipóteses diagnósticas.

3.2.4 SANPAP

3.2.4.1 DEFINIÇÃO

O SANPAP – Sistema Especialista Probabilístico de Apoio a Nutrição Parenteral Pediátrica – ainda não foi desenvolvido, mas será um Sistema Especialista Probabilístico que utiliza a Teoria da Probabilidade para tratar a incerteza, ou seja, é uma aplicação desenvolvida para auxiliar no cálculo da dieta parenteral, uma das formas de tratamento nutricional para crianças de 0 a 2 anos de idade, com base nos sinais e sintomas e dados antropométricos e nos diagnósticos do SISPAN e SACI. A CLIDENP deverá ser desenvolvida, prevendo uma futura integração deste sistema [REIS, 01].

3.3 MODELO DE AGENTES CLIDENP

A partir do estudo dos sistemas legados, ilustrados na seção anterior, determinou-se que o modelo de agentes da CLIDENP será estruturado de acordo com o processo de obtenção do diagnóstico, representando assim, o funcionamento de todo o sistema especialista integrado através da cooperação de agentes prevista no sistema multiagente. A comunicação estará por conta de uma linguagem única, através de mensagens simples, contendo os parâmetros de entrada e saída de cada agente. Estas mensagens serão transmitidas através do protocolo HTTP e armazenadas no banco de dados MySQL da aplicação multiagentes.

Numa análise geral, pode-se dizer que um Sistema Multiagente “ideal” provê mecanismos complexos e robustos de suporte à tolerância à falhas, assim como bibliotecas de “comportamentos” codificados e agentes para plataformas heterogêneas (tais como agentes Java), entre outras facilidades [GESZYCHTER, 99].

Porém, uma componente de suporte e tolerância à falhas robusto e complexo não será implementada no Sistema Multiagente CLIDENP, devido ao fato do sistema tornar-se mais pesado e complexo de implementar, mudandoos objetivos da pesquisa, conforme é proposto. A CLIDENP será construída com um mecanismo reduzido de tolerância à falhas, que não deverá traduzir-se em queda de desempenho da aplicação. Basicamente, este mecanismo fará o salvamento dos agentes (e assim de seus estados vigentes), para evitar a perda de informações manipuladas e de seus estados, diretamente na base de

dados. Por fim, para diminuir as possíveis falhas nos agentes devido à necessidade de comunicação (ponto crítico em sistemas multiagentes), toda a comunicação entre os agentes deverá ser realizada através do banco de dados, evitando assim a manipulação direta de sockets, padronizando o armazenamento das mensagens e mantendo assim um histórico das interações entre os agentes, possibilitando uma recuperação mais eficaz do estado vigente.

Os Agentes CLIDENP são agentes de *software* (Figura 16), não móveis, distribuídos em um ambiente heterogêneo, pois são agentes *Web*, que apresentam aquelas três propriedades primordiais de um agente (descritas na seção 2.4.1).

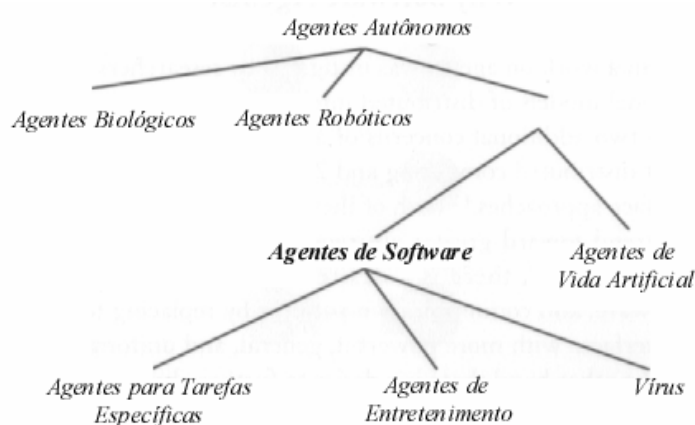


Figura 16 - Enquadramento dos Agentes CLIDENP [BRADSHAW, 97]

Por estarem assim classificados, são funções dos agentes CLIDENP:

- (a) desempenhar tarefas a favor do usuário;
- (b) treinar ou ensinar o usuário;
- (c) ajudar diferentes usuários colaboradores;
- (d) monitorar eventos e processos.

Para aumentar o alcance e utilização da CLIDENP, o sistema estará disponível a partir de um *site* que será disponibilizado por esta universidade. Optou-se realizar este projeto na plataforma *Web*, pois a *Internet* oferece alguns benefícios e facilidades para a aplicação, no momento de se desenvolver um Sistema Médico ou de apoio à Medicina de uma forma geral. Esses fatores facilitadores podem ser divididos em três categorias principais:

- ◆ **Facilidade para a integração com os sistemas legados:** em geral, nas instituições de saúde, há um grande número de sistemas de informação de diferentes fornecedores, com diferentes interfaces, vocabulários e modelo de dados, bem como em plataformas de *hardware* e *software* diferentes. São os chamados sistemas legados (*Legacy Systems*). A tecnologia *Web* permite a integração de todos esses sistemas, através de uma interface comum, utilizando os navegadores da *Web* (*Web browsers*), tais como o *Netscape* e o *Internet Explorer*, que acessam os bancos de dados legados, de forma distribuída e, de certa forma, unificam os modelos de dados. Obviamente, isso não é uma tarefa trivial, pois a padronização de interfaces e mapas entre os diversos vocabulários é um trabalho árduo, que exige muito tempo e o acompanhamento de especialistas. Por outro lado, somente a tecnologia *Web* é capaz de oferecer essa integração de forma mais rápida e simples, quando comparada a outras soluções.
- ◆ **O aumento do consumo:** nos últimos anos, com o aumento da poder de compra dos consumidores, um grande mercado surgiu em resposta a essa demanda. Esse crescente aumento de serviços de comércio eletrônico em outras áreas faz crescer as expectativas por serviços dessa natureza no mercado de saúde, sendo a *Internet* o meio pelo qual esses podem chegar facilmente ao alcance dos novos consumidores.
- ◆ **Características próprias da Tecnologia *Web*:** a tecnologia *Internet* oferece muitas características que favorecem a rápida penetração em todos os aspectos do negócio, tendo a área da saúde a oportunidade de aproveitá-las. Características como acessibilidade via *Web browsers* com rápida ou nenhuma curva de aprendizado, rapidez de distribuição visto que não se têm preocupações com questões relacionadas ao sistema operacional das máquinas clientes e, sendo os sistemas baseados na *Web* mais fáceis e mais baratos para desenvolver e manter do que sistemas client-server e mainframes tornam a tecnologia *Web* extremamente adequada para qualquer tipo de empresa ou instituição.

Para atender as necessidades de interface e ajuda dos sistemas especialistas, os usuários da CLIDENP poderão contar com outros dois agentes: o agente Interface e o agente Ajuda. Estes agentes serão construídos para realizar as seguintes tarefas:

- (a) tornar o sistema mais amigável voltada aos processos (agentes) envolvidos;
- (b) permitir a navegação em outros *sites* médicos, no intuito de melhorar o treinamento do usuário;
- (c) realizar a comunicação entre o usuário e os outros agentes do sistema;
- (d) explicar como todo o resultado foi obtido;
- (e) realizar um treinamento de acordo com o sistema especialista escolhido;
- (f) responder às dúvidas do usuário, a qualquer momento, durante a utilização da CLIDENP.

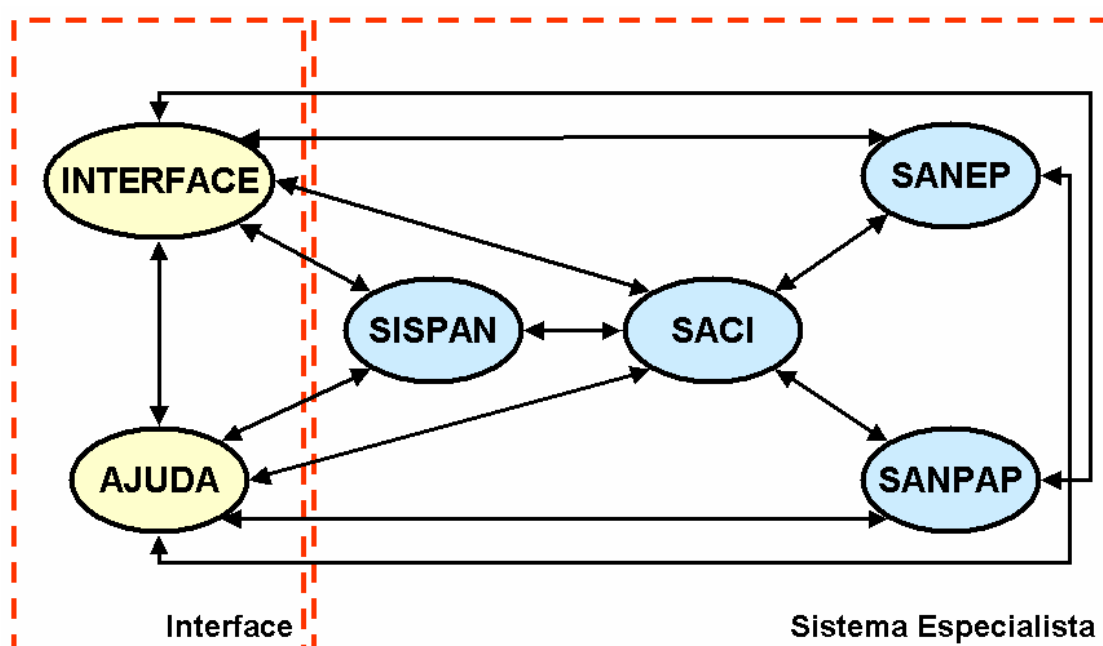


Figura 17 – Modelo de Agentes da CLIDENP

Conforme definido na Figura 17, os agentes CLIDENP estarão divididos em dois grupos de agentes, de acordo com a sua função dentro do sistema multiagente:

- ◆ **agentes de interface:** tem a função de atender o usuário, iniciar e terminar processos, coletar dados dos agentes especialistas e tirar dúvidas do usuário;
- ◆ **agentes especialistas:** realizam as atividades ordenadas, cooperam entre si para a obtenção do resultado, retornam os resultados aos agentes de interface.

O grupo dos agentes de interface é formado por dois agentes diferentes: o agente interface e o agente de ajuda. Já o grupo dos agentes especialistas, é formado por quatro

agentes diferentes: o agente SISPAN, o agente SACI, o agente SANEP e o agente SANPAP. Este último grupo deverá contemplar também, uma integração com o Netica, pois as bases de conhecimento dos sistemas especialistas representados foram construídas utilizando redes bayesianas, por meio da *Shell* Netica, com o objetivo de realizarem o raciocínio inferencial por meio do Teorema de Bayes.

De acordo com todas as definições apresentadas, o sistema multiagente CLIDENP possui algumas características importantes de salientar a partir das classificações associadas a ele. Por ser um MAS federado, possui agentes de interface que conhecerão parcialmente as habilidades dos agentes especialistas com o objetivo de coordenar suas ações, durante toda a utilização do sistema. Além disso, é um MAS democrático, pois apesar de ter dois grupos bem distintos de agentes, todos possuem o mesmo nível hierárquico dentro do modelo proposto. Por fim, é um MAS fechado, pois a composição da comunidade de agentes é fixa, devido ao fato de representar a união dos sistemas especialistas envolvidos, mas permitindo que todos os serviços associados aos agentes especialistas estejam sempre disponíveis ao usuário.

A base de dados que será utilizada na CLIDENP, além de viabilizar a comunicação e o mecanismo de tolerância à falhas, também deverá ser utilizada para guardar informações das consultas nutricionais realizadas no sistema, dos diagnósticos realizados para acompanhamento futuro do sistema e de sua utilização.

A comunicação dos agentes da comunidade CLIDENP será realizada através de uma linguagem simples, livre de contexto, sobre a qual uma semântica particular possa ser introduzida, ou mesmo um protocolo do tipo KQML possa ser implantado.

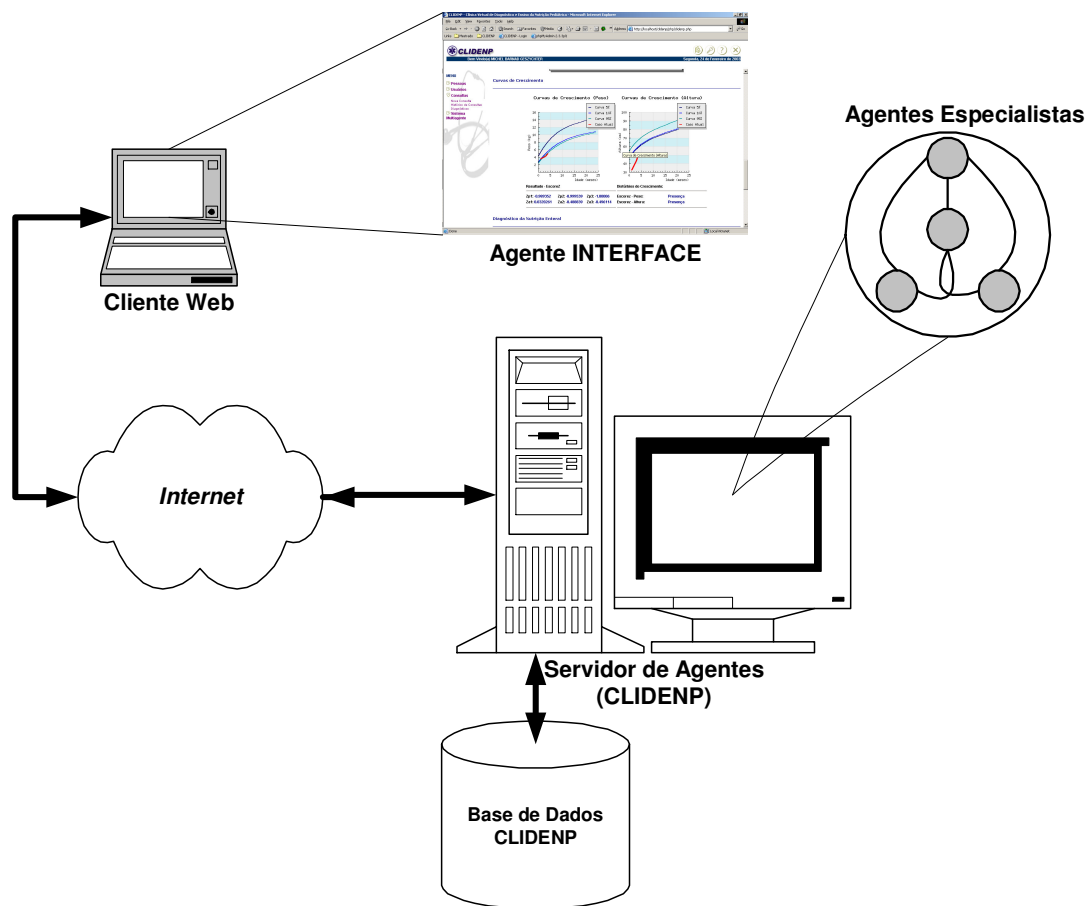


Figura 18 – Utilização da CLIDENP via Internet

A combinação de todos os conceitos associados a este sistema proposto, tanto os claramente enunciados como: sistemas multiagentes e sistemas especialistas quanto os não enunciados claramente como: sistemas distribuídos, comércio eletrônico, banco de dados e etc. com as tecnologias e paradigmas de programação a serem utilizados: orientação a objetos, orientação a agentes, ergonomia, ajuda, etc. demonstram a relevância computacional da implementação, desta proposta, ou seja, um sistema multiagente que relaciona sistemas especialistas probabilísticos de apoio ao diagnóstico em forma de clínica virtual para ser acessado via *Web*.

4 IMPLEMENTAÇÃO DA CLIDENP

Com base no estudo realizado no Capítulo 2, e na especificação de uma modelagem escalável de um sistema multiagente especialista que solucionasse o problema da interligação dos sistemas legados SISPAN, SACI, SANEP e SANPAP proposta no Capítulo 3, foi desenvolvido um protótipo, denominado CLIDENP – Clínica Virtual para Diagnóstico e Educação em Nutrição Pediátrica.

O desenvolvimento deste protótipo foi inspirado no paradigma de programação orientada por agentes (AOP) [SHOHAM, 93], e no paradigma de orientação a objetos (OOP). No paradigma de programação orientada por agentes, o problema central de um dado ambiente é dividido em partes, não necessariamente iguais, para que o mesmo seja resolvido por agentes específicos construídos para tal finalidade. Estes agentes são projetados com autonomia, independência e agência para que possam resolver os problemas, tanto de forma isolada, quanto interagindo com os outros agentes da comunidade.

Já no paradigma de Orientação a Objetos, o ambiente a ser modelado é observado segundo uma ótica de objetos ali existentes. Uma vez identificadas às entidades que deverão fazer parte do projeto, uma perspectiva de classes e objetos é montada. Não existem regras fixas para a determinação do que deve ser visto como objeto ou como a classificação deve ser feita. A principal questão a ser observada é que o modelo deve estar o mais próximo possível da realidade, ou seja, de como as coisas realmente existem e quais as relações existentes entre elas.

Como o objetivo deste projeto é a integração dos sistemas especialistas probabilísticos legados, em forma de um sistema de agentes “inteligentes” (sistema multiagente), há a preocupação que o mesmo seja capaz de reunir e representar, de forma adequada, os diversos modos de pensar e agir do especialista em seus diversos domínios de aplicação, e, desta forma, ser capaz de tratar a imprecisão. Além disso, a forma de representar o conhecimento deve ser padronizada, para que o mesmo seja trocado e tratado por

agentes especialistas distintos sem perda de informação. Todos os sistemas especialistas legados envolvidos na construção do sistema multiagente utilizam inferência probabilística (redes bayesianas).

O diálogo paciente-médico é o componente mais básico e essencial do atendimento clínico, sendo o processo através do qual há uma ativa troca de saber entre as partes quanto ao status quo e tendências futuras do estado de saúde do primeiro e uma negociação entre ambos quanto a como proceder a partir da constatação final obtida. [REIS, 2001]

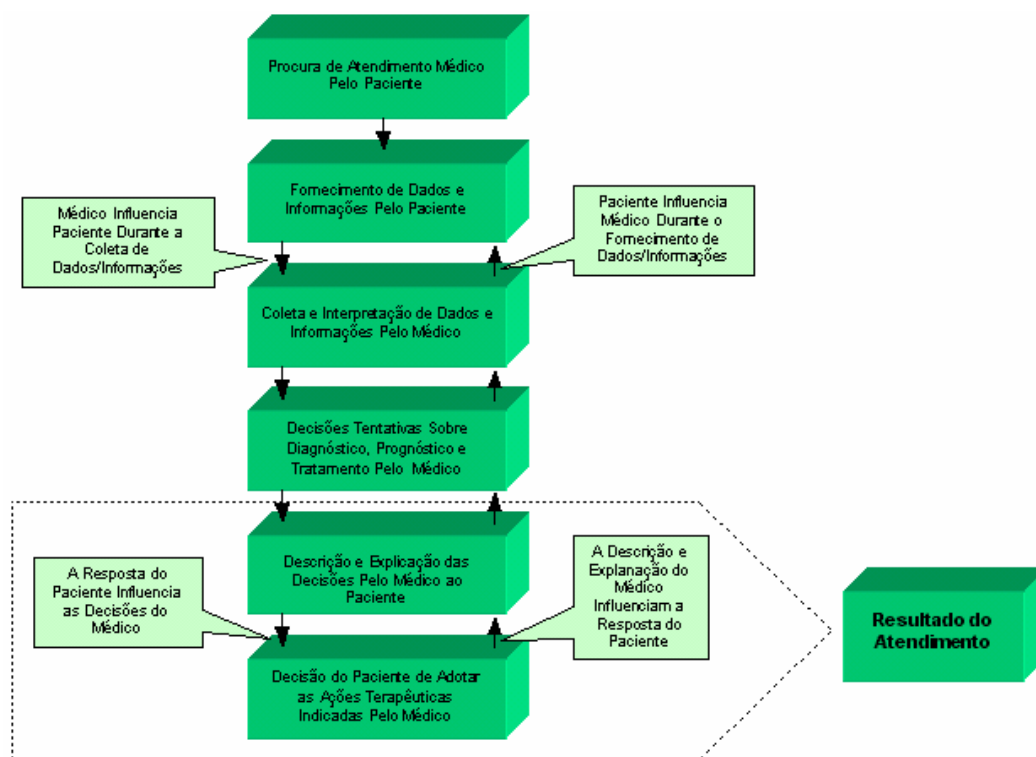


Figura 19 - Etapas para o atendimento/diagnóstico médico(IATROS)

Naturalmente, as etapas descritas na Figura 19 representam apenas uma de inúmeras possibilidades, mas apresenta a vantagem de explicitar objetivamente as etapas da consulta, o fluxo do saber (dados e informações), as instâncias de tomada de decisão e as interações ocorridas [REIS, 2001]. Dessa forma, o sistema multiagente especialista deverá possuir uma boa interface de acesso, prevendo a melhor representação da relação paciente-médico e suas implicações.

Para facilitar a Implementação da CLIDENP, esta etapa foi dividida em duas sub-etapas, onde foram definidos o modelo funcional do sistema contendo suas principais entidades, seus relacionamentos e modos de funcionamento e, também, um modelo de implementação da ferramenta baseado no modelo funcional e no paradigma de Orientação a Objetos.

4.1 MODELO FUNCIONAL

O modelo funcional da CLIDENP, foi concebido considerando-se essencialmente a necessidade de realizar o diagnóstico da Desnutrição Energético Protéica (DEP), da Falência do Crescimento (FC) e da indicação da melhor dieta por Nutrição Enteral (NE) para crianças de 0 a 2 anos de idade, prevendo a integração entre os agentes através de uma base de dados, e de uma interface que pudesse ser acessada via Web.

Para tal, foram definidas cinco entidades fundamentais (Agentes) que compõe a CLIDENP:

- ◆ **INTERFACE:** esta é a entidade responsável por permitir ao usuário: a) acessar o sistema multiagente especialista localmente ou remotamente, pois é uma interface desenvolvida para a Web; b) inserir os dados relevantes aos pacientes, médicos e consultas; c) estimular os outros agentes a sugerirem um diagnóstico adequado ao caso selecionado na base de dados; e ainda, de uma forma gráfica e interativa, d) gerenciar o ciclo de vida do MAS CLIDENP, influenciando diretamente sobre os outros agentes especialistas;
- ◆ **SISPAN:** é a entidade mais importante do MAS CLIDENP, pois todo o funcionamento dos outros agentes especialista do sistema multiagente depende dela. Por ser um Agente Especialista, possui acoplada ao seu código fonte, a DLL do Netica para interação com a rede baysiana de diagnóstico da Desnutrição Energético Protéica (DEP). Esta entidade estará rodando junto ao servidor web da aplicação multiagente e, ao ser comandado pelo Agente INTERFACE, deverá consultar a base de dados de consultas, submeter às evidências da DEP retiradas da base de dados à rede baysiana e interagir junto

ao Agente INTERFACE para mostrar a sugestão de diagnóstico da DEP realizada pela rede baysiana.

- ◆ **SACI:** esta entidade tem como tarefa realizar o diagnóstico da falência do crescimento a partir do diagnóstico da DEP sugerido pelo Agente SISPAN. Por também ser um Agente Especialista, possui a DLL do Netica em seu código fonte para interação com a rede baysiana, neste caso a Rede Baysiana para Diagnóstico da Falência do Crescimento (FC). Assim como os outros agentes especialistas, esta entidade estará rodando junto ao servidor web da aplicação multiagente e, ao ser estimulado pelo Agente INTERFACE e pelo Agente SISPAN, deverá consultar a base de dados de consultas, submeter às evidências da FC retiradas da base de dados à rede baysiana e interagir junto ao Agente INTERFACE para mostrar a sugestão de diagnóstico da FC realizada pela rede baysiana.
- ◆ **SANEP:** é a entidade do MAS CLIDENP responsável por, a partir dos diagnósticos sugeridos pelos Agentes Especialistas SISPAN e SACI, sugerir o melhor leite para a Nutrição Enteral do caso selecionado. Assim como os outros Agentes Especialistas, possui em seu código fonte, a DLL do Netica para interação com a rede baysiana da Nutrição Enteral (NE). Esta entidade estará rodando junto ao servidor web da aplicação multiagente e, ao ser solicitado pelo Agente INTERFACE ou pelo Agente SACI, deverá consultar a base de dados de consultas, submeter às evidências retiradas da base de dados à rede baysiana e interagir junto ao Agente INTERFACE para mostrar a sugestão de diagnóstico realizada pela rede baysiana.
- ◆ **SANPAP:** é a entidade do MAS CLIDENP responsável por, a partir dos diagnósticos sugeridos pelos Agentes Especialistas SISPAN e SACI, sugerir o melhor leite para a Nutrição Enteral do caso selecionado. Assim como os outros Agentes Especialistas, possui em seu código fonte, a DLL do Netica para interação com a rede baysiana da Nutrição Parenteral (NP). Esta entidade estará rodando junto ao servidor Web da aplicação multiagente e a partir do momento que o Agente INTERFACE ou Agente SACI requisitarem, deverá consultar a base de dados de consultas, submeter às evidências retiradas da base de dados à

rede baysiana e interagir junto ao Agente INTERFACE para mostrar a sugestão de diagnóstico realizada pela rede baysiana.

A Figura 20 mostra que as entidades fundamentais estarão interagindo entre si e com o Banco de Dados CLIDENP, ilustrando o funcionamento cooperativo das entidades INTERFACE (Agente INTERFACE), SISPAN, SACI, SANEP e SANPAP (Agentes Especialistas) para a obtenção das hipóteses diagnósticas previstas nos agentes especialistas.

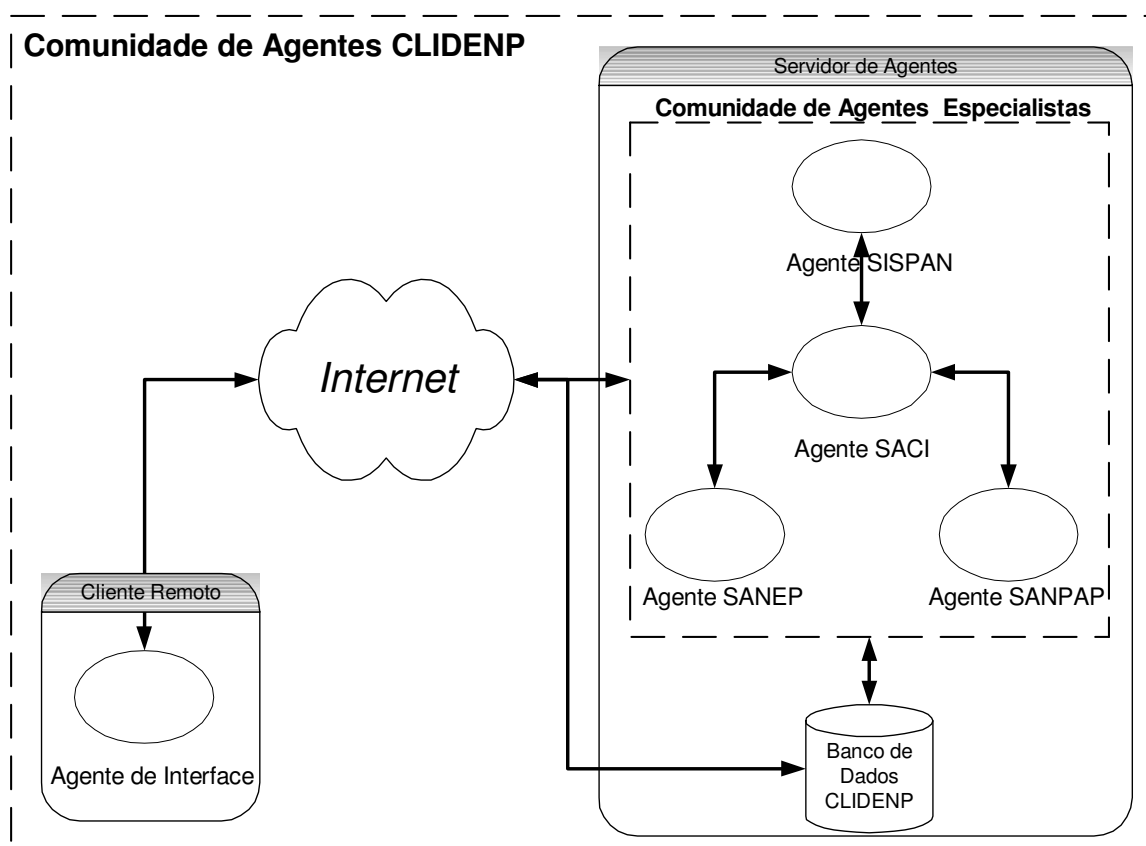


Figura 20 – Modelo Funcional das Entidades Fundamentais CLIDENP

Apesar da CLIDENP ser um sistema multiagente fechado e estático, foi necessário desenvolver uma outra entidade no MAS CLIDENP para representar a figura do agente facilitador, necessário à Sistemas Multiagentes que instanciam agentes remotamente, ou que possuem agentes móveis. Esta entidade foi implementada devido à necessidade do sistema multiagente possuir um mecanismo de tolerância à falhas e gerenciamento dos

agentes. Assim o Agente GERENTE, possui funções muito específicas e também de relevante importância durante o ciclo de vida do sistema multiagente em questão, sendo um agente facilitador, totalmente dedicado a auxiliar o usuário no gerenciamento dos agentes especialistas no servidor de agentes da CLIDENP, através do Agente INTERFACE.

Basicamente, ele gerencia o funcionamento dos outros agentes, atuando diretamente sobre os mesmos a pedido do usuário, permitindo a execução, o encerramento e a checagem do funcionamento de um dado agente especialista. O Agente GERENTE pode receber suas ordens localmente através da sua própria interface ou remotamente a partir de mensagens enviadas a partir do Agente INTERFACE.

A figura 21 mostra as interações entre as entidades do sistema multiagente CLIDENP. Note-se que a comunidade de agentes continua interagindo como ilustrado na Figura 20, preservando as características do sistema multiagente especialista, apenas adicionando-se as características de gerenciamento dos agentes, realizadas entre os agentes INTERFACE e GERENTE.

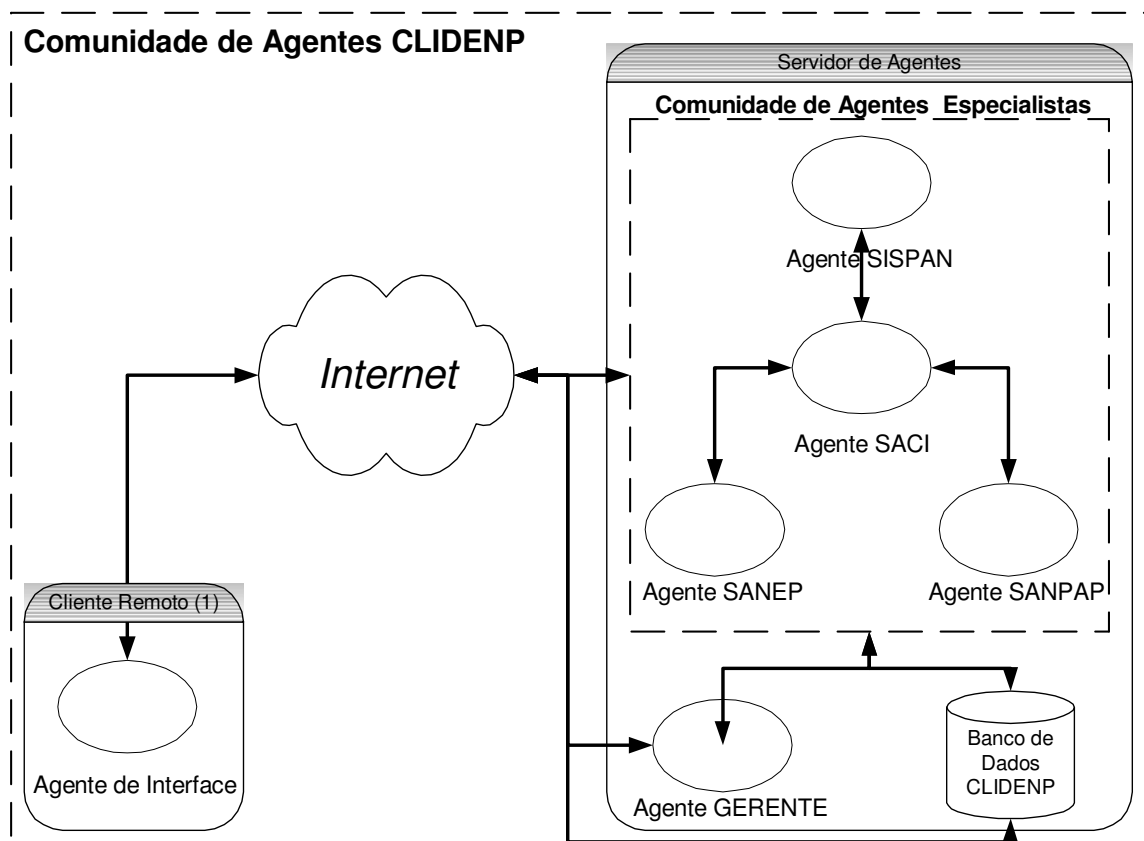


Figura 21 – Modelo Funcional CLIDENP

Este modelo foi desenvolvido para ser executado a partir de um servidor web, com o objetivo de estar sempre disponível na Internet, tornando-se mais que um sistema multiagente convencional, ou seja, consolidando o conceito de sistema multiagente web pronto a reponder requisições de diversos agentes de interface espalhados na grande rede. A idéia principal é que os agentes especialistas estarão à disposição de quaisquer agentes que se comuniquem com o sistema multiagente em questão para resolver os problemas inseridos na base de dados.

Fisicamente, a comunidade de agentes fica distribuída, sendo que os Agentes Especialistas estão com seus processos presos ao sistema do servidor de agentes, e os Agente INTERFACE e AJUDA, com seus processos alocados nas máquinas clientes dos usuários, interagindo com o usuário do *Browser* local, como mostra a Figura 16. Diversos Agentes INTERFACE poderão estar instanciados em diversos clientes,

solicitando diagnósticos e informações aos agentes especialistas instanciados no servidor de agentes, formando assim diversas comunidades de agentes CLIDENP distintas, doravante chamadas de comunidades lógicas de agentes.

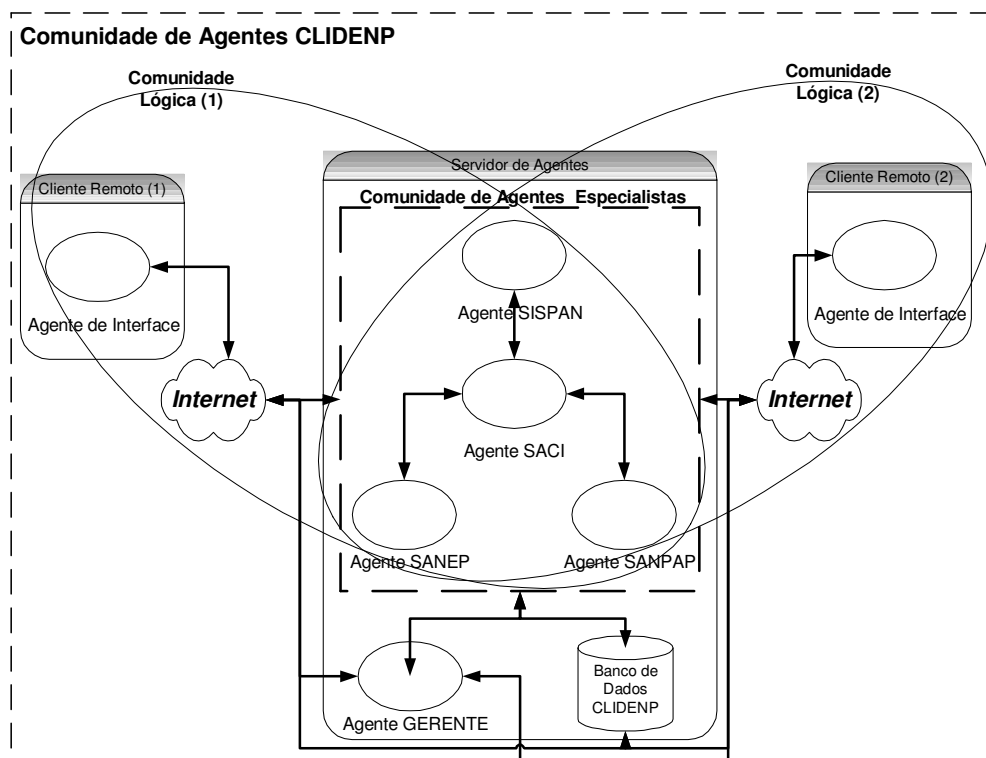


Figura 22 – Comunidades Lógicas de Agentes da CLIDENP

A decisão de não limitar o número de agentes de interfaces no sistema multiagente permite um melhor aproveitamento dos recursos disponibilizados pelos agentes especialistas, pois num ciclo normal de um sistema multiagente especialista como o CLIDENP, os agentes podem ficar ociosos por algum tempo, sendo então aproveitados para sugerir os diagnósticos solicitados por outra comunidade lógica presente no MAS CLIDENP.

A figura 23 mostra como é feita a comunicação entre as entidades Agentes (Especialistas ou Interface), utilizando o banco de dados como caixa de entrada para as mensagens recebidas e como destino das mensagens enviadas de cada agente do sistema, separando-se as correspondências dentro do banco de dados através do (agente) destinatário das mesmas. No Sistema Multiagente CLIDENP, as mensagens não são

enviadas e recebidas diretamente pelos próprios agentes, ou seja, todo o processo de comunicação dos agentes é realizado através do banco de dados.

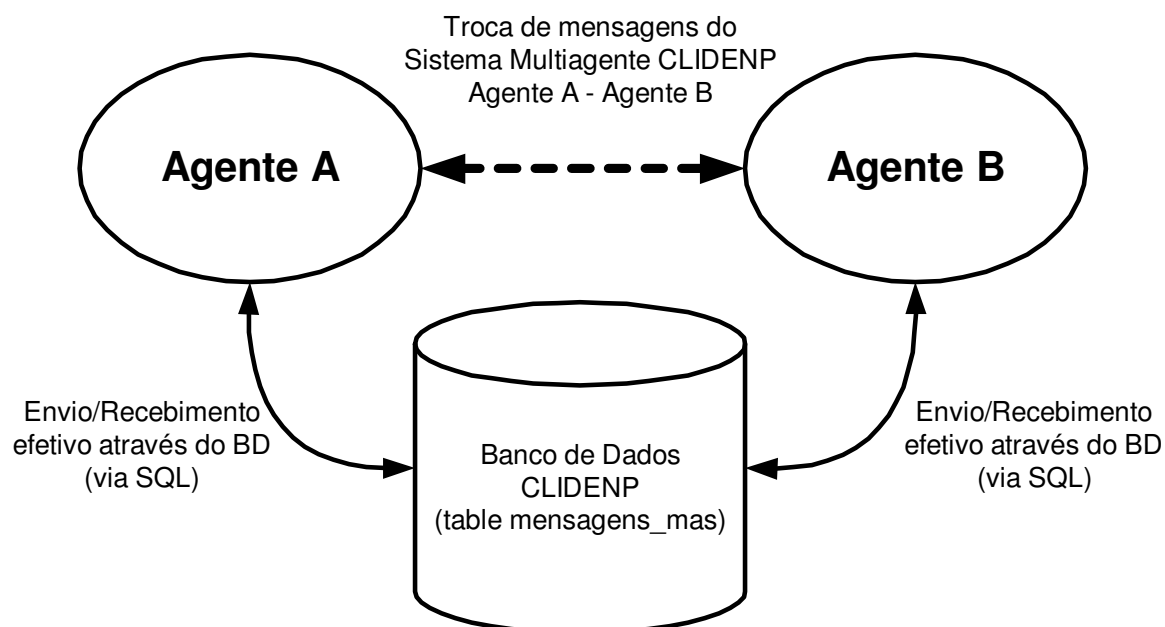


Figura 23 – Modelo Funcional da Comunicação dos Agentes CLIDENP

Desta maneira, os agentes se comunicam sempre através do Banco de Dados CLIDENP. Assim para enviar mensagens, o Agente A insere o conteúdo das mensagens em uma tabela que é conhecida e acessada por todos os Agentes CLIDENP. Para receber suas mensagens, o Agente B acessa periodicamente a mesma tabela de mensagens e consome apenas as informações que são destinadas a ele.

A seguir, será explanado como o sistema multiagente funcionalmente especificado neste capítulo foi implementado.

4.2 MODELO DE IMPLEMENTAÇÃO

Como as entidades da CLIDENP, descritas na seção anterior, são agentes construídos a partir de sistemas legados, foi realizada uma padronização utilizando os paradigmas AOP e OOP, gerando assim modelos de objetos e agentes distintos.

Esta padronização foi necessária, porque as entidades envolvidas possuíam diferenças consideráveis, apesar de terem sido desenvolvidas e preparadas para serem integradas.

Isto ocorreu devido aos seguintes fatos:

- ◆ foram utilizadas linguagens de programação distintas;
- ◆ não foi seguido um padrão de implementação (as implementações foram realizadas por pessoas diferentes);
- ◆ nem todas as entidades comunicavam-se com uma base de dados;
- ◆ a informação deveria estar centralizada em uma única base de dados.

Apesar das diferenças encontradas em cada um dos sistemas especialistas legados, ao padronizá-los na forma de agentes, foi visível a proximidade de algumas de suas características (atributos e métodos), já que são agentes, possuem uma interface gráfica comum, comunicam-se (enviam e recebem mensagens), realizam determinadas tarefas (a partir das mensagens recebidas) e interagem com os outros agentes da comunidade ou com a Base de Dados.

Ao se concluir que uma padronização global era inevitável, os paradigmas de orientação a objetos (OOP) e de programação orientada por agentes (AOP) foram utilizados.

Outros, aspectos também foram considerados:

- ◆ Linguagem de programação;
- ◆ Plataforma de cada agente;
- ◆ Acesso à base de dados comum;
- ◆ Escalabilidade do Sistema.

4.2.1 O AGENTE PADRÃO

Para iniciar a referida padronização, foi desenvolvido um conjunto de classes que contém as principais características e ações necessárias a um agente da CLIDENP, que doravante será chamado de Agente Padrão. O objetivo disto foi permitir que a conversão dos sistemas legados envolvidos fosse facilmente ajustável às necessidades

de um sistema multiagente. Assim, o Agente Padrão permite que sejam derivados outros tipos de agentes, bastando apenas realizar uma especialização de suas classes principais.

Além disso, a linguagem de programação foi padronizada para *Borland C++ Builder 5.0*, o que facilitará a manutenção de todos os agentes existentes na CLIDENP, bem como a inserção de novos agentes. O banco de dados e a forma de acesso (ODBC) também foram padronizados. Foi utilizado o banco de dados *MySQL for Windows*, pois além de oferecer bons recursos tecnológicos (software), permite também a interação dos novos agentes criados a partir do Agente Padrão com um Agente Web, por exemplo, entre outras funcionalidades.

A representação do Agente Padrão é composta das seguintes classes de objetos:

Agentes: esta é a classe principal do Agente Padrão. Possui todos os atributos e métodos necessários para a correta representação das características de uma aplicação agente. Será a partir desta classe que todos os outros agentes da CLIDENP serão derivados através do conceito de herança da OOP.

Interfaces: é uma classe que atua como intermediária entre o usuário e os métodos da classe Agentes. Sua principal função é exibir os atributos do agente que o usuário achar relevantes, bem como uma lista de suas atividades fornecendo informações e resultados de operações, se for o caso. Por fim, permite o gerenciamento de algumas tarefas do agente, tais como, a leitura de suas mensagens e seu encerramento.

Mensagens: é a classe que recebe os atributos da mensagem atual enviada ao agente. Sua função é permitir o acesso rápido às informações e ordens contidas na última mensagem recolhida da caixa de entrada. Esta classe não possui métodos, pois os métodos de gerenciamento das mensagens recebidas estão contidos na classe Agentes, já que um objeto desta classe é atributo de Agentes

A figura 24 mostra a interação entre as classes de objetos do Agente Padrão. O modelo de objetos detalhado pode ser visto no Anexo C.

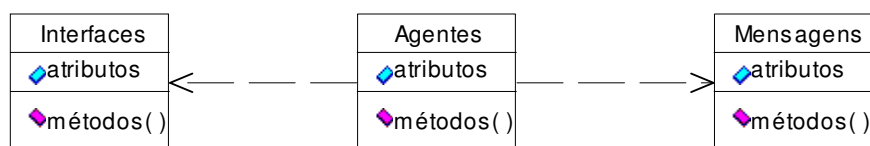


Figura 24 – Modelo de Objetos do Agente Padrão

Desta forma, foi possível padronizar a implementação tanto do modelo do sistema multiagente especialista quanto das entidades a seguir explicadas.

4.2.2 O AGENTE GERENTE

O Agente GERENTE é um agente facilitador que permite ao usuário a gerenciar o sistema multiagente especialista através do Agente INTERFACE. São funções do Agente GERENTE:

- ◆ Executar os agentes da comunidade multiagente;
- ◆ Encerrar os agentes da comunidade multiagente;
- ◆ Fazer a checagem por solicitação do usuário, nos agentes;

Para desempenhar estas funções o Agente GERENTE foi modelado utilizando todas as classes do Agente Padrão, juntamente com a criação uma nova classe chamada AgentesGerentes que é uma especialização da classe Agentes.

A representação do Agente GERENTE é composta das seguintes classes de objetos:

- ◆ AgentesGerentes: será a classe principal do Agente GERENTE. Além de possuir todos os atributos e métodos herdados do Agente Padrão, possui também os métodos necessários para o atendimento de suas funções primordiais, descritas anteriormente.
- ◆ InterfacesGerentes: esta é a classe de interface com o usuário do Agente Gerente. Possui todos os atributos e métodos herdados da classe Interfaces e foi especializada com o objetivo de permitir o gerenciamento dos agentes sem a necessidade do Agente INTERFACE, se for o caso.

Mensagens: utiliza a mesma classe do Agente Padrão.

A Figura 25 ilustra o modelo de objetos do Agente GERENTE. O modelo de objetos detalhado pode ser visto no Anexo C.

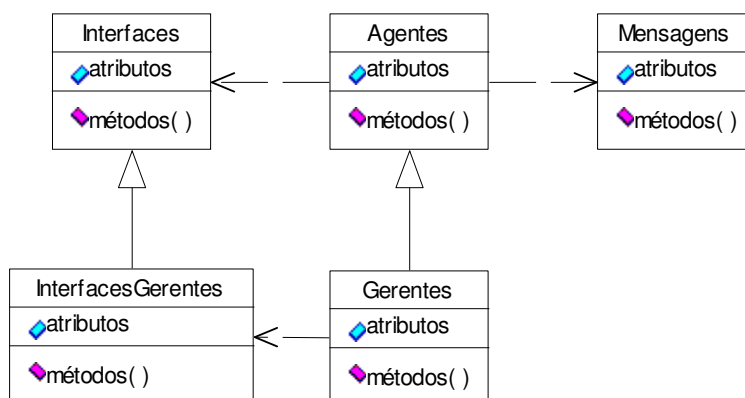


Figura 25 – Modelo de Objetos do Agente GERENTE

4.2.3 OS AGENTES ESPECIALISTAS

Os Agentes Especialistas são os agentes mais importantes do Sistema Multiagente CLIDENP. Isto porque são eles que calculam as hipóteses diagnósticas a partir das informações recolhidas pelo Agente INTERFACE e salvas na base de dados. São funções dos Agentes Especialistas:

- ◆ Obedecer às ordens dos Agentes de INTERFACE e GERENTE;
- ◆ Comunicar-se com os outros Agentes Especialistas da comunidade
- ◆ Buscar na base de dados CLIDENP as informações necessárias ao seu funcionamento;
- ◆ Realizar o cálculo das hipóteses diagnósticas;
- ◆ Salvar as probabilidades encontradas na base de dados CLIDENP.

Apesar dos Agentes Especialistas não executarem as mesmas tarefas, eles têm as mesmas funções junto ao Sistema Multiagente. Sendo assim, os modelos de objetos dos Agentes SISPAN, SACI, SANEP e SANPAP foram modelados utilizando todas as classes do Agente Padrão, juntamente com a criação de uma nova classe chamada AgentesEspecialistas, que é uma especialização da classe Agentes.

A representação dos Agentes Especialistas é composta das seguintes classes de objetos:

- ◆ Consultas: esta é a classe com responsável por armazenar, informar e manipular os dados referentes à consulta (dados pessoais e secretos que o especialista obtém do paciente). Outra função desta classe é permitir o acesso rápido as informações das consultas sem precisar contatar a base de dados mais de uma vez.
- ◆ Diagnósticos: é a classe que recebe as hipóteses diagnósticas da consulta atual, quando o agente termina sua execução principal. Sua função é permitir o armazenamento temporário das informações diagnósticas até que o agente decida salva-las na base de dados CLIDENP.
- ◆ Especialistas: esta é a classe que possui métodos necessários para o atendimento de suas funções primordiais, todos contruídos baseados nos sistemas legados e previamente descritos, além de suas funções individuais de cálculo das probabilidades diagnósticas.
- ◆ AgentesEspecialistas: é a classe principal dos Agentes Especialistas. É a especialização das classes Agentes e Especialistas, possuindo assim todos os atributos e métodos herdados, transformando o seu objeto em um
- ◆ Interfaces: utiliza a classe do agente padrão, com as mesmas funções.
- ◆ Mensagens: utiliza a classe do agente padrão, com as mesmas funções.

A Figura 26 mostra o modelo de objetos dos Agentes Especialistas. O modelo de objetos detalhado pode ser visto no Anexo C.

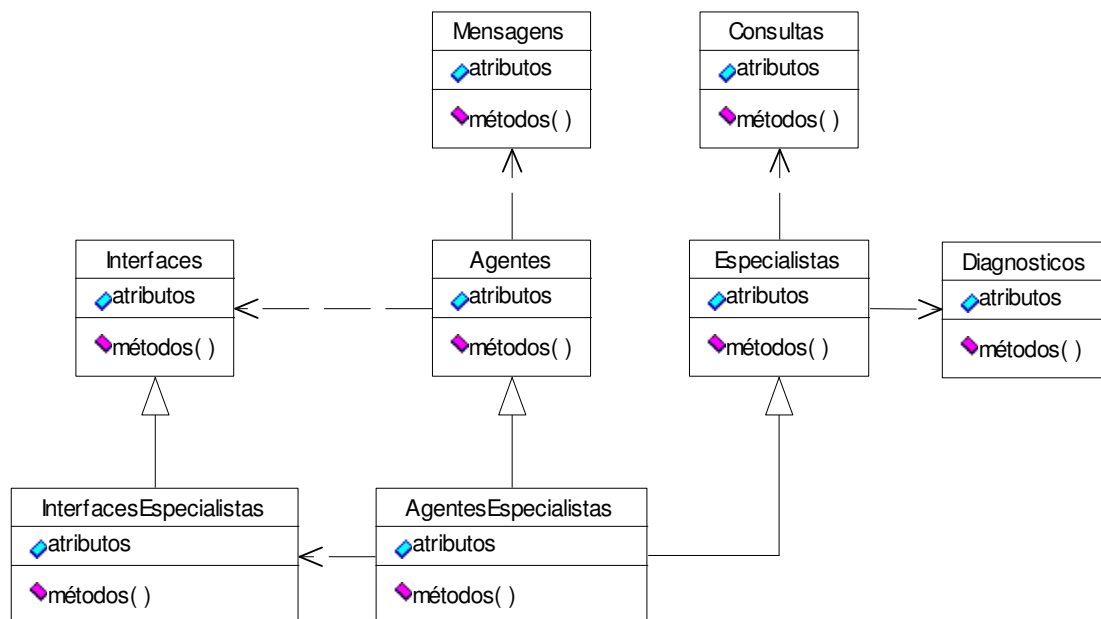


Figura 26 – Modelo de Objetos dos Agentes Especialistas

4.2.4 O AGENTE INTERFACE

A interface como usuário é uma peça chave de qualquer sistema (multiagente ou não), pois é a partir dela que o usuário interage com o sistema. Ela deve ser amigável e de fácil utilização, tendo um módulo de ajuda, se possível.

A partir deste conceito foi construído o Agente INTERFACE. Diferentemente dos outros agentes deste Sistema Multiagente, este é um agente funções voltadas para o usuário, a saber:

- ◆ Estar sempre disponível para a interação do usuário;
- ◆ Interagir com os Agentes Especialistas em nome do usuário;
- ◆ Ser o intermediário entre os Agentes Especialistas e o usuário;
- ◆ Permitir que as solicitações possam ser exibidas ao usuário de forma adequada.

Para estar sempre disponível para a interação do usuário, o Agente INTERFACE foi desenvolvido para ser um Agente Web. Foram utilizadas as linguagens de programação HTML, PHP e Java para que ele fosse totalmente acessível via Internet. Desta forma, o Agente INTERFACE fica disponível através de um endereço do tipo

<http://clidenp.inf.ufsc.br> a qualquer usuário da Internet, mediante uma autenticação por login e senha.

O banco de dados utilizado foi o MySQL, para tornar as interações com a base de dados CLIDENP mais rápidas, pois o PHP possui rotinas nativas de acesso a ele e por ser um Banco de Dados leve.

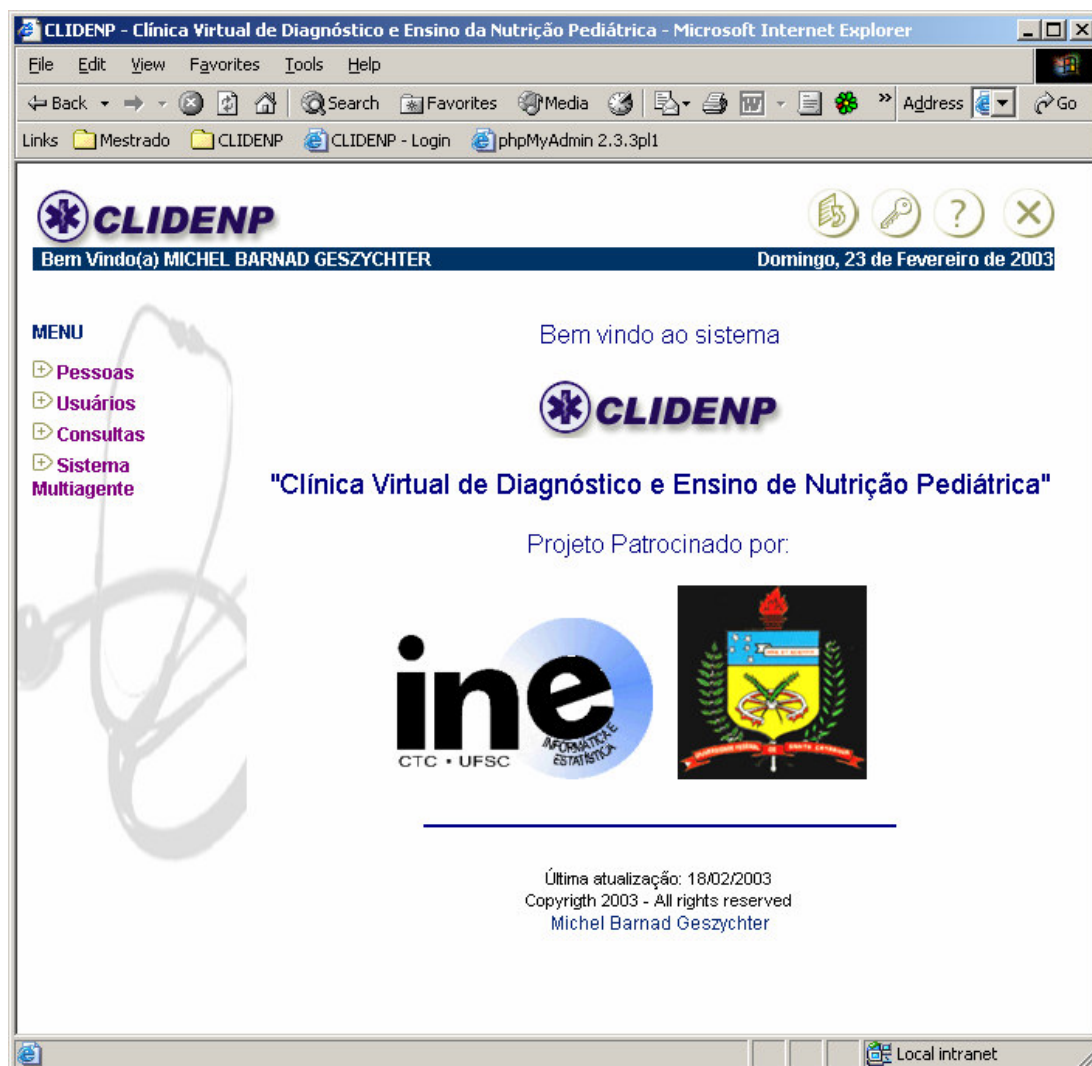


Figura 27 – Interface do Agente INTERFACE

Por ser um Agente Web, optou-se por não modelá-lo com OOP. Seu modelo, foi construído, com base no Diagrama de Fluxo de Dados (DFD) para acesso às informações e solicitação de diagnóstico. A Figura 28 mostra o DFD do Agente INTERFACE.

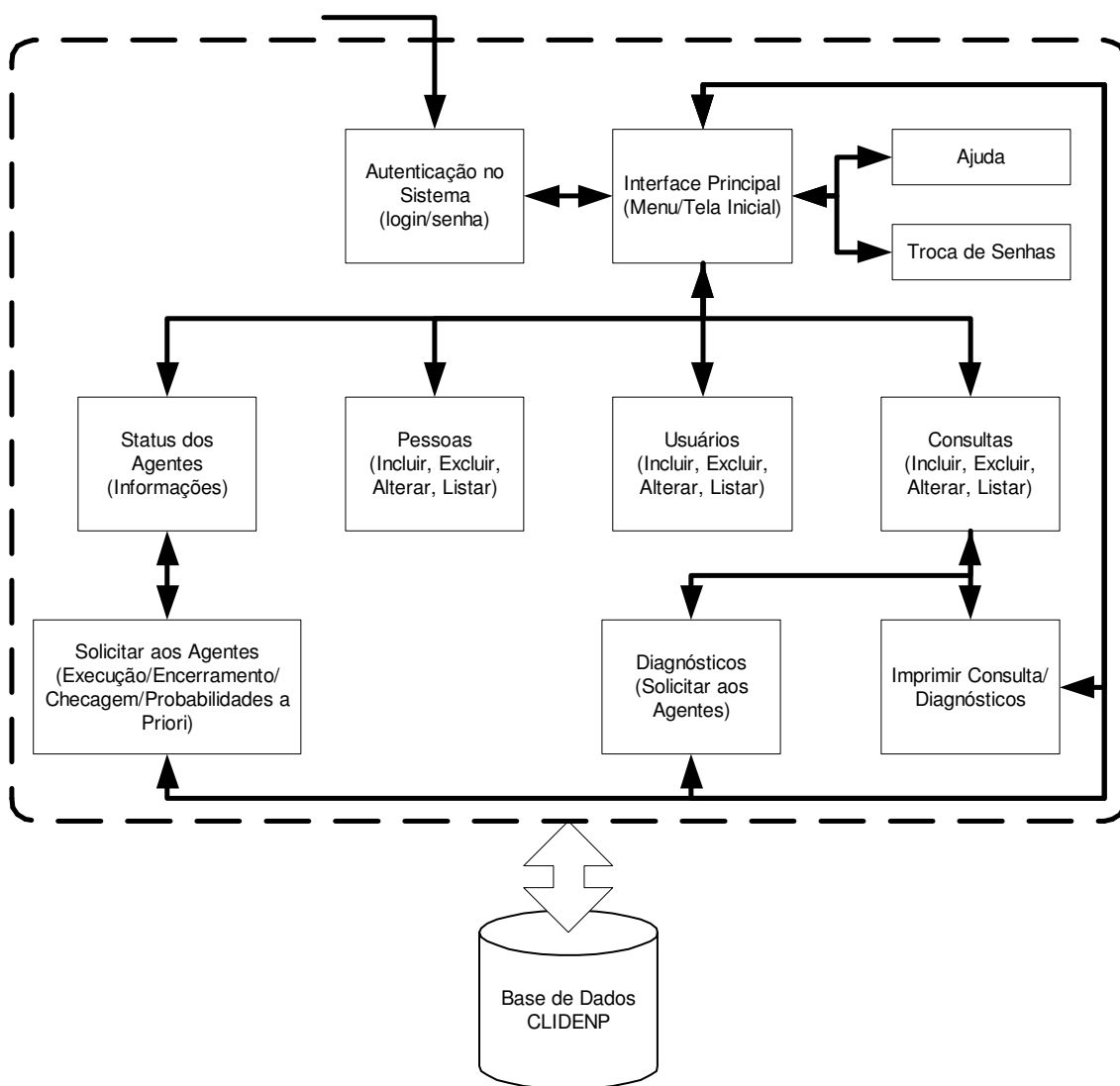


Figura 28 – Diagrama de Fluxo de Dados do Agente INTERFACE

Assim ele possui rotinas (scripts PHP) que realizam todas as suas tarefas, interagindo com o banco de dados e com os agentes.

4.2.5 A BASE DE DADOS CLIDENP

Para suportar todas as interações entre os agentes, manter as informações a respeito das consultas e hipóteses diagnósticas encontradas, controlar o acesso ao sistema entre outras funções do Sistema Multiagente CLIDENP, a base de dados CLIDENP foi criada

utilizando-se os conceitos de banco de dados relacional. Esta base de dados possui tabelas e estruturas responsáveis por controlar e fornecer informações aos diversos agentes/módulos do sistema.

A figura 29 mostra as entidades (tabelas) criadas para permitir o armazenamento das informações e posterior acompanhamento dos dados dos pacientes. O modelo de entidade e relacionamento completo da Base de Dados CLIDENP pode ser visto no Anexo B.

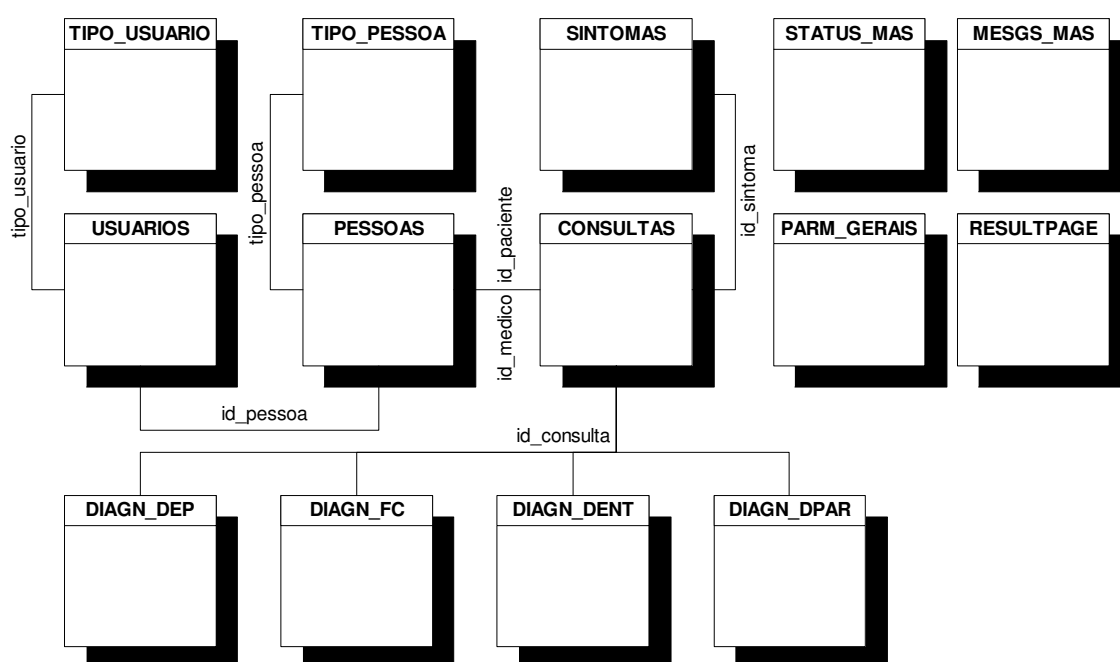


Figura 29 – Base de Dados CLIDENP

4.3 CICLOS DO SISTEMA MULTIAGENTE

Nas seções anteriores foram apresentados o modelo funcional e o modelo de implementação com os modelos de objetos de cada entidade presente. Nesta seção será apresentado o ciclo de vida completo do Sistema Multiagente CLIDENP.

O ciclo de vida do Sistema Multiagente CLIDENP é compreendido por três grandes fases: lançamento (execução) do sistema, interação entre os agentes e encerramento do sistema.

4.3.1 LANÇAMENTO DO SISTEMA

Esta é a primeira fase do ciclo de vida do Sistema Multiagente CLIDENP. É nela que serão instanciados os agentes especialistas da CLIDENP no Servidor de Agentes. Esta operação é solicitada pelo usuário ou através do Agente INTERFACE, ou diretamente ao Agente GERENTE. É importante salientar que ambos deverão estar aguardando o acesso do usuário (INTERFACE) e as ordens necessárias para o lançamento do sistema (GERENTE). Ao ser lançado, o agente deve salvar seus dados relevantes (probabilidades a priori, localização, etc.) na tabela `status_mas`, para que o Agente INTERFACE possa realizar o acompanhamento do Sistema Multiagente CLIDENP.

Por ser um agente web, o Agente INTERFACE não participa do lançamento. Isto porque um sistema ou agente web, está sempre “ligado”, ou seja, sempre disponível para utilização através do servidor de páginas (Web). Isto também ocorre com o Agente GERENTE, já que ele é quem irá instanciar os processos dos Agentes Especialistas, e por isso já deverá estar sendo executado no Servidor dos Agentes.

O lançamento do sistema pode ocorrer de duas formas:

- ◆ **Lançamento Completo:** Realizado através do próprio Agente GERENTE e/ou solicitado pelo Agente INTERFACE, este lançamento instancia processos para todos os Agentes Especialistas no momento inicial, com apenas uma ordem do usuário;
- ◆ **Lançamento Individual:** Realizado através do próprio Agente GERENTE e/ou solicitado pelo Agente INTERFACE, este lançamento instancia processos apenas para os Agentes Especialistas que o usuário desejar. Poderá ser realizado a qualquer momento durante o ciclo de vida, sendo necessária uma ordem do usuário para cada lançamento. Esta funcionalidade permite que o usuário instancie parcialmente o Sistema Multiagente CLIDENP, se desejar.

Como pôde ser visto, o lançamento é realizado apenas para os Agentes Especialistas (SISPAN, SACI, SANEP, SANPAP). Pode-se concluir então que a fase de lançamento do sistema é muito importante, pois influirá diretamente nas interações entre as entidades fundamentais nas outras fases do ciclo de vida do Sistema Multiagente CLIDENP.

4.3.2 INTERAÇÃO ENTRE OS AGENTES

Após a fase de lançamento do sistema ter sido concluída com sucesso, ou seja, o Sistema Multiagente CLIDENP está sendo executado, torna-se possível começar a utilização dos recursos implementados nos Agentes Especialistas. Inicialmente, o sistema está aguardando as solicitações dos diversos usuários, conectados através do Agente INTERFACE.

Basicamente, a interação dos agentes ocorre a partir do Agente INTERFACE, que envia as ordens para que Agentes Especialistas realizem as tarefas para as quais estão programados. Assim, uma série de sub-etapas deve ser realizada com sucesso para que a interação entre os agentes seja feita. São elas:

- ◆ **Cadastro de Pessoas:** inserção dos dados dos médicos, pacientes, usuários e outros, que deverão estar presentes no sistema para seu correto funcionamento. É importante que o cadastro dos médicos e pacientes seja realizado inicialmente, pois são dados fundamentais para o cadastro de consultas;
- ◆ **Cadastro de Consultas:** é efetivamente o cadastro das informações que o médico recolhe em uma consulta, ou seja, os dados de anamnese. É imprescindível que o médico ou o usuário desta seção do sistema, cadastre de forma correta estes dados. Será com base nestas informações que os agentes irão calcular as hipóteses diagnósticas.

Após a conclusão das duas primeiras sub-etapas, o Sistema Multiagente CLIDENP está apto a suportar todas as interações entre os agentes da comunidade do sistema atual, que

ocorrem basicamente entre o Agente INTERFACE e os Agentes Especialistas. Estas interações acontecerão mediante as seguintes situações:

- ◆ **Solicitação do Diagnóstico:** ao solicitar o cálculo das hipóteses diagnósticas, o Agente INTERFACE envia mensagem ao Agente Especialista selecionado, e este, caso precise, interage com os outros Agentes Especialistas da comunidade e retorna o resultado ao Agente INTERFACE para a exibição das respostas ao usuário;
- ◆ **Checagem da Atividade dos Agentes:** esta interação ocorre entre o Agente INTERFACE e os Agentes Especialistas ou entre o Agente GERENTE e os Agentes Especialistas para a verificação do funcionamento das entidades envolvidas.
- ◆ **Lançamento e Encerramento do sistema:** ao lançar um determinado Agente Especialista, há interações com o Agente GERENTE. Ao realizar o encerramento de um determinado agente, há uma interação entre o Agente INTERFACE e o Agente Especialista selecionado.

Todas as alterações e problemas ocorridos durante a execução do sistema, poderão, agora com o sistema lançado, ser acompanhados e até corrigidos através do Agente INTERFACE. Essas alterações serão sempre provenientes dos agentes via troca de mensagens. Além disso, todas as informações sobre o *status* atual e todas as mensagens trocadas pelos Agentes Especialistas estarão presentes na tela de “Status dos Agentes” e no relatório “Log de Mensagens” no Agente INTERFACE.

Com todas estas interações, o Sistema Multiagente CLIDENP simula o papel dos médicos em seus raciocínios e conclusões, calculando as hipóteses diagnósticas para cada solicitação cadastrada, funcionando como uma Clínica Virtual, ou seja, uma ferramenta de apoio ao diagnóstico da desnutrição infantil, auxiliando o trabalho de médicos, enfermeiros nutricionistas e demais profissionais da saúde.

4.3.3 ENCERRAMENTO DO SISTEMA

Esta é a terceira e mais simples fase do ciclo de vida do Sistema Multiagente CLIDENP. Considerando-se que o sistema foi lançado corretamente e está executando suas interações e tarefas adequadamente, esta etapa será realizada com o objetivo de finalizar todos os Agentes Especialistas, bem como o Agente GERENTE. O Agente INTERFACE não será finalizado, conforme descrito na seção 4.2.4.

As ações de encerramento do sistema podem ser realizadas através da entidade (aplicação) Agente GERENTE, que deverá enviar mensagens a todos os agentes da comunidade. Caso seja intenção do usuário realizar esta tarefa remotamente ou encerrar o Agente GERENTE no mesmo processo, será o Agente INTERFACE quem desempenhará tal função. Uma vez recebida a mensagem, o outro agente deverá salvar seus dados relevantes na base de dados CLIDENP, para permitir o controle da execução dos agentes e encerrar a sua execução imediatamente.

Logo que os agentes da comunidade do sistema atual são finalizados, independentemente do tipo de encerramento, todas as informações sobre o *status* atual e todas as mensagens trocadas pelos Agentes Especialistas estarão atualizadas na tela de “Status dos Agentes” e no relatório “Log de Mensagens” no Agente INTERFACE.

4.4 ESCALABILIDADE DO SISTEMA MULTIAGENTE

Ao integrar-se um conjunto de sistemas existentes na forma de um sistema multiagente, foram detectadas, analisadas e resolvidas diversas dificuldades e pendências mostradas até agora neste documento. Porém, a dificuldade maior de um modelo como este é, apesar do objetivo central que é a integração dos sistemas especialistas legados, manter a escalabilidade do sistema multiagente.

4.4.1 INTEGRANDO NOVOS AGENTES ESPECIALISTAS À CLIDENP

Uma das características dos sistemas multiagentes é a escalabilidade, ou seja, a capacidade de crescer tanto em número de agentes iguais para o sistema, quanto à inserção de novos agentes a posteriori. Na tentativa de fazer com que o Sistema

Multiagente CLIDENP fosse o mais escalável possível, foi previsto um roteiro para o desenvolvimento de novos agentes e integração dos mesmos ao sistema no futuro.

O roteiro para integração de novos agentes especialistas a CLIDENP é todo baseado no Agente PADRÃO, explicado anteriormente na seção 4.2.1. Este roteiro tem os seguintes passos:

- ◆ **Derivação do novo Agente:** para realizar a derivação de um novo agente no sistema, o desenvolvedor deverá proceder conforme explicado na seção 4.2. Já que a modelagem e implementação do sistema multiagente foi feita utilizando-se os conceitos da programação orientada a objetos, basta criar novas classes que herdem os atributos e métodos das classes disponíveis (Interface, Agentes, Mensagens, Consultas e Diagnósticos) especializando seus métodos conforme a necessidade – a OOP/AOP garante reaproveitamento de código e rapidez no desenvolvimento. Este procedimento vale tanto para os Agentes Especialistas quanto para outros tipos de agentes.
- ◆ **Configuração da Base de Dados:** assim que o desenvolvedor derivar seu agente, é preciso realizar algumas configurações na base de dados para que os outros agentes possam saber que existe um novo agente, e que podem se comunicar com ele. Além do mais, esta configuração é necessária também para o correto gerenciamento do ciclo de vida do Sistema Multiagente CLIDENP, já que os Agentes INTERFACE e GERENTE utilizam-se da base de dados para a comunicação o monitoramento do status dos demais agentes. Estas configurações, devem ser realizadas na tabela status_mas. Caso seja necessário, o usuário poderá criar novas tabelas na base de dados, que estarão disponíveis automaticamente para os outros agentes da comunidade.
- ◆ **Alterações no Agente INTERFACE:** é o último passo para a inserção de um novo agente no Sistema Multiagente CLIDENP e é necessário pois o Agente INTERFACE precisará enviar ordens e receber os resultados através de mensagens. Também poderá ser utilizado como base no desenvolvimento, as rotinas já existentes, pois qualquer novo agente deverá ter seu mecanismo de comunicação igual ao dos agentes já existentes.

Assim, ao concretizar e execução destes três passos, a inserção de novos Agentes (Especialistas ou não) estará completa. Estes passos são uma configuração padronizada que garante a escalabilidade do Sistema Multiagente CLIDENP.

5 CONCLUSÕES

A finalidade da Inteligência Artificial é construir programas funcionais que resolvam problemas nas mais diversas áreas de interesse. Na teoria, não importa como este processo é realizado, qual linguagem é utilizada para alcançar os objetivos ou em que máquina é feito o processamento [GESZYCHTER, 99].

Nesta dissertação foi desenvolvida uma modelagem multiagente escalável para a integração de sistemas especialistas, bem como um protótipo que evidencia a aplicabilidade desta modelagem, o Sistema Multiagente CLIDENP.

Nas etapas de desenvolvimento da CLIDENP, os objetivos previamente estabelecidos foram atingidos devido aos seguintes fatos:

- ◆ foram utilizadas as redes bayesianas para representação do conhecimento médico;
- ◆ a interface gráfica desenvolvida permitiu ao usuário uma fácil utilização do sistema;
- ◆ O Agente Web trouxe ao sistema uma alta disponibilidade na Internet;
- ◆ os relatórios e gráficos desenvolvidos possibilitaram ao usuário realizar análise e comparação sobre os dados gerados e disponibilizados no sistema;
- ◆ a base de dados permite um acompanhamento dos pacientes “testados” ao longo do tempo,
- ◆ o modelo multiagente concebido conseguiu manter a escalabilidade do sistema devido ao fato de terem sido desenvolvidos 3 agentes especialistas a partir dela sem deixar o sistema fechado,

É importante ressaltar que a integração das redes bayesianas com os códigos-objeto dos Agentes Especialistas apresentou resultados satisfatórios, já que tanto as evidências indicadas, quanto às hipóteses diagnósticas verificadas não sofreram distorções. Os Agentes Especialistas apresentaram os mesmos resultados que os Sistemas Especialistas Legados apresentam, a partir dos mesmos dados iniciais e, por conseqüência, os

mesmos resultados que são obtidos quando a rede baysiana é acessada diretamente na *shell* Netica. Por fim, o tempo de resposta de acesso à rede bayesiana foi considerado satisfatório, assim como a exibição dos resultados via Agente INTERFACE (através da Internet).

A avaliação do sistema junto aos especialistas da área (médicos) atendeu as expectativas, pois o sistema foi considerado adequado ao realizar o cálculo das hipóteses diagnósticas. Isto pôde ser comprovado através das diversas consultas reais e fictícias realizadas ao Sistema Multiagente CLIDENP. A avaliação da base de conhecimento foi considerada adequada, pois atingiu um percentual de acerto muito próximo de 100%.

Concluiu-se também, que as ferramentas de apoio ao diagnóstico são importantíssimas, pois utilizam o processamento da máquina para cálculos tornando o trabalho do médico mais rápido, resultando em um diagnóstico mais veloz, por conseqüência. A partir da modelagem e da implementação do protótipo, conforme proposto, constatou-se que o profissional da saúde pode utilizar melhor o seu tempo, preocupando-se mais com as doenças e com o diagnóstico.

De acordo com as características e tendências acima apresentadas, fica evidente o valor acrescentado com o desenvolvimento da CLIDENP, uma ferramenta de fácil utilização, oferecendo não apenas um alto nível de conforto ao usuário, mas permitindo uma rápida resposta às suas solicitações. Além disso, permite um efetivo armazenamento das informações dos pacientes ao longo do tempo e está voltada primordialmente ao ensino e diagnóstico da desnutrição infantil. Outro aspecto importante é que esta ferramenta é baseada na filosofia de derivação de sistemas, possibilitando o desenvolvimento de aplicações multiagente escaláveis, permitindo seu crescimento para contemplar áreas médicas afins.

Por tudo isto que foi pesquisado e demonstrado neste trabalho, comprova-se que a utilização de modelagens multiagentes escaláveis podem ser implementadas e utilizadas para atender a grande demanda existente nos ambientes de trabalho e acadêmico, por

modelagens capazes de fazer com que os sistemas possam trocar informações e serviços entre si, resolvendo problemas computacionais de forma distribuída; organizada em grupos de trabalho e aplicada a problemas que não podem ser resolvidos isoladamente, devido a sua complexidade ou necessidade computacional.

5.1 TRABALHOS FUTUROS

Uma série de complementos ao sistema ora implementado foi identificada ao longo do seu desenvolvimento. Neste sentido, como sugestão para trabalhos futuros, pode-se citar alguns que são considerados dos mais interessantes do ponto de vista conceitual:

- ◆ diversificar as formas de cadastro de dados e consultas do Agente INTERFACE para melhorar a visão dos usuários, bem como um controle mais eficiente de acesso às informações do Sistema Multiagente CLIDENP;
- ◆ desenvolvimento do Agente SANPAP, nos moldes dos Agentes Especialistas CLIDENP, customizando algumas de suas propriedades a partir de um código genérico já existente (Agente PADRÃO);
- ◆ permitir aos agentes criados uma independência com relação as mais diversas plataformas computacionais existentes atualmente (implementação em Java);
- ◆ desenvolvimento de um módulo de recuperação e tolerância à falhas capaz de garantir de forma automática e mais robusta a integridade do sistema e, ao mesmo tempo, facilmente configurável e gerenciável para usuários não especialistas em sistemas distribuídos e redes;
- ◆ ampliação do conhecimento das redes bayesianas utilizadas pelos Agentes Especialistas, incluindo mais variáveis ou mesmo refinando algumas das probabilidades a priori, podendo assim, ampliar a abrangência da utilização da CLIDENP;
- ◆ desenvolvimento das rotinas contidas na DLL do Netica nativamente nos Agentes Especialistas, para aumentar a velocidade do cálculo das hipóteses diagnósticas e diminuir o tamanho do executável do agente e da memória utilizada por ele;

- ◆ Utilização desta modelagem em outras áreas de conhecimento para comprovação de sua aplicabilidade em qualquer tipo de problema distribuído.

6 REFERÊNCIAS BIBLIOGRÁFICAS

ANGELONI, M., et al. **Estudo Comparativo entre Raciocínio Baseado em Casos e Redes Bayesianas Aplicados ao Diagnóstico de DEP.** In: CONGRESSO BRASILEIRO DE ENGENHARIA BIOMÉDICA, XVII, 2000. Florianópolis. Anais do XVII Congresso Brasileiro de Engenharia Biomédica.

AUER, K. **Agents.** 1995. Disponível em: <<http://www.biplane.com.au/~kauer/project/main.htm>>. Acesso em: 17/11/2002.

BARRETO, J., **Apostila de Inteligência Artificial.** Departamento de Informática, Universidade Federal de Santa Catarina, 2001. Disponível em: <<http://www.inf.ufsc.br/~barreto/IA/index.htm>>. Acesso em: 15/11/2002.

BLOIS, M.; SHORTLIFFE, E. The computer meets medicine: emergence of a discipline. In: SHORTLIFFE, E.; PERREAULT, L. **Medical informations: Computer applications in medical care.** Massachusetts: Addison- Wesley, 1990, p. 1-36.

COSTA, Augusto Cesar Pinto Loureiro da, **Expert-Coop - Um ambiente para desenvolvimento de Sistemas Multi-Agentes Cognitivos.** Departamento de Pós-Graduação em Engenharia Elétrica, Universidade Federal de Santa Catarina, 1997.

COSTA, Cláudio G.A. - **Desenvolvimento e Avaliação Tecnológica de um Sistema de Prontuário Eletrônico do Paciente, Baseado nos Paradigmas da World Wide Web e da Engenharia de Software.** 2001. Dissertação (Mestrado em Engenharia Elétrica) - Faculdade de Engenharia Elétrica e de Computação, Universidade Estadual de Campinas, Campinas, SP.

BRADSHAW, Jeffrey M., **Software Agents.** MIT Press, pp.9-18, 1997.

BOND., A.; GASSER, L. **An Analysis of Problems and Research in DAI.** Readings in Distributed Artificial Intelligence, pp.3-35, Morgan Kaufmann, 1988.

CARD. Centre for Agent Research & Development – Department of Computing and Mathematics, Manchester Metropolitan University, Manchester, United Kingdom. Disponível em: <http://www.card.mmu.ac.uk/>. Acesso em: 17/11/2002.

DEMAZEAU, Y. – **Multi-agents systems methodology**. XIV Simpósio Brasileiro de Inteligência Artificial (SBIA '98), p-172-185, Porto Alegre, 1998.

DEMAZEAU, Y. – **Introduction to Multi-Agent Systems**. Seminário UNISINOS, São Leopoldo, 2001.

FININ, T.; FRIZSON, R. - **KQML - A Language and Protocol for Knowledge and Information Exchange**, Technical Report CS-94-02, Computer Science Department, University of Maryland.

FININ, T.; LABROU, Y., and MAYFIELD, J. - **KQML as an Agent Communication Language**. MIT Press, Cambridge, 1995.

FIPA. <http://www.fipa.org>.

FLEISCHHAUER, L.I.A. **O uso da tecnologia de agentes na Programação da Produção**. 1996. Dissertação (Mestrado em Engenharia de Produção) – Programa de Pós-Graduação, em Engenharia de Produção da Universidade Federal de Santa Catarina, Florianópolis, SC.

FRANKLIN, S.; GRAESSER, A. - **Is it an Agent, or just a Program ? : A Taxonomy for Autonomous Agents**, Proceedings ECAI'96 / Third International Workshop on Agent Theories, Architectures and Languages, Springer-Verlag, 1996.

FROZZA, R.; ALVAREZ, L.O.C. **Um Ambiente para o Desenvolvimento de Sistemas Multiagentes Reativos**. Anais do XXIV Seminário Integrado de *Software e Hardware* – SEMISH. Brasília, DF, 1997. p-375-386.

GENESARETH, M.; KETCHPEL, S.: **Software Agents**. Communications of ACM n.7, v.4, july 1994. P.48-53.

GESZYCHTER, M.; JACOMINO, A. **Massyve Kit – Um Ambiente de Suporte ao Desenvolvimento de Aplicações Multiagente**. Trabalho de Conclusão (Bacharelado em Ciências da Computação) – Departamento de Informática e Estatística, Universidade Federal de Santa Catarina, Florianópolis, 1999.

GSI. **Sistemas Especialistas Aplicados à Medicina – Introdução**. Site desenvolvido pelo Grupo de Sistemas Inteligentes, DIN – UEM. Disponível em: <<http://www.din.uem.br/~ia/medicina/introducao.html>>. Acesso em: 16/10/2002.

HOGARTH, M. Informática Médica: Um pouco de História. **Revista de Informática Médica**, v.1, n.5, 1998. Disponível em: <<http://www.epub.org.br/informaticamedica/n0105/hogarth.htm>>. Acesso em: 03/11/2002.

HON - Health On the Net Foundation. **Código de Conduta (HON Code) para sites de medicina e saúde**. Disponível em: <<http://www.hon.ch/HONcode/Portuguese>>. Acesso em: 20/10/2002.

IEB-UFSC. **Instituto de Engenharia Biomédica – IEB - UFSC**. Disponível em: <<http://domino.ieb.ufsc.br/Portal2.nsf?OpenDatabase>>. Acesso em: 01/11/2002.

JENNINGS, N. - **Cooperation in Industrial Multi-Agent Systems**, World Scientific Series in Computer Science (Vol 43), 1994.

KOEHLER, C. **Uma Abordagem Probabilística para Sistemas Especialistas**. 1998. Dissertação (Mestrado em Ciências da Computação) – Departamento de Informática e Estatística, Universidade Federal de Santa Catarina, Florianópolis, SC.

MAES, P. **Agents that Reduce Work and Information Overload**. Communications of the ACM, vol. 37, nro. 7, julho 1994. Disponível em: <http://pattie.www.media.mit.edu/people/pattie> . Acesso em: 22/10/2002.

MOULIN, B.; CHAIB-DRAA, B. An Overview of Distributed Artificial Intelligence. In: O'HARE, G.; JENNINGS, N. (Eds.) **Foundations of Distributed Intelligence**. New York. John Wiley, 1996. p. 3-47.

NASSAR, S.M.. **Tratamento de Incerteza: Sistemas Especialistas Probabilísticos (Apostila)**. Florianópolis, 2000. Departamento de Pós-Graduação em Ciências da Computação, Universidade Federal de Santa Catarina.

RABELO, R.; CAMARINHA-MATOS, L.M. - **HOLOS : a methodology for deriving scheduling systems, em Balanced Automation Systems - Architectures and Design Methods**, Eds. Luis M. Camarinha-Matos e Hamideh Afsarmanesh, Chapman & Hall, pp. 181-194, 1995.

RABELO, R.J. - **A Framework for the Development of Manufacturing Agile Scheduling Systems – A Multi-agent Approach**, Ph.D. Thesis, New University of Lisbon, Portugal, 1997.

REIS, L. **SANEP – Sistema Especialista Probabilístico de Apoio a Nutrição Enteral Pediátrica**. Florianópolis, 2001. Dissertação(Mestrado em Ciências da Computação) – Departamento de Informática e Estatística, Universidade Federal de Santa Catarina.

RABUSKE, R. **Inteligência Artificial**. Florianópolis: Editora da UFSC, p.240, 1995.

REIS, L.; CARGNIN, M. **SDDEP – Uma Aplicação na Área Médica Utilizando Raciocínio Baseado em Casos**. Trabalho de Conclusão (Bacharelado em Ciências da Computação) – Departamento de Informática e Estatística, Universidade Federal de Santa Catarina, Florianópolis, 1997.

SABBATINI, R.M.E. **Microcomputers Applications in Medicine: a Review**. Genova: World Health Organization, Information Systems Support Division, 1985.

SABBATINI, R.M.E. **Uso do Computador no Apoio ao Diagnóstico Médico**. Revista Informédica, v.1, n.1, 1993.

SABBATINI, R.M.E. História da Informática em Saúde no Brasil. **Revista de Informática Médica**, v.1, n.5, 1998. Disponível em: <<http://www.epub.org.br/informaticamedica/n0105/sabbatini.htm>>. Acesso em: 03/11/2002

SICHMAN, Jaime Simão; DEMAZEAU, Yves; BOISSIER, Olivier. **How can knowledge-based systems be called agents?** In: SIMPÓSIO BRASILEIRO DE INTELIGÊNCIA ARTIFICIAL, IX, 1992, Rio de Janeiro. Anais do IX SIMPÓSIO BRASILEIRO DE INTELIGÊNCIA ARTIFICIAL, Pontifícia Universidade Católica do Rio de Janeiro, 1992. p. 173-185.

SHOHAM, Y. - **Agent-Oriented Programming**, Artificial Intelligence, N 60, pp-51-92, Elsevier, 1993.

SHORTLIFFE, E.H.; PERRAULT, E.. **Medical Informatics: Computer Applications in Helth Care**. Addison Wesley, 1990.

SIMÕES, P.W.T.A. **SACI – Sistema de apoio na Avaliação de Distúrbios do Crescimento Infantil**, Florianópolis, 2001. Dissertação (Mestrado em Ciências da Computação) – Departamento de Informática e Estatística, Universidade Federal de Santa Catarina.

SMITH, S. - **OPIS: A Methodology and Architecture for Reactive Scheduling**, in Intelligent Scheduling, Eds. M. ZWeben & M. Fox, Morgan Kaufmann, pp. 29-66, 1994.

SOUZA, I. Nutrição Enteral. MedPress **Revista Médica Virtual**, publicado em 20/10/1998. Disponível em: <<http://www.medpress.med.br/art/nutricao.htm>>. Acesso em: 21/10/2002

SOUZA, E.M.S. **Uma Estrutura de Agentes para Acessória na Internet**. 1997. Dissertação (Mestrado em Engenharia de Produção) – Programa de Pós-Graduação, em Engenharia de Produção da Universidade Federal de Santa Catarina, Florianópolis, SC.

STEIN, C. E. **Sistema Especialista Probabilístico: Base de Conhecimento Dinâmica**. Florianópolis, 2000. Dissertação (Mestrado em Ciências da Computação) – Departamento de Informática e Estatística, Universidade Federal de Santa Catarina.

SZELKE, E.; KERR, R. - **Knowledge-Based Reactive Scheduling**, Int. Journal of Production Planning & Control, Vol 5 N 2, Taylor & Francis, pp. 124-145, 1994.

WITTIG, T. (editor) - **ARCHON : An Architecture for Multi-agent Systems**, Ellis Horwood, 1992.

WOOLDRIDGE, M.; JENNINGS, N. **Intelligent Agents: Theory and Practice**. Knowledge Engineering Review v.10, nro. 2, junho 1995. Disponível em: <<http://www.doc.mmu.ac.uk/STAFF/mike/ker95.ps>>. Acesso em: 17/10/2002.

ANEXO A: LINGUAGEM DE COMUNICAÇÃO DOS AGENTES

CONTEÚDO DAS MENSAGENS MANIPULADAS PELOS AGENTES ESPECIALISTAS

Recebe	Ordem (Significado)	Executa	Responde (Envia)
Ao ser lançado (início da execução)			
-	Informar lançamento	iniciaAgente();	-
Durante as interações normais da execução do Sistema Multiagente CLIDENP			
DIAG	Realizar diagnóstico	sugereDiagnostico();	DIAG_OK
CHECK	Informar se Ativo	checkAgente();	CHECKED
BYE	Encerrar execução	FormAgente.Close();	-
Antes de encerrar sua execução			
-	Informar encerramento	finalizaAgente();	-

CONTEÚDO DAS MENSAGENS MANIPULADAS PELO AGENTE GERENTE

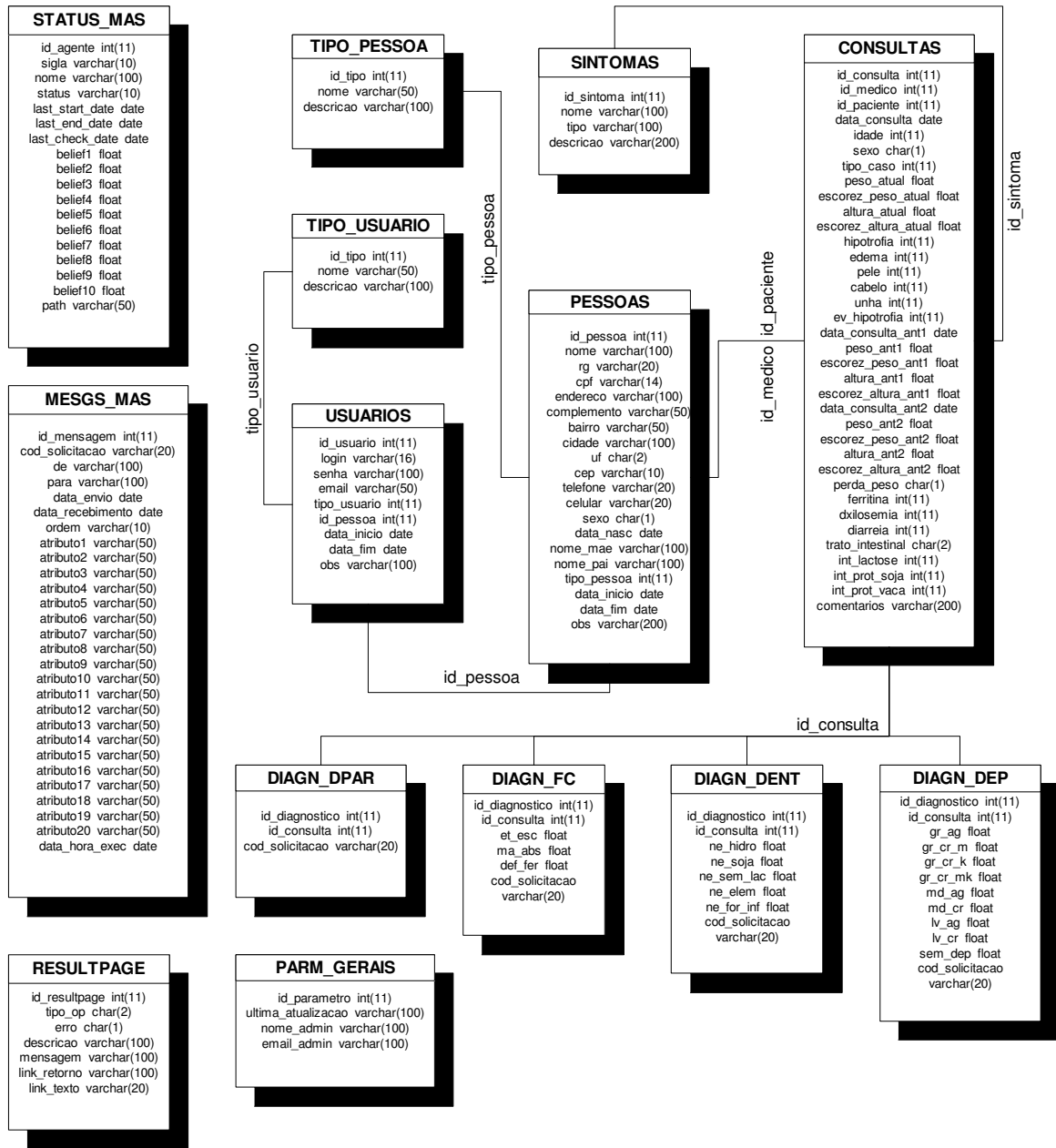
Recebe	Ordem (Significado)	Executa	Responde (Envia)
Ao ser lançado (início da execução)			
-	Informar lançamento	iniciaAgente();	-
Durante as interações normais da execução do Sistema Multiagente CLIDENP			
EXECALL	Lançar Agentes	execAll();	EXECALL_OK
EXEC	Lançar um Agente	exec();	EXEC_OK
KILLALL	Encerrar Agentes	killAll();	-
KILL	Encerrar um Agente	kill();	-
CHECK	Informar se Ativo	checkAgente();	CHECKED
BYE	Encerrar execução	FormAgente.Close();	-
Antes de encerrar sua execução			
-	Informar	finalizaAgente();	-

	encerramento		
--	--------------	--	--

CONTEÚDO DAS MENSAGENS MANIPULADAS PELO AGENTE INTERFACE

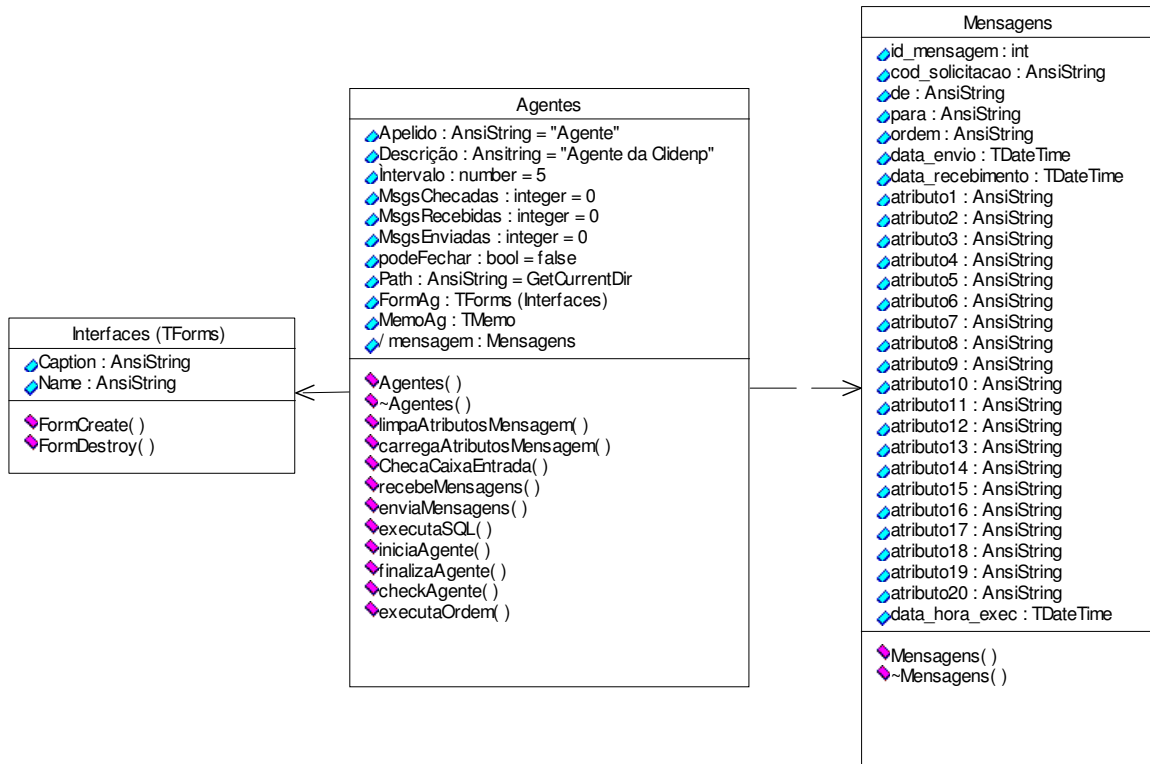
Recebe	Ordem (Significado)	Executa	Responde (Envia)
Durante as interações normais da execução do Sistema Multiagente CLIDENP			
DIAGDEP_OK	Diagnóstico da DEP foi realizado com sucesso.	mostradep.php	-
DIAGFC_OK	Diagnóstico da FC foi realizado com sucesso.	mostrafc.php	-
DIAGNE_OK	Diagnóstico da NE foi realizado com sucesso.	mostrane.php	-
CHECKED	Agente está ativo e funcionando.	mostrastatus.php	-

ANEXO B: MODELO DE ENTIDADE E RELACIONAMENTO DA BASE DE DADOS CLIDENP

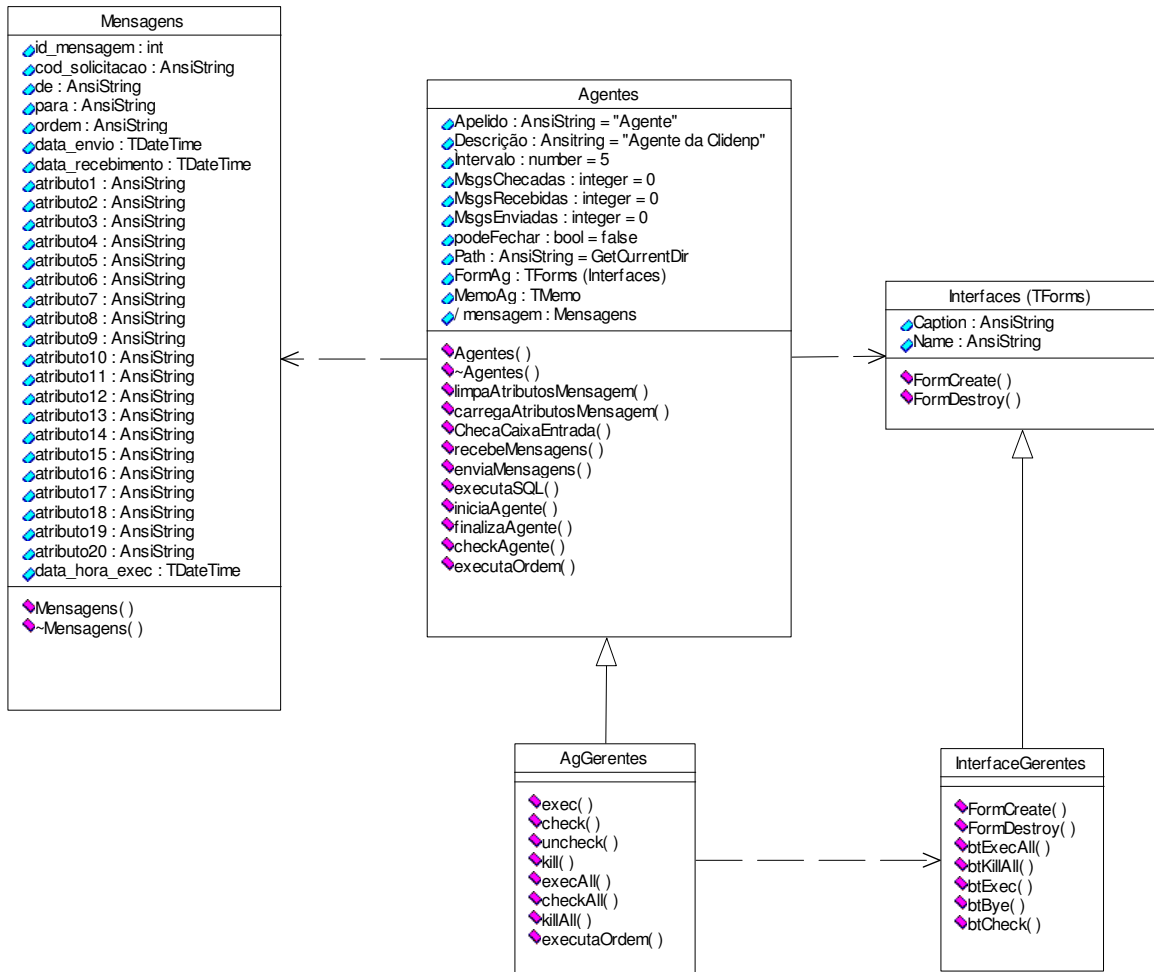


ANEXO C: MODELOS DE OBJETOS DA CLIDENP

MODELO DO AGENTE PADRÃO



MODELO DO AGENTE GERENTE



MODELO DOS AGENTES ESPECIALISTAS

